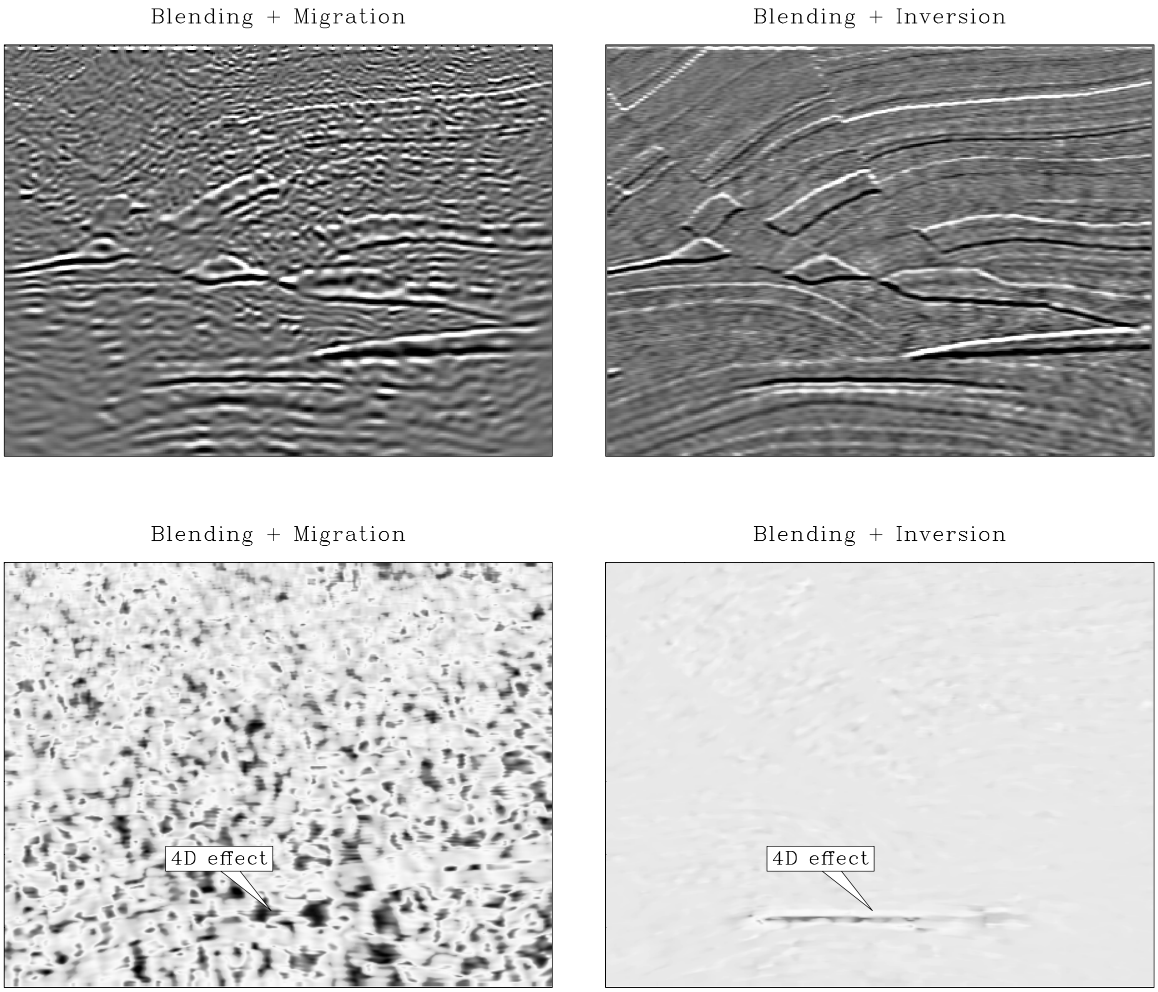
# STANFORD EXPLORATION PROJECT

*Abdullah Al Theyab, Gboyega Ayeni, James Berryman, Biondo Biondi, Robert Clapp, Haohuan Fu, Claudio Guerra, Seth Haines, Adam Halpert, Yunyue Li, Mohammad Maysami, Oskar Mencer, Nader Moussa, Oliver Pell, Sjoerd de Ridder, Shuki Ronen, Xukai Shen, Jeff Shragge, Yaxun Tang, and Mandy Wong*

**Report Number 138, May 2009**



# Preface

The electronic version of this report[1] 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 library[2]. We assume you have a UNIX workstation with Fortran, Fortran90, 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, 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 Fortran90 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 online[3].

[1] http://sepwww.stanford.edu/private/docs/sep138

[2] http://sepwww.stanford.edu/public/docs/sepdatalib/toc_html

[3] http://sepwww.stanford.edu/research/redoc/

# SEP138 — TABLE OF CONTENTS

## Novel imaging techniques

## Velocity model building

## Hessian based inversion

## Near surface

## Beyond P-waves

## Computation

# Delayed-shot migration in TEC coordinates

*Jeff Shragge*

## ABSTRACT

This paper extends the analytical Riemannian wavefield extrapolation (RWE) approach to 3D coordinate systems. I formulate an inline delayed-shot migration procedure in tilted elliptical-cylindrical (TEC) coordinate systems. When inline coordinate tilt angles are well-matched to the inline source ray parameters, the TEC coordinate extension affords accurate propagation of both steep-dip and turning-wave components. I show that wavefield extrapolation in TEC coordinates is no more complicated than propagation in elliptically anisotropic media. Impulse response tests illustrate the accuracy and lack of numerical anisotropy of the implemented scheme. I apply this approach to a realistic 3D wide-azimuth synthetic derived from a field Gulf of Mexico data set. The resulting images demonstrate the imaging advantages made possible through 3D RWE implementations, including the improved imaging of steeply dipping salt flanks, potentially at a reduced computational cost. Narrow-azimuth migration results demonstrate the applicability of the approach to typical Gulf of Mexico field data.

## INTRODUCTION

Wave-equation migration (WEM) methods routinely generate accurate seismic images in areas of complex geology. One common class of WEM approaches is shot-profile migration using one-way wavefield extrapolation. The first shot-profile migration step is to specify source and receiver wavefields that consist of modeled point sources and an individual shot profile, respectively. The migration algorithm propagates these two wavefields through the velocity model and correlates them at each extrapolation step to form an image. Although this procedure generates high-quality migration results, two drawbacks make shot-profile migration a less-than-ideal strategy. The first issue is that each individual shot migration requires a large aperture to propagate energy to wide offsets. The second drawback is that one migrates each shot record individually, which can be computationally expensive for large surveys with a high shot density.

One way to make the shot-profile style of WEM more efficient is to migrate a reduced number of composite source and receiver profiles each covering a broader aperture. For example, one can image a number of shot profiles simultaneously on the same migration domain. The key idea is that one makes a computationally advantageous trade-off of a broader migration aperture for a reduced number of shots. Shot-profile migration with composite wavefields, though, leads to the mixing of information from different shots and generates image crosstalk. A number of authors address this problem using a variety of phase-encoding migration approaches (Morton and Ober, 1998; Jing et al., 2000; Romero et al., 2000; Sun et al., 2002), that minimize the deleterious crosstalk effects.

Plane-wave migration (PWM) is one technique for reducing total migration cost using composite wavefields (Whitmore, 1995; Mosher and Foster, 1998; Duquet et al., 2001; Zhang et al., 2003; Liu et al., 2004, 2006). As originally demonstrated by Whitmore (1995), the key idea is to synthesize from the full wavefield volume the set of composite receiver wavefields that would have been recorded were a planar source function used. Generally, the number of synthesized wavefields is fewer than the corresponding number of shot profiles. One generates PWM images by propagating the modeled planar source and composite receiver wavefields through the velocity and computing a (weighted) correlation. Liu et al. (2006) and Duquet and Lailly (2006) demonstrate that PWM is equivalent to shot-profile migration in the limit where one uses many plane waves with well-sampled plane-wave dip spectra. Liu et al. (2006) also prove that 3D PWM is equivalent to conical-wave migration of individual sail lines synthesized as inline composite wavefields. The approach is termed conical wave because the source wavefronts form conic sections (in constant media) for non-zero inline plane-wave ray parameters.

The migration of plane- and conical-wave data, though more efficient than shot-profile migration, is similarly restricted in accuracy by one-way wavefield extrapolation assumptions. The most common limitation is a difficulty in propagating waves at large angles and turning waves by design, both of which are important for accurately imaging salt flanks in complex geologic areas. Shan and Biondi (2004) circumvent this problem by implementing 3D PWM in tilted Cartesian meshes. This coordinate system effectively orients the wavefield extrapolation axis toward the plane-wave take-off vector, enabling more accurate bulk propagation of plane-wave energy. One logistical complication of performing fully 3D PWM is that it requires propagating image-space-sized data volumes on a number of meshes tilting in both the inline and cross-line directions. This leads to a number of computational issues associated with the significant memory footprint.

This paper presents an alternative to the phase-encoding approach of Shan and Biondi (2004), which similarly uses alternative coordinate systems. The key differences between these two approaches are two-fold. The first difference is that I phase encode only according to the inline source coordinate, leading to the inline delayed-shot migration algorithm. This leads to a straightforward coarse-grain parallelization of the migration tasks across individual sail lines, where each migration has a significantly smaller aperture than the corresponding image-space-sized PWM volumes. A second efficiency gain over PWM is a reduction in the total number of migrations, because the number of sail lines is quite often fewer than the required number of cross-line plane waves. Thus, the inline-delayed shot approach has attractive computational advantages over the 3D PWM technique.

The second difference is that I migrate data in tilted elliptical-cylindrical (TEC) coordinates, rather than tilted Cartesian meshes. The key idea is that, because the geometry of the TEC coordinate system closely resembles the shape of a line-source impulse response, TEC meshes afford accurate propagation of most steep-dip and turning waves in all directions. TEC coordinate systems, formed by concatenating a set of the 2D elliptical coordinates (Shragge and Shan, 2008) along the invariant third axis, are thus well-suited for migrating individual sail lines. I extrapolate the inline delay-shot synthesized wavefield volumes outward on a series of elliptical-cylindrical shells. This allows source and receiver wavefields with zero inline dip to overturn in the cross-line direction, if necessary. I introduce an extra degree of freedom that permits the coordinate system to tilt along the invariant inline axis, thus enabling the propagation of turning waves inline. Consequently, inline delayed-shot migration in TEC coordinates allows wavefields with most non-zero dips to propagate and overturn to all azimuths as appropriate.

The paper begins by examining 3D full-plane-wave and inline delayed-shot migration theory. I then introduce the TEC coordinate geometry and develop the corresponding wavenumber that forms the basis of the TEC wavefield extrapolation operator. I discuss the finite-difference extrapolation implementation and present the 3D impulse response. I apply the technique to a 3D wide-azimuth synthetic data set derived from real Gulf of Mexico velocity model to demonstrate the imaging advantages of 3D RWE migration. I then discuss the numerical costs associated with performing inline delayed-shot migration in TEC coordinates relative to Cartesian meshes. The paper concludes with narrow-azimuth migration results demonstrate the applicability of the approach to typical Gulf of Mexico field data.

## 3D PLANE-WAVE MIGRATION

The full plane-wave and inline delayed-shot migration theory discussed herein draws largely from Liu et al. (2006). I restate a number of key points for completeness, though with a slightly different notation. As in previous chapters, I define Cartesian coordinates by $\boldsymbol{x} = [x_1, x_2, x_3]$ and a generalized coordinate system by $\boldsymbol{\xi} = [\xi_1, \xi_2, \xi_3]$.

### Full plane-wave phase-encoding migration

Performing 3D plane-wave migration is similar in many respects to 3D shot-profile migration. The main differences derive from how the composite source and receiver wavefield volumes, $\overline{S}$ and $\overline{R}$, are re-synthesized from individual source and receiver profiles, $S_{jk}$ and $R_{lm}$, prior to imaging. The complete wavefields are generated by filtering the source and receiver profiles by a function dependent on the inline and cross-line plane-wave ray parameters, $\boldsymbol{p_\xi} = [p_{\xi_1}, p_{\xi_2}]$. These wavefields are then propagated through the migration domain to generate the full source and receiver wavefield volumes

$$\overline{S(\boldsymbol{\xi}|\omega)} = \sum_{j=1}^{A}\sum_{k=1}^{B} S_{jk}(\boldsymbol{\xi}|\omega) f(\omega) \mathrm{e}^{i\omega[p_{\xi_1}\Delta\xi_1(j-p)+p_{\xi_2}\Delta\xi_2(k-q)]}, \tag{1}$$

$$\overline{R(\boldsymbol{\xi}|\omega)} = \sum_{l=1}^{A}\sum_{m=1}^{B} R_{lm}(\boldsymbol{\xi}|\omega) f(\omega) \mathrm{e}^{i\omega[p_{\xi_1}\Delta\xi_1(l-p)+p_{\xi_2}\Delta\xi_2(m-q)]}, \tag{2}$$

where $f(\omega)$ is a frequency filter to be discussed below, $\Delta\xi_1$ and $\Delta\xi_2$ are the inline and cross-line sampling intervals, $p$ and $q$ are reference spatial indices in the inline and cross-line directions, $j$ and $k$ are indices fixing the inline and crossline source position, $l$ and $m$ are indices fixing the inline and cross-line receiver position, and $A$ and $B$ are the number of inline and cross-line source records, respectively. The phase encoding, implemented at the surface independent of wavefield extrapolation, is valid for any generalized coordinate system. Note that the wavefield propagation throughout the migration volume in equations 1 and 2 is understood, and assumed to be governed by the wavefield propagation techniques described in Shragge (2008).

An image volume $I(\boldsymbol{\xi})$ is formed from a series of individual full plane-wave migration images, $I^{PW}(\boldsymbol{\xi}|\boldsymbol{p_\xi})$, by correlating the composite plane-wave source and receiver wavefields and stacking the results over frequency. The plane-wave migration kernel mixes source and receiver wavefield energy, $S_{jk}(\boldsymbol{\xi}|\omega)$ and $R_{lm}(\boldsymbol{\xi}|\omega)$, according to

$$\begin{aligned} I(\boldsymbol{\xi}) &= \sum_{p_{\xi_1}} \sum_{p_{\xi_2}} \sum_{j,l=1}^{A} \sum_{k,m=1}^{B} I^{PW}_{jklm}(\boldsymbol{\xi}|\boldsymbol{p_\xi}) \qquad (3) \\ &= \sum_{p_{\xi_1}} \sum_{p_{\xi_2}} \sum_{j,l=1}^{A} \sum_{k,m=1}^{B} \sum_{\omega} |f(\omega)|^2 \, S^*_{jk}(\boldsymbol{\xi}|\omega) \, R_{lm}(\boldsymbol{\xi}|\omega) \mathrm{e}^{i\omega[p_{\xi_1}\Delta\xi_1(j-l)+p_{\xi_2}\Delta\xi_2(k-m)]}, \end{aligned}$$

where $^*$ indicates complex conjugate.

Generally, mixing wavefields of differing $S_{jk}$ and $R_{lm}$ indices introduces image crosstalk. A plane-wave migration image will be crosstalk-free, though, in the following limits:

$$\begin{aligned} \lim_{N_{p_{\xi_1}}\to\infty} \sum_{\alpha=-N_{p_{\xi_1}}}^{N_{p_{\xi_1}}} \mathrm{e}^{i\omega\alpha\Delta p_{\xi_1}\Delta\xi_1(j-l)} &= |\omega|^{-1}\delta_{jl}, \\ \lim_{N_{p_{\xi_2}}\to\infty} \sum_{\alpha=-N_{p_{\xi_2}}}^{N_{p_{\xi_2}}} \mathrm{e}^{i\omega\alpha\Delta p_{\xi_2}\Delta\xi_2(k-m)} &= |\omega|^{-1}\delta_{km}. \qquad (4) \end{aligned}$$

where $N_{p_{\xi_1}}$ and $N_{p_{\xi_2}}$ are the number of plane waves in the $\xi_1$ and $\xi_2$ directions. Assuming that equation 4 approximately is valid (i.e., for large values of $N_{p_{\xi_1}}$ and $N_{p_{\xi_2}}$), I rewrite equation 3 as

$$I(\boldsymbol{\xi}) \approx \sum_{j=1}^{A} \sum_{k=1}^{B} \sum_{\omega} |f(\omega)|^2 |\omega|^{-2} S^*_{jk}(\boldsymbol{\xi}|\omega) R_{jk}(\boldsymbol{\xi}|\omega), \qquad (5)$$

which, by defining $|f(\omega)|^2 = |\omega|^2$, generates the following expression:

$$I(\boldsymbol{\xi}) \approx \sum_{j=1}^{M} \sum_{k=1}^{N} \sum_{\omega} S^*_{jk}(\boldsymbol{\xi}|\omega) R_{jk}(\boldsymbol{\xi}|\omega). \qquad (6)$$

This demonstrates the equivalence between plane-wave and shot-profile migration (Liu et al., 2006).

## Inline delayed-shot migration

An alternate 3D migration formulation is to phase-encode individual sail lines for a given ray parameter, $p_{\xi_1}$, solely according to inline source position. This phase-encoding approach is related to conical-wave migration, which requires $j - l = 0$ in equation 3. However, I choose to not make this restriction because it is realized only by straight sail lines and non-flip-flop sources (Liu et al., 2006). Rather, I present an alternative theory of inline delayed-shot migration that allows more general crossline source and receiver distribution.

Inline delayed-shot wavefields, propagated through the migration domain to generate the full source and receiver wavefield volumes, are defined by

$$\overline{S(\boldsymbol{\xi}|\omega)} = \sum_{l=1}^{A}\sum_{j=1}^{B} S_{jl}(\boldsymbol{\xi}|\omega) f(\omega) \mathrm{e}^{i\omega[p_{\xi_1}\Delta\xi_1(j-p)]}, \tag{7}$$

$$\overline{R(\boldsymbol{\xi}|\omega)} = \sum_{l=1}^{A}\sum_{k=1}^{B} R_{kl}(\boldsymbol{\xi}|\omega) f(\omega) \mathrm{e}^{i\omega[p_{\xi_1}\Delta\xi_1(k-p)]}, \tag{8}$$

where $j$ and $k$ are the source and receiver inline position, respectively, $B$ is the number of inline records, $l$ is the sail line index out of a total of $A$ sail lines, and $p$ is a reference inline index.

An image volume $I(\boldsymbol{\xi})$ is generated from a series of inline delayed-shot migration images, $I_l^{DS}(\boldsymbol{\xi}|p_{\xi_1})$, formed by correlating the composite inline source and receiver wavefields and stacking the results over frequency. The inline delayed-shot migration kernel mixes source and receiver wavefield energy, $S_{jl}(\boldsymbol{\xi}|\omega)$ and $R_{kl}(\boldsymbol{\xi}|\omega)$, according to

$$\begin{aligned} I(\boldsymbol{\xi}) &= \sum_{l=1}^{A}\sum_{p_{\xi_1}}\sum_{j=1}^{B}\sum_{k=1}^{B} I_{jkl}^{DS}(\boldsymbol{\xi}|p_{\xi_1}) \\ &= \sum_{l=1}^{A}\sum_{p_{\xi_1}}\sum_{j=1}^{B}\sum_{k=1}^{B}\sum_{\omega} |f(\omega)|^2\, S_{jl}^{*}(\boldsymbol{\xi}|\omega)\, R_{kl}(\boldsymbol{\xi}|\omega) \mathrm{e}^{i\omega[p_{\xi_1}\Delta\xi_1(j-k)]}, \end{aligned} \tag{9}$$

Similar to plane-wave migration, mixing wavefields of differing $S_{jl}$ and $R_{kl}$ indices will introduce crosstalk into the image volume. However, inline delayed-shot migration will be crosstalk-free in the following limit:

$$\lim_{N_{p_{\xi_1}}\to\infty} \sum_{\alpha=-N_{p_{\xi_1}}}^{N_{p_{\xi_1}}} \mathrm{e}^{i\omega\alpha\Delta p_{\xi_1}\Delta\xi_1(j-k)} = |\omega|^{-1}\delta_{jk}, \tag{10}$$

Defining $|f(\omega)|^2 = |\omega|$ and using the approximation in equation 10, I rewrite

$$I_l^{DS}(\boldsymbol{\xi}) \approx \sum_{j=1}^{B}\sum_{\omega} S_{jl}^{*}(\boldsymbol{\xi}|\omega) R_{jl}(\boldsymbol{\xi}|\omega). \tag{11}$$

Stacking over all inline delayed-shot sail-line migration results yields the full image volume,

$$I(\boldsymbol{\xi}) \approx \sum_{l=1}^{A} I_l^{DS}(\boldsymbol{\xi}) \approx \sum_{l=1}^{A}\sum_{j=1}^{B}\sum_{\omega} S_{jl}^{*}(\boldsymbol{\xi}|\omega) R_{jl}(\boldsymbol{\xi}|\omega). \tag{12}$$

This proves the equivalence of inline delayed-shot and shot-profile migration.

## TILTED ELLIPTICAL-CYLINDRICAL COORDINATES

One question to be addressed is what coordinate system geometry optimally conforms to the impulse response of a conical wavefield? I assert that the best geometry is that of the TEC coordinate system shown in Figures 1 and 2. One advantage is that the breadth of the first extrapolation step at the surface allows multiple streamers of a single sail line to be positioned directly on a single mesh. Hence, this geometry is applicable to both narrow- and wide-azimuth acquisition. A second advantage is that one direction of large-angle propagation can be handled by coordinate system tilting, while the other is naturally handled by the ellipticality of the mesh. (Note that the geometry of another natural mesh - cylindrical polar coordinates - would not be a judicious choice for because the geometry permits migration of only single-streamer data and has singular points located on the surface at the first extrapolation step.)

I set up the migration geometry of the elliptical-cylindrical mesh as follows:

- $\xi_3 \in [0, \infty]$ is the extrapolation direction, where surfaces of constant $\xi_3$ form concentric elliptical cylinders, shown in Figure 1a.
- $\xi_2 \in [0, 2\pi)$ is the crossline direction, where surfaces of constant $\xi_2$ are folded hyperbolic planes, shown in Figure 1b; and
- $\xi_1 \in [-\infty, \infty]$ is the inline direction, where surfaces of constant $\xi_1$ are 2D elliptical coordinate meshes, shown in Figure 1c;

The mapping relationship between the two coordinate systems, adapted from Arfken (1970), is

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \xi_1 \cos\theta - a \sinh \xi_3 \sin \xi_2 \sin\theta \\ a \cosh \xi_3 \cos \xi_2 \\ \xi_1 \sin\theta + a \sinh \xi_3 \sin \xi_2 \cos\theta \end{bmatrix}, \tag{13}$$

where $\theta$ is the inline tilt angle of the coordinate system and parameter $a$ controls the coordinate system breadth. Panels 2a and 2b show the TEC coordinate system at $0^\circ$ and $25^\circ$ tilt angles, respectively.

## TEC extrapolation wavenumber

A metric tensor $g_{jk}$ can be specified from the mapping relationship given in equations 13:

$$[g_{jk}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & A^2 & 0 \\ 0 & 0 & A^2 \end{bmatrix}, \tag{14}$$

where $A = a\sqrt{\sinh^2 \xi_3 + \sin^2 \xi_2}$. The determinant of the metric tensor is: $|\boldsymbol{g}| = A^4$. The corresponding inverse weighted metric tensor, $m^{jk}$ as developed in Shragge (2008), is given by:

$$\left[m^{jk}\right] = \begin{bmatrix} A^2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{15}$$

Note that even though the metric of the TEC coordinate system varies spatially, the local curvature parameters ($n^j = \frac{\partial m^{jk}}{\partial \xi_k}$) remain constant: $n^1 = n^2 = n^3 = 0$. The corresponding extrapolation wavenumber, $k_{\xi_3}$, can be generated by inputting tensor $m^{jk}$ and fields $n^j$ into the general wavenumber expression for 3D non-orthogonal coordinate systems

$$k_{\xi_3} = \pm\sqrt{A^2 s^2 \omega^2 - A^2 k_{\xi_1}^2 - k_{\xi_2}^2}, \tag{16}$$

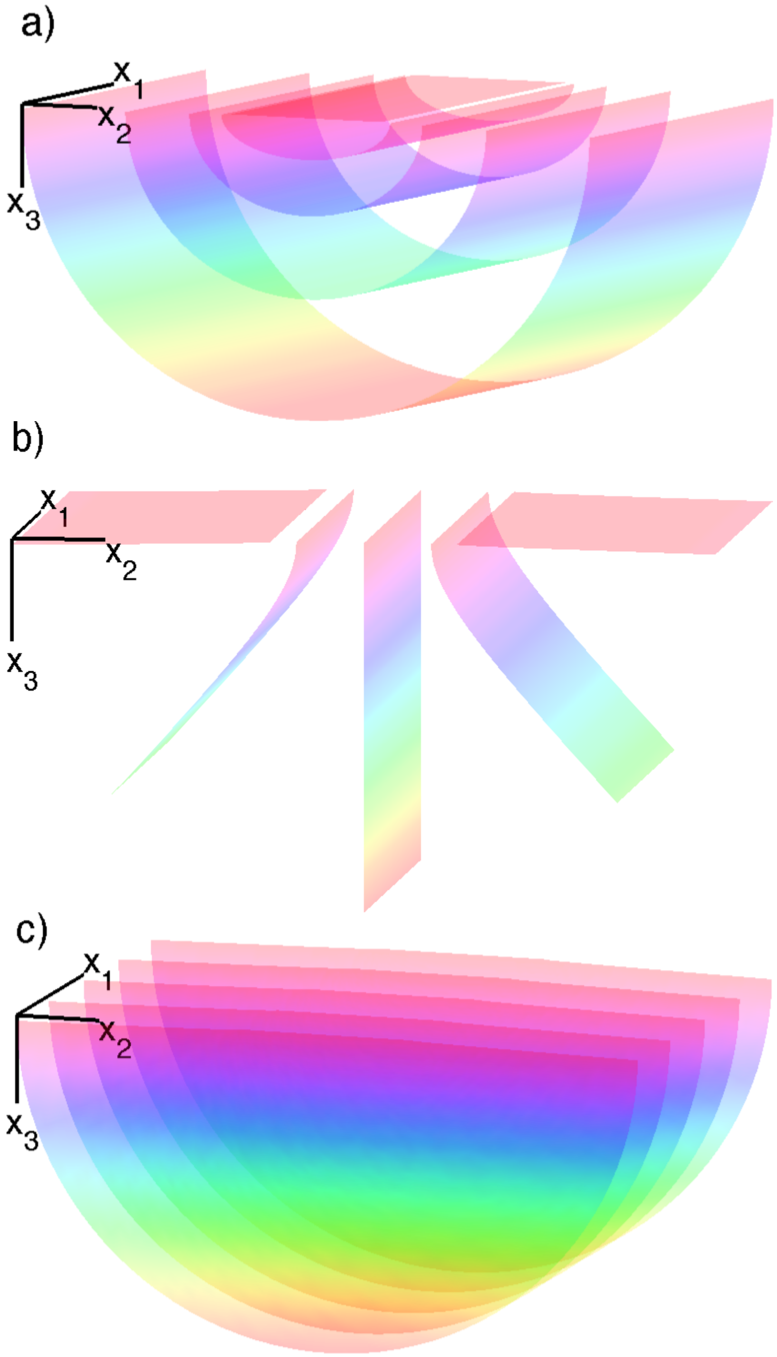


Figure 1: Constant surfaces of the elliptical-cylindrical coordinate system (with zero inline tilt). Cartesian coordinate axes are given by the vector diagram. a) constant $\xi_3$ surfaces forming confocal elliptical-cylindrical shells that represent the direction of extrapolation direction. b) constant $\xi_2$ surfaces representing folded hyperbolic planes. c) constant $\xi_1$ surfaces representing 2D elliptical meshes. **NR** jeff1/. TECgeom

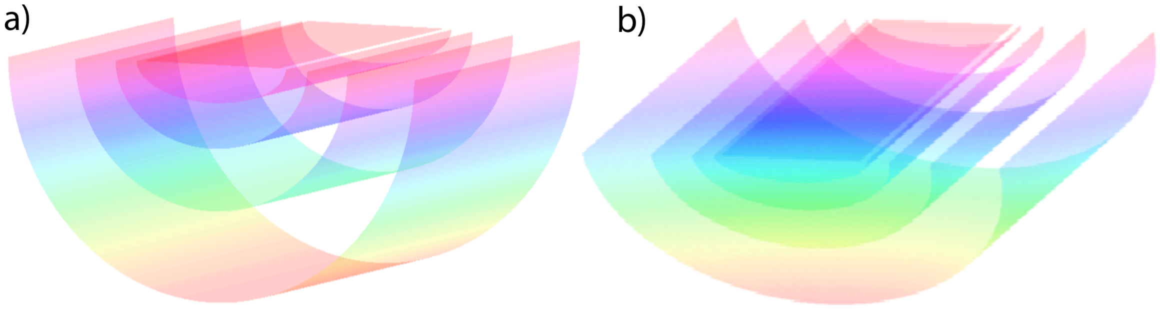


Figure 2: Four extrapolation steps in $\xi_3$ of an TEC coordinate system, where the $\xi_1$ and $\xi_2$ coordinate axes are oriented in the inline and crossline directions, respectively. a) 0° tilt angle. b) 25° tilt angle. **NR** jeff1/. TEC

where $s$ is the slowness (reciprocal of velocity), $k_{\xi_3}$ is the extrapolation wavenumber, and $k_{\xi_1}$ and $k_{\xi_2}$ are the inline and crossline wavenumbers, respectively.

The wavenumber specified in equation 16 is central to the inline delayed-shot migration algorithm. The first step is to extrapolate the source and receiver wavefields

$$E_{\xi_3}[S_{jl}(\xi_3,\xi_1,\xi_2|\omega)] = S_{jl}(\xi_3+\Delta\xi_3,\xi_1,\xi_2|\omega), \quad (17)$$

$$E^*_{\xi_3}[R_{kl}(\xi_3,\xi_1,\xi_2|\omega)] = R_{kl}(\xi_3+\Delta\xi_3,\xi_1,\xi_2|\omega), \quad (18)$$

where $E_{\xi_3}[\cdot]$ and $E^*_{\xi_3}[\cdot]$ are the extrapolation operator and its conjugate, respectively. The results herein were computed using the $\omega-\xi$ finite-difference extrapolators discussed below. The second step involves summing the individual inline delayed-shot images contributions, $I^{DS}_{jk}(\boldsymbol{\xi})$, into the total image volume, $I(\boldsymbol{\xi})$ according to equation 12.

## 3D IMPLICIT FINITE-DIFFERENCE EXTRAPOLATION

One obvious concern is whether the dispersion relationship in equation 16 can be implemented accurately and efficiently in a wavefield extrapolation scheme. I address this question by comparing the elliptical-cylindrical dispersion relationship to that for elliptically anisotropic media in Cartesian coordinates. By defining an effective slowness $s_A = As$ and rewriting equation 16 as

$$\frac{k_{\xi_3}}{\omega s_A} = \sqrt{1 - A^2\frac{k^2_{\xi_1}}{\omega^2 s^2_A} - \frac{k^2_{\xi_2}}{\omega^2 s^2_A}}, \quad (19)$$

the TEC coordinate dispersion relationship resembles that of elliptically anisotropic media (Tsvankin, 1996). More specifically, extrapolation in TEC coordinates is related to a special case where the Thomsen parameters (Thomsen, 1986) obey $\epsilon = \delta$:

$$\left.\frac{k_{x_3}}{\omega s}\right|_{\epsilon=\delta} = \sqrt{\left.\frac{1-(1+2\epsilon)\frac{k^2_{x_1}+k^2_{x_2}}{\omega^2 s^2}}{1-2(\epsilon-\delta)\frac{k^2_{x_1}+k^2_{x_2}}{\omega^2 s^2}}\right|_{\epsilon=\delta}} = \sqrt{1-(1+2\epsilon)\frac{k^2_{x_1}}{\omega^2 s^2}-(1+2\epsilon)\frac{k^2_{x_2}}{\omega^2 s^2}}. \quad (20)$$

From equation 20 we see that equation 16 is no more complex than the dispersion relationship for propagating waves in elliptically anisotropic media, which is now routinely handled

| Coeff. order $j$ | Coeff. $a_j$ | Coeff. $b_j$ |
|---|---|---|
| 1 | 0.040315157 | 0.873981642 |
| 2 | 0.457289566 | 0.222691983 |

Table 1: Coefficients used in 3D implicit finite-difference wavefield extrapolation.

with finite-difference approaches (Zhang et al., 2001; Baumstein and Anderson, 2003; Shan and Biondi, 2005).

A general approach to 3D implicit finite-difference propagation is to approximate the square-root by a series of rational functions (Ma, 1982)

$$S_{\xi_3} = \sqrt{1 - A^2 S_{\xi_1}^2 - S_{\xi_2}^2} \approx \sum_{j=1}^{n} \frac{a_j S_r^2}{1 - b_j S_r^2}, \tag{21}$$

where $S_{\xi_j} = \frac{k_{\xi_j}}{\omega s_A}$ and $S_r^2 = A^2 S_{\xi_1}^2 + S_{\xi_2}^2$, for $j = 1, 2, 3$, and $n$ is the order of the coefficient expansion.

An optimal set of coefficients can be found by solving an optimization problem (Shan and Biondi, 2005),

$$E(a_j, b_j) = \min \int_0^{\sin\phi} \left[ \sqrt{1 - S_r^2} - \sum_{j=1}^{n} \frac{a_j S_r^2}{1 - b_j S_r^2} \right]^2 \mathrm{d}S_r, \tag{22}$$

where $\phi$ is the maximum optimization angle. I generated the following results using a 4th-order approximation and coefficients found in Table 1 (Lee and Suh, 1985).

Specifying a finite-difference extrapolator operator using the 4th-order approximation is equivalent to solving a cascade of partial differential equations (Shan and Biondi, 2005)

$$\begin{aligned}
\frac{\partial}{\partial \xi_3} U_{\xi_3 + \Delta\xi_3/3} &= \qquad\qquad i\omega s U_{\xi_3}, \\
\frac{\partial}{\partial \xi_3} U_{\xi_3 + 2\Delta\xi_3/3} &= i\omega s \left[ \frac{\frac{a_1}{\omega^2 s^2} \frac{\partial^2}{\partial \xi_1^2}}{1 + \frac{b_1}{\omega^2 s^2} \frac{\partial^2}{\partial \xi_1^2}} + \frac{\frac{a_1}{\omega^2 s_A^2} \frac{\partial^2}{\partial \xi_2^2}}{1 + \frac{b_1}{\omega^2 s_A^2} \frac{\partial^2}{\partial \xi_2^2}} \right] U_{\xi_3 + \Delta\xi_3/3}, \\
\frac{\partial}{\partial \xi_3} U_{\xi_3 + \Delta\xi_3} &= i\omega s \left[ \frac{\frac{a_2}{\omega^2 s^2} \frac{\partial^2}{\partial \xi_1^2}}{1 + \frac{b_2}{\omega^2 s^2} \frac{\partial^2}{\partial \xi_1^2}} + \frac{\frac{a_2}{\omega^2 s_A^2} \frac{\partial^2}{\partial \xi_2^2}}{1 + \frac{b_2}{\omega^2 s_A^2} \frac{\partial^2}{\partial \xi_2^2}} \right] U_{\xi_3 + 2\Delta\xi_3/3}.
\end{aligned} \tag{23}$$

I solve these equations implicitly at each extrapolation step by a finite-difference splitting approach that alternatively advances the wavefield in the $\xi_1$ and $\xi_2$ directions. Splitting methods allow the direct application of the $A$ scaling factor in equation 21 by introducing the original slowness model, $\frac{s_A}{A} = s$, for the $\xi_1$ direction split.

One drawback to finite-difference splitting methods is that they commonly generate numerical anisotropy. To minimize these effects, I apply a Fourier-domain phase-correction filter $L[\cdot]$ (Li, 1991)

$$L[U] = U \mathrm{e}^{i \Delta\xi_3 k_L}, \tag{24}$$

where

$$k_L = \sqrt{1 - \frac{k_{\xi_1}^2}{(\omega s_1^r)^2} - \frac{k_{\xi_2}^2}{(\omega s_2^r)^2}} - \left[ 1 - \sum_{j=1}^{2} \left( \frac{a_j (\frac{k_{\xi_1}}{\omega s_1^r})^2}{1 - b_j (\frac{k_{\xi_1}}{\omega s_1^r})^2} - \frac{a_j (\frac{k_{\xi_2}}{\omega s_2^r})^2}{1 - b_j (\frac{k_{\xi_2}}{\omega s_2^r})^2} \right) \right], \tag{25}$$

and $s_1^r$ and $s_2^r$ are reference slownesses chosen to be the mean value of $s_{eff}^A$ and $s$ defined above, respectively. Note that while this phase-shift correction is explictly correct for v($\xi_3$) media, the Li filter in v($\xi_3, \xi_1, \xi_2$) media is only approximate and will introduce error.

### Impulse response tests

I conducted impulse response tests on a 500x400x400 cube in a homogeneous medium of slowness $s = 0.0005$ sm$^{-1}$. The initial wavefield consisted of three horizontally smoothed point sources at $t$=0.5, 0.75, and 1 s. The impulse responses are expected to consist of three hemispherical surfaces of radii $r$=1000, 1500, and 2000 m.

Figures 3a and 3b show the inline and crossline responses. The three lines overlying the analytic curves show the correct impulse response locations. Note that the impulse responses are restricted at large angles both by the coordinate system boundaries and by the 50 sample cosine-taper function along the edges of the TEC mesh. Figure 4 shows a impulse response slice extracted at 1300 m depth. The symmetric response indicates that the numerical anisotropy from the numerical splitting is accounted for by the Li phase-correction filter.

## 3D WIDE-AZIMUTH SYNTHETIC TESTS

This section presents the inline delayed-shot migration algorithm test results on a wide-azimuth synthetic data set generated from a realistic 3D Gulf of Mexico velocity model. Figure 5 presents some depth slices and sections through the model. The velocity model is comprised of typical Gulf of Mexico sedimentary profile with a velocity gradient of approximately 0.2 s$^{-1}$, with a number of salt bodies of complex 3D geometry characterized by smoothly varying salt tops and steep flanks below overhangs. Key imaging targets include the steep salt flanks and the on-lapping sedimentary units that comprise the likely exploration areas.

Table 2 summarizes the acquisition geometry of the data set. The data used for migration consisted of 72 sail lines separated 250 m apart. Each sail line consists of 100 shots sampled at a 250 m shot interval. The receiver pattern for each shot record contains 321 inline samples with a maximum offset of ±8000 m computed at a 50 m interval, and 161 crossline samples with a maximum offset of ±4000 m at a 50 m interval.

A total of 192 frequencies were selected for migration starting at 1.42 Hz at a sampling rate of 0.075 Hz. Filtered data from each sail line data were transformed into a plane-wave data set by phase-encoding over a range of inline ray parameters, $p_{\xi_1}$. I selected a total of 101 inline ray parameters between ±8.33x10$^{-4}$ sm$^{-1}$ at a sampling rate of 8.33x10$^{-6}$ sm$^{-1}$. Given the 1500 ms$^{-1}$ water velocities at the surface, the maximum values correspond to a surface take-off angle of ±38.7°.

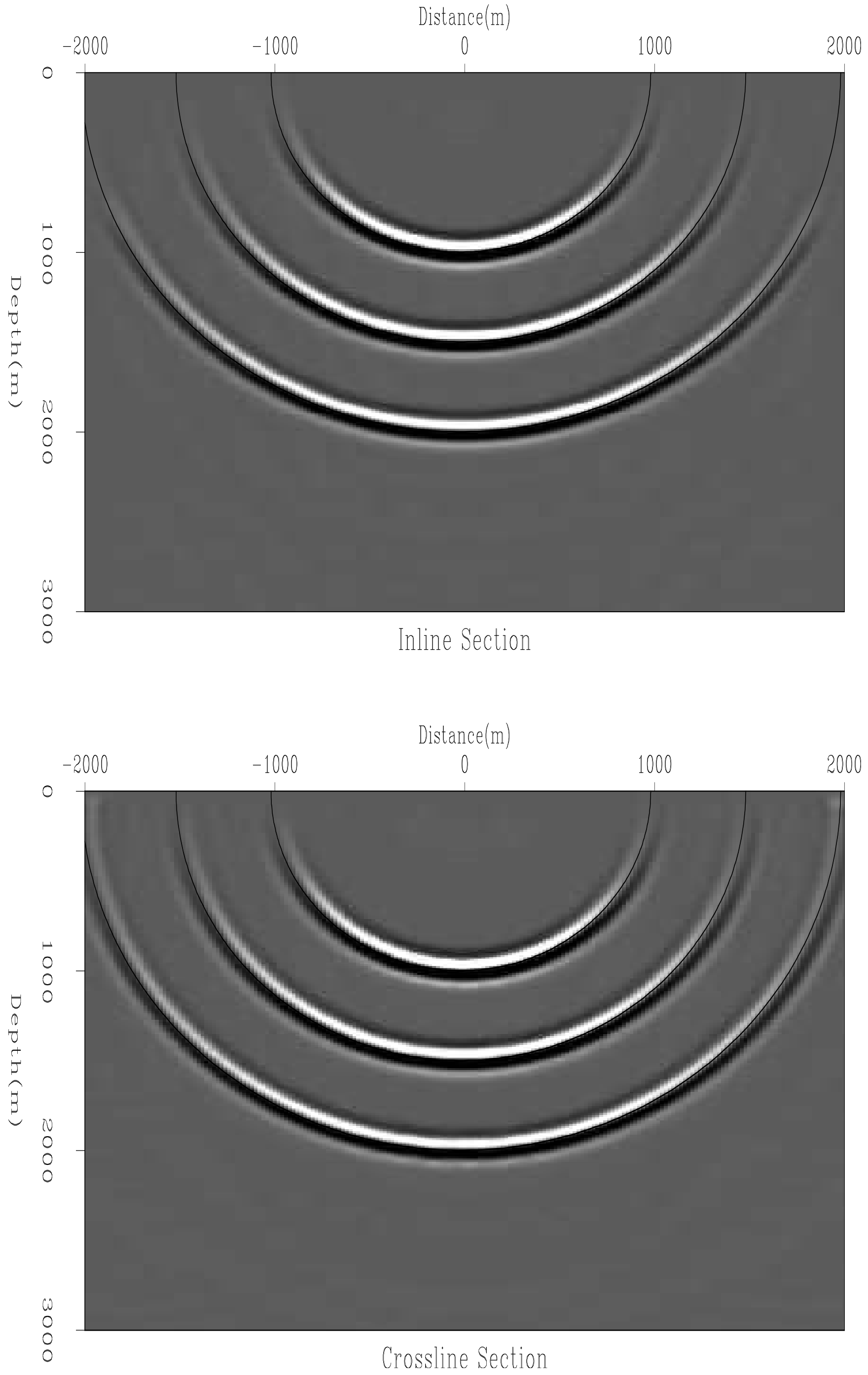


Figure 3: Elliptical-cylindrical-coordinate impulse-response tests. a) Inline section. b) Crossline section. **CR** jeff1/. CrossIn

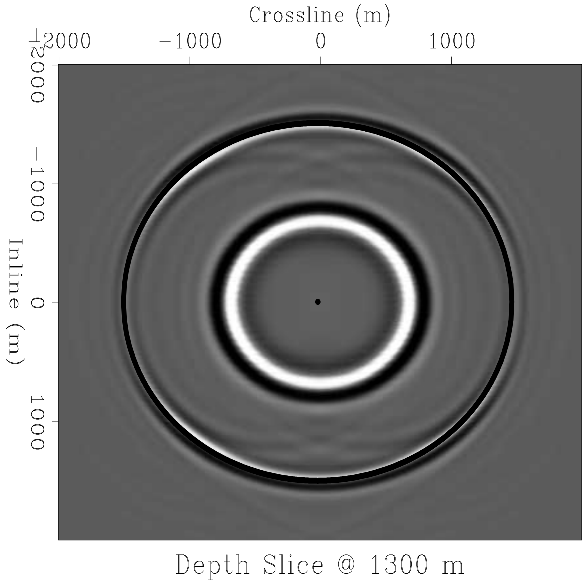


Figure 4: Elliptical-cylindrical impulse response at 1300 m depth. Note the circular symmetry of the impulse response indicating little-to-no numerical anisotropy. **CR** jeff1/. Depth1300

| Source Parameter | Value | Receiver Parameter | Value |
|---|---|---|---|
| Number of sail lines | 72 | Max. inline offset (m) | ±8000 |
| Sail line interval (m) | 250 | Max. crossline offset (m) | ±4000 |
| Shots per sail line | 100 | Inline receiver interval (m) | 50 |
| Shot interval (m) | 250 | Crossline receiver interval (m) | 50 |

Table 2: Parameters associated with the 3D synthetic data set.

I applied the inline delayed-shot migration technique to the plane-wave data on a sail-line by sail-line basis, which allowed for a coarse-grain computational parallelism at a scripting level. (The migration code was also OMP-enabled, which led to a second level of coarse-grain parallelism over the frequency axis.) Migration runs were conducted for Cartesian coordinate (CC) and TEC geometries with both tilting and non-tilting meshes. For CC migrations, the data volumes were zero-padded with 40 samples on each inline side and 95 samples on each crossline side. The data volume for TEC migrations were padded with 40 samples on the inline sides, but only one sample on each crossline side because the coordinate system aperture expands naturally in the crossline direction.

Figure 6 presents the 15400 m cross section from the 24500 m sail-line migration image (for 101 plane-waves) for the TEC (top panel) and the CC (bottom panel) geometries. The gently dipping sedimentary reflections in both sections are imaged across a 6000 m swath. The TEC migration, relative to that in CC geometry, shows a significant improvement in the vertical salt flank on the right-hand-side of the image. Although the salt-flank is weakly present in the CC image under strong clipping, it is mis-positioned due to the 80° limit of extrapolation operator accuracy.

Figures 7 and 8 present crossline sections from the full TEC and CC image volumes. Figure 7 presents the EC and Cartesian crossline sections at the 33700 m inline coordinate in the upper and lower panels, respectively. The TEC image has an improved left-hand salt flank (marked A) that is more correctly positioned relative to the CC image. Similarly, the right-hand salt flank (marked B) is more accurately positioned and forms a more continuous

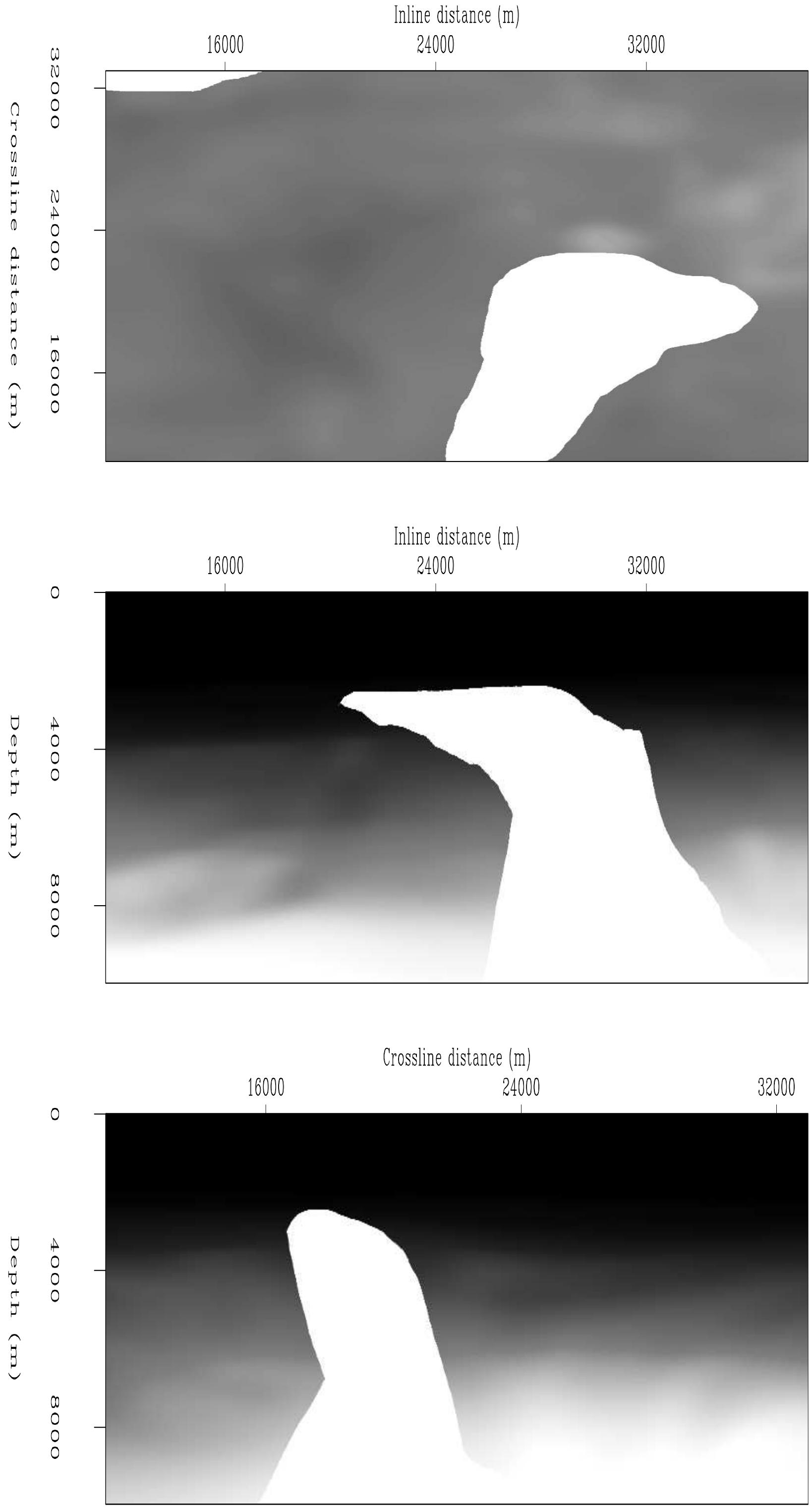


Figure 5: Depth section and inline crossline sections of the Gulf of Mexico velocity model through complex 3D salt bodies. Top: 3900 m depth slice. Middle: 33000 m inline section. Top: 16000 m crossline section. **ER** jeff1/. VELCUT

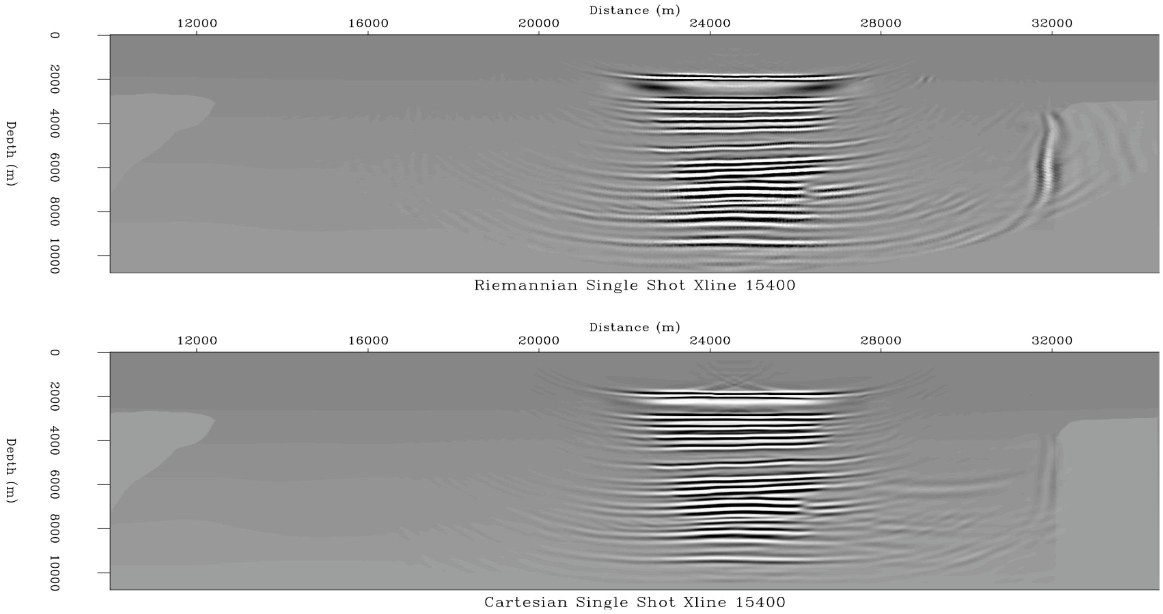


Figure 6: Sections for the 24500 m sail line at the 15400m crossline coordinate. Top: Elliptical-cylindrical coordinate image. Bottom: Cartesian coordinate image. **NR** jeff1/. TESTY15400

reflector. Figure 8 presents the 15100 m crossline sections extracted from the two image volumes. Note the differences in the vertical right-hand salt flank (marked A) between the ECC (top panel) and CC (bottom panel) images. The TEC image exhibits a stronger reflector that is better positioned than that in the CC image (again because of the high-angle limits of the extrapolation operator).

Figure 9 shows the 21750 m inline section through the complete TEC (upper panel) and CC (lower panel) image volumes. The left-hand salt flank (marked A) is more accurately located and continuous in the TEC image. The right-hand salt flank (marked B), again, is more continuous in the TEC image. Another observation is that the TEC image (and in Figures 7-8) does not contain the same spatial frequency content as the CC images (see below).

Figure 10 presents slices extracted at 6150 m depth from the TEC (top panel) and CC (bottom panel) images. The images are again fairly similar, though are there slight differences that correspond to amplitude differences between the weakly imaged steep flank reflectors. Examples include the regions marked A and C that corresponds to the salt flanks in Figure 9 and Figure 7, respectively. Finally, the migration algorithm has well-imaged the set of channels denoted in region B in both coordinate system images.

## Discussion

Relative computational cost is one important metric to consider when comparing the migration algorithms in different coordinate systems. In the above tests, padding in the crossline direction tended to be the most important factor in determining the migration run time.

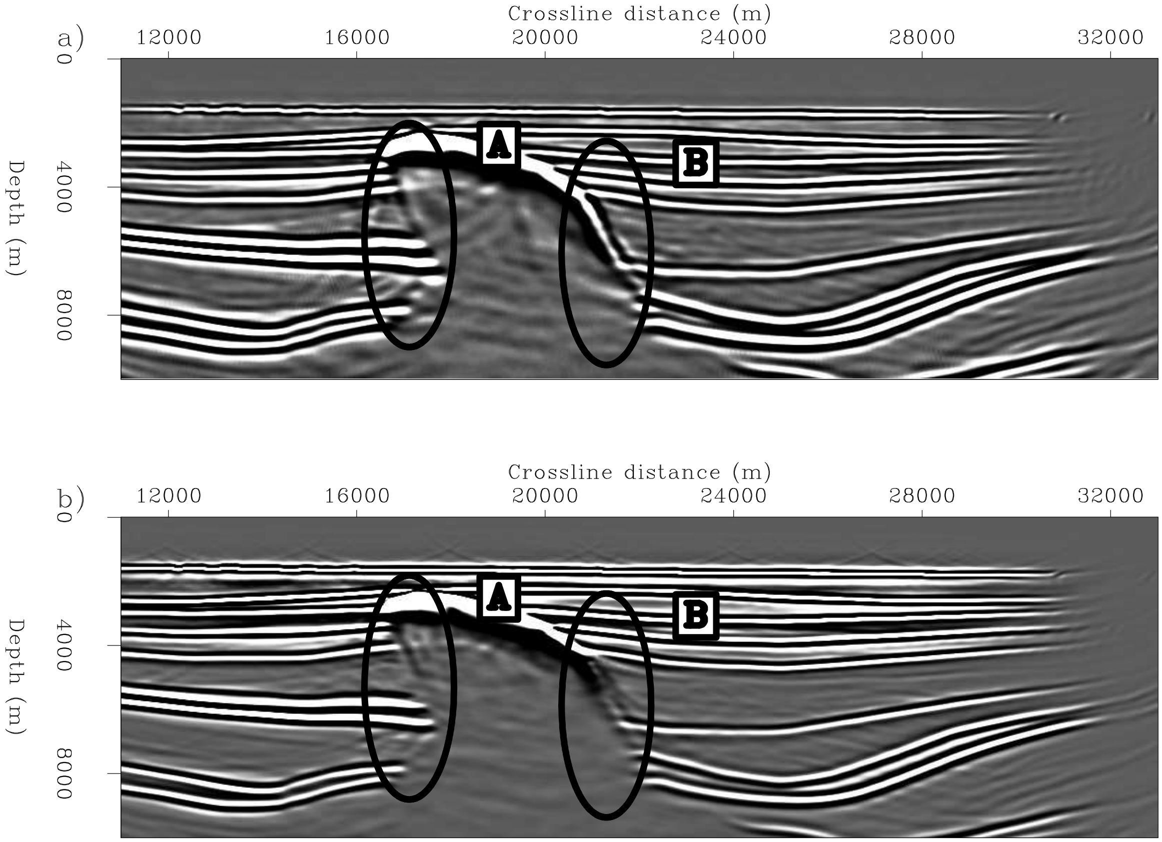


Figure 7: crossline sections through the velocity model and full image volumes at inline coordinate 33700 m. Top panel: Elliptical-cylindrical coordinate image. Bottom panel: Cartesian coordinate image. The imaging improvements for the left-hand salt flank are denoted by the oval marked A. The oval marked B illustrates a more continuous and correctly placed reflector in the TEC coordinate system. **NR** jeff1/. FIG3

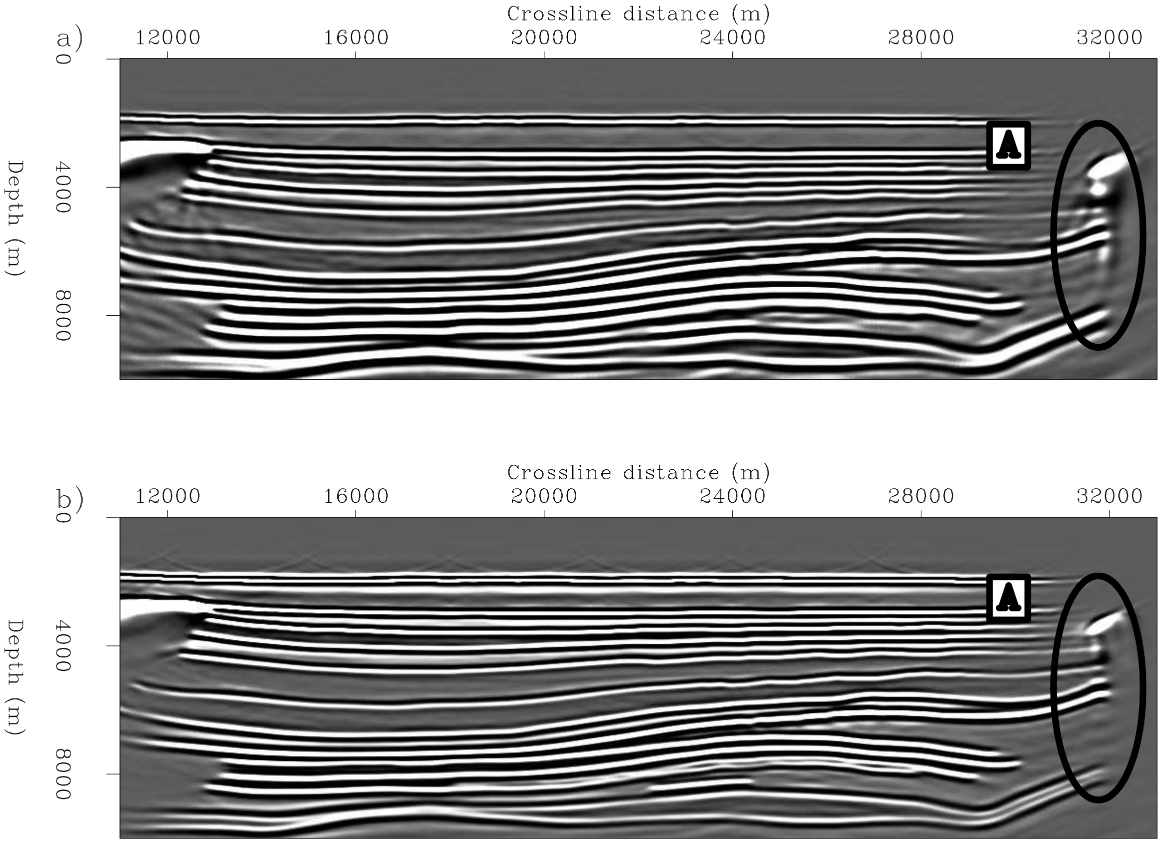


Figure 8: crossline sections through the velocity model and full image volumes at inline coordinate 15100 m. Top panel: Elliptical-cylindrical coordinate image. Bottom panel: Cartesian coordinate image. The oval marked A indicates the location of the vertical salt flank that is better imaged in TEC coordinates. **NR** jeff1/. FIG4

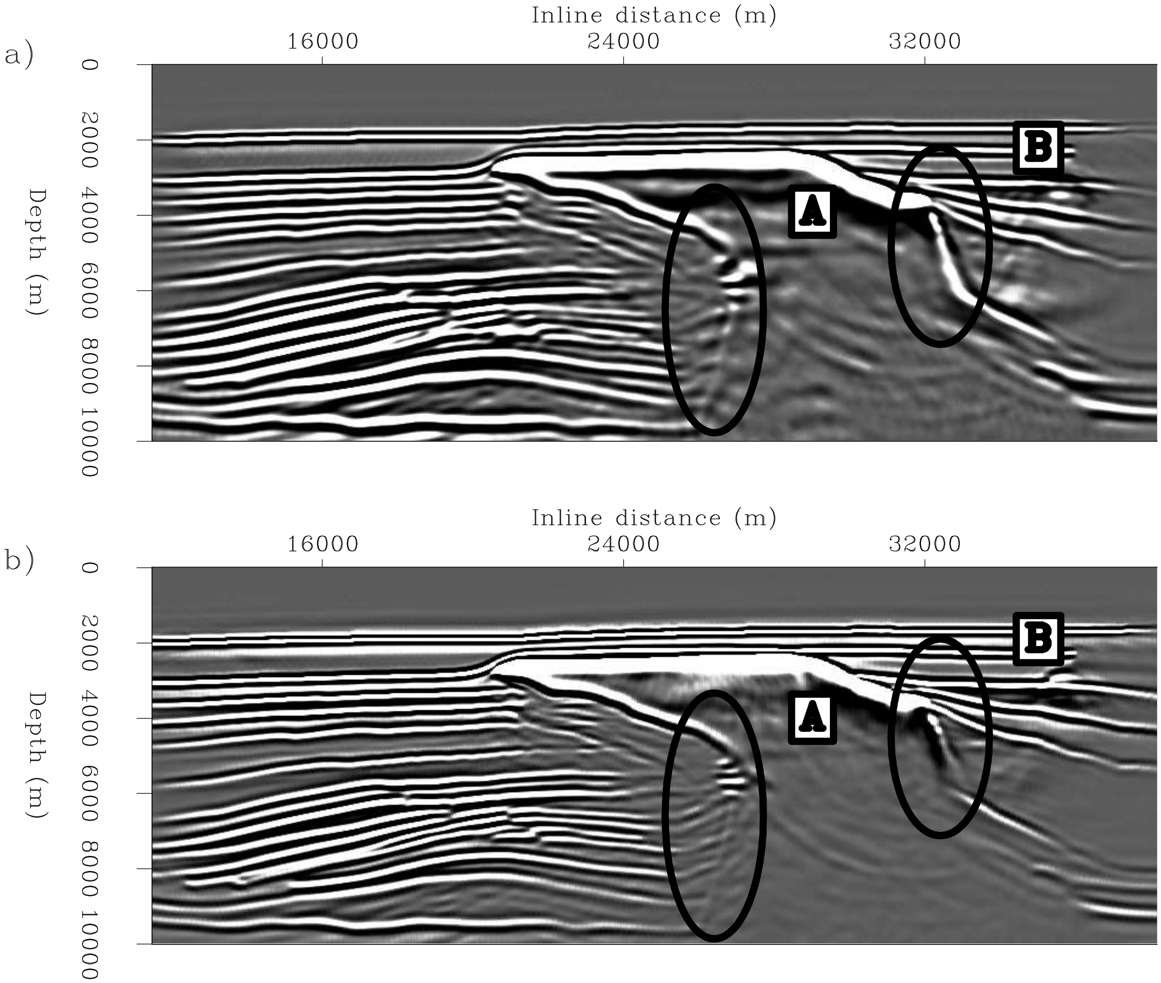


Figure 9: Inline sections through the velocity model and full image volumes at crossline coordinate 21750 m. Top panel: Velocity section. Middle panel: Elliptical-cylindrical coordinate image. Bottom panel: Cartesian coordinate image. The left-hand salt flank, shown in oval A, is more accurately positioned in the TEC coordinate image, while the right-hand flank, marked by oval B, is similarly more accurately positioned and continuous. **NR** jeff1/. FIG1

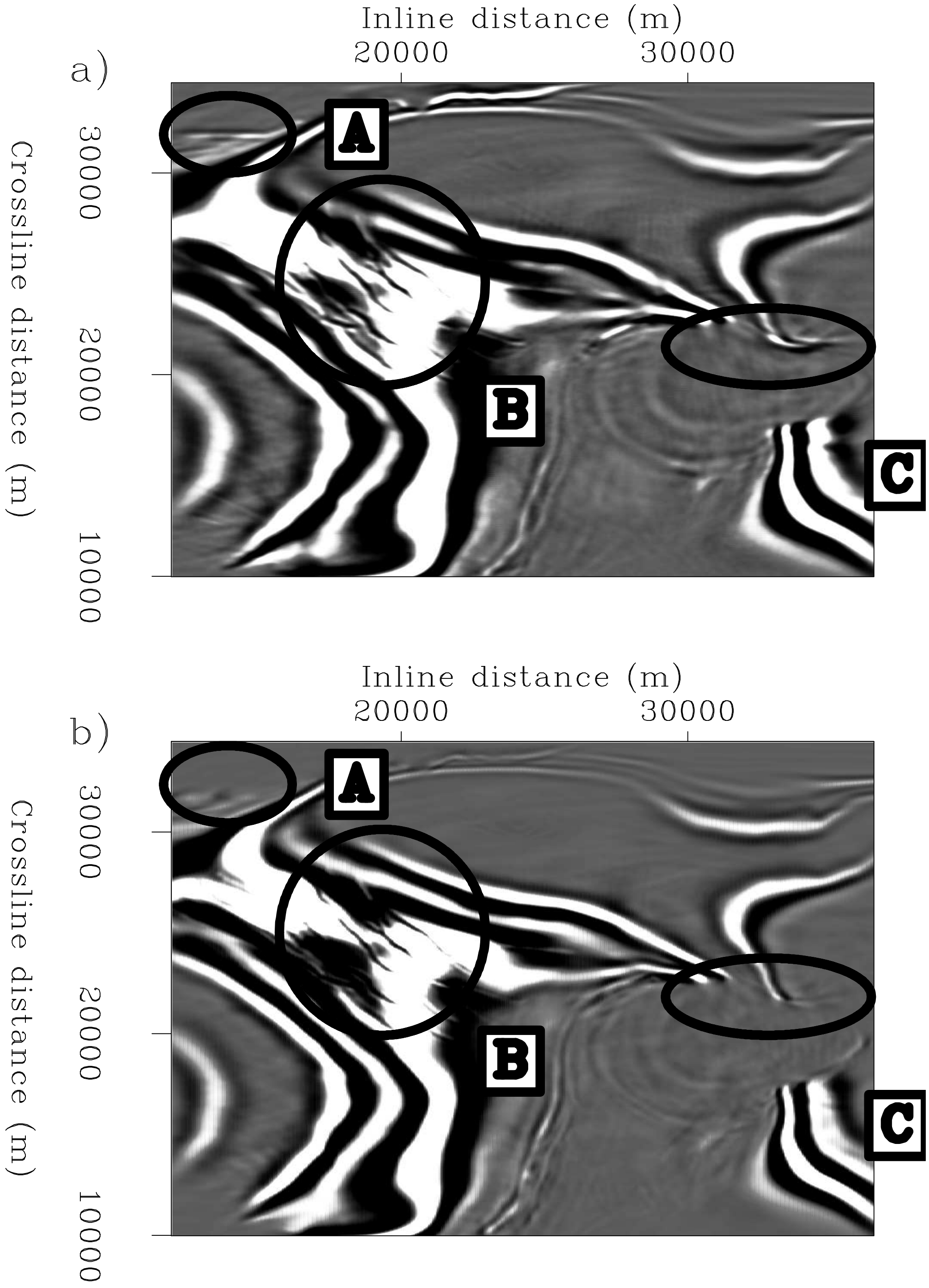


Figure 10: Depth slices through the velocity model and image volumes at 6150 m depth. Top panel: Elliptical-cylindrical coordinate image. Bottom panel: Cartesian coordinate image. Oval A illustrates the improved TEC image for the vertical salt flank shown in Figure 8. Oval B demarcates a region where some of the smaller-scale fractures are well imaged in both images. Oval C shows the region where the near-vertical flank shown in TEC coordinate image in Figure 7 is better imaged. **NR** jeff1/. FIG5

| Migration<br>style | Coordinate<br>system | Mean run<br>time (hrs) |
|---|---|---|
| Plane-wave | Tilted elliptical cylindrical | 37.2 |
| Plane-wave | Cartesian | 45.0 |
| Shot-profile | Tilted elliptical cylindrical | 15.5 |
| Shot-profile | Cartesian | 20.0 |

Table 3: Run-time comparisons for the delayed-shot migration and shot-profile styles in the tilted elliptical-cylindrical and Cartesian coordinate systems.

One benefit of the TEC geometry is its naturally outward-expanding mesh in the elliptical direction that effectively increasing the migration aperture. Thus, TEC migrations usually require less zero-padding in the crossline direction relative to CC geometries. I performed the TEC migrations on meshes with inline-by-crossline-by-depth grids of 720x324x400. Migrations in CC geometries required a 720x512x400 mesh in order to achieve similar crossline aperture, which resulted in a fairly significant additional computational overhead.

Table 3 shows the comparative costs for various TEC and CC migration runs for both the shot-profile and delayed-shot migration styles. I used 72 data points in specifying each median runtime times for the four different migration runs. The test migrations indicate that the TEC geometry migrations were faster than the those in Cartesian tests (for equivalent effective aperture), with 29% and 21% computational cost reduction for the shot-profile and inline delayed-shot migration strategies, respectively.

One question worth addressing is how far can the TEC sampling be reduced before imaging artifacts become apparent? As one moves outward between successive extrapolation surfaces, the TEC geometry expands at increasingly larger step sizes. Fortunately, most realistic velocity models have velocity increasing with depth, causing the wavelengths of the propagated waves to lengthen. This phenomenon acts as a natural wavefield filter that, in most cases, prevents wavenumbers from aliasing (except near-surface in the grid extremities). A good rule-of-thumb is that one must ensure that the grid point of TEC coordinate system mesh does not go below one grid point for every two CC grid points in each direction; however, maintaining this relationship throughout the image volume is not a straightforward task. Additional work on the craft of 3D coordinate-system interpolation is necessary and would likely help restore some of the absent high frequency information.

An additional consideration of parameter choice is the interpolation window over which the surface wavefields are injected onto the TEC coordinate mesh. Not using a sinc-based interpolation over the near-surface depth axis can lead to significant artifacts; however, choosing too large of a window will blend information from different extrapolation steps leading to smoother and lower frequency images. Figures 7-10 show the result of a somewhat overcautious parameter choice (interpolating wavefields three additional depth steps) that led to the lower spatial wavenumber content of the TEC images relative to the CC images. I assert that his effective low-pass filtering can be reduced by interpolating only one or two additional steps in depth.

## NARROW-AZIMUTH FIELD DATA TEST

This section presents the results of applying the inline delayed-shot imaging procedure to a 3D Gulf of Mexico narrow-azimuth data set provided by ExxonMobil. The velocity model, shown in Figure 11, consists of typical sediment-controlled $v(\xi_3)$ velocity structure, save for the salt body intruding in the center of the block, and offset associated with throw along the moderately dipping fault planes. By agreement with ExxonMobil, the depths shown in all figures differ from the true values. Key imaging targets in this model include the steep salt flanks around the salt structure and the onlapping sediments. Previous imaging work in this area indicates that the sediments surrounding the salt body exhibit moderate-to-strong degrees of anisotropy. Bear et al. (2005) estimated the vertical velocity and anisotropy parameters (assuming VTI media) using a joint inversion technique that combined surface seismic and borehole constraints. Shan (2008), using a 3D tilted Cartesian coordinate plane-wave migration algorithm for transversely isotropic (TTI) media, demonstrated that accounting for anisotropy greatly improves migrated image quality for this data set.

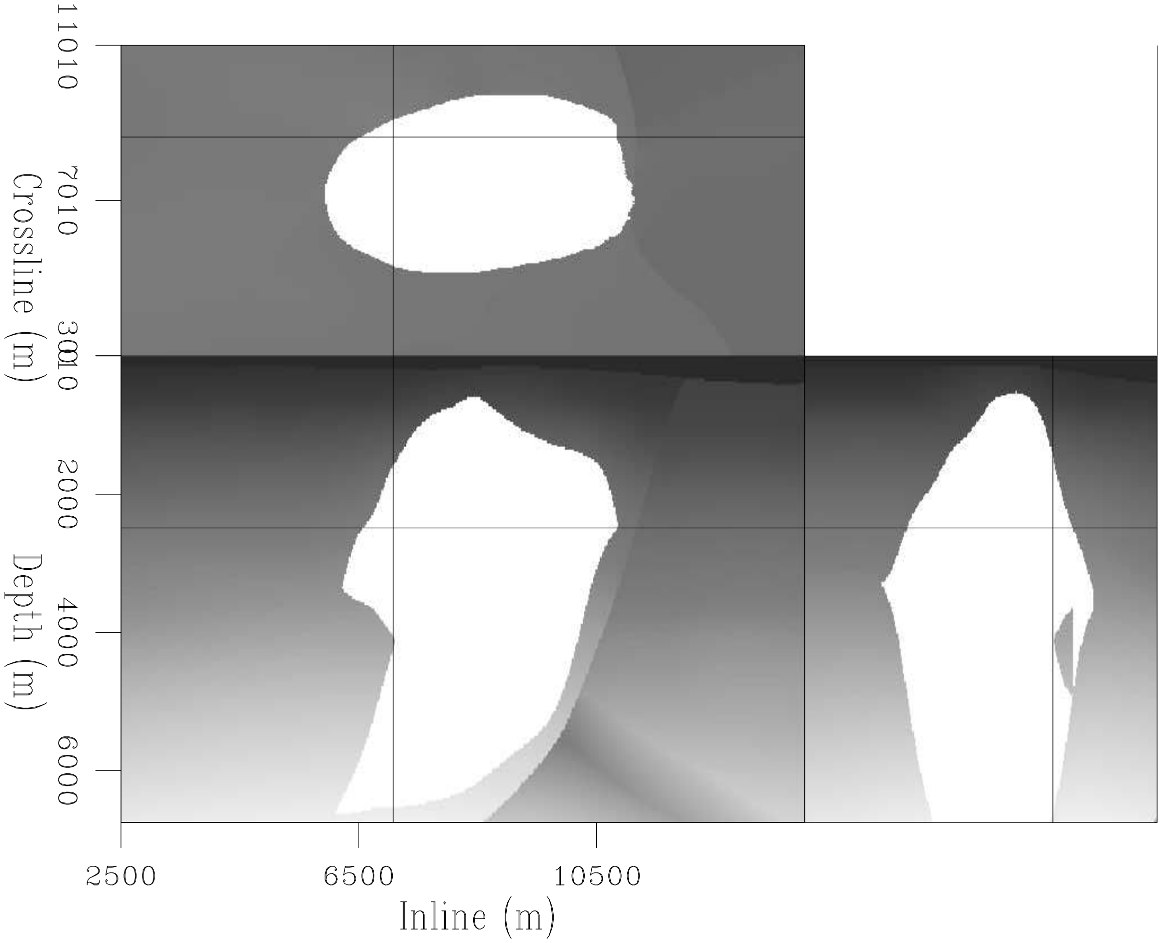


Figure 11: Velocity model example for Gulf of Mexico field data set. **ER** jeff1/. XOM-VEL

The migration strategy presented herein differs from that in Shan (2008) in a number of respects. First, I perform migration using only isotropic vertical-velocity sediment flood model that does not incorporate anisotropy. Second, I use a multi-streamer data set for

| Source Parameter | Value | Receiver Parameter | Value |
|---|---|---|---|
| Number of sail lines | 54 | Max. inline offset (m) | ±3725 |
| Sail line interval (m) | 160 | Max. crossline offset (m) | ±500 |
| Shots per sail line | 300 | Inline receiver interval (m) | 25 |
| Shot interval (m) | 50 | Crossline receiver interval (m) | 80 |

Table 4: Approximate acquisition parameters associated with the 3D Gulf of Mexico field data set.

imaging, rather than the more optimally regularized single-streamer data formed through azimuthal move-out preprocessing (Biondi, 2004).

Table 4 summarizes the acquisition geometry of the data set. The data used for migration consisted of 54 sail lines separated roughly 160 m apart, each sail line consists of approximately 300 shots acquired every 50 m. I binned the sources in 25 m and 80 m intervals in the inline and crossline, respectively. Figure 12 shows the source distribution, and illustrates the sail line direction, herein chosen to be the inline direction. Figure 13 shows the chosen offset distribution. The receiver points fall to both positive and negative offsets, as the sail lines were acquired in two directions. The gap in offset coverage between offsets of ±2500-2750 m arises due to a corrupted data tape. Receivers were binned at 25 m in both the inline and crossline directions.

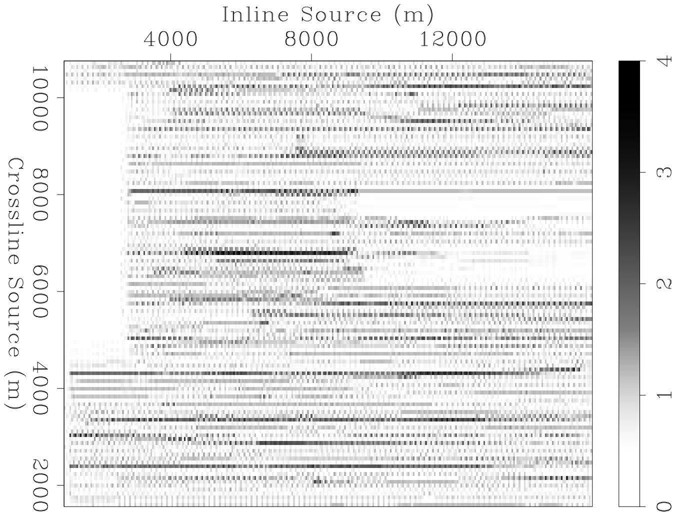


Figure 12: Chosen source distribution for the field data set. **CR** jeff1/. XOM-SRC

I prepared the data for migration by applying an inline delay-shot phase-encoding algorithm according to the inline source position. A total of 54 plane-wave sub-volumes were generated from the total 5D shot record volume, each consisting of 41 plane-waves equally sampled between ±20°. I chose a total of 244 frequencies between 3 Hz and 25 Hz for migration. The data were imaged on migration grids with dimensions of 800x350x300 samples. Migrations in TEC coordinates were performed using tilt angles between ±20° at 1° increments.

Figures 14-16 present comparative slices from the 3D Gulf of Mexico migration images computed in the TEC and Cartesian coordinate systems. Figure 14 presents an inline section taken at the constant 8750 m crossline coordinate for the TEC (top panel) and

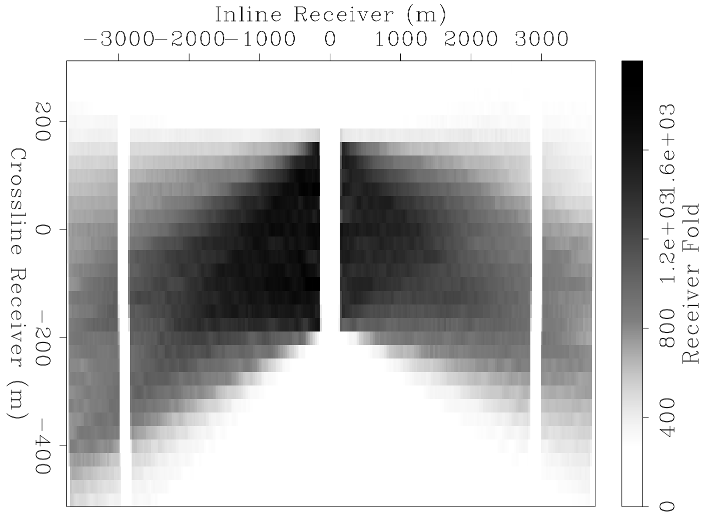


Figure 13: Chosen receiver distribution for the field data set. The missing data between offsets ±2500-2750 m is due to a corrupted data tape. **CR** jeff1/. XOM-RCV

CC (bottom panel) images corresponding to the front face of Figure 11. The top of salt body is well-imaged in both images; however, the near-vertical salt-flanks to the right are nearly entirely absent. Oval A shows the imaging improvements in TEC coordinates for the left-hand flank. Figure 15 presents crossline sections for the TEC (top panel) and CC (bottom panel) images. Both images are subject to fair amounts of near-surface aliasing a sali line contribution every 160 m in the crossline direction, as well as the artifaces due to using a non-regularized data set with locations where no data are present. The outlines of the salt body reflector, though, are imaged. Oval A shots an example of an area where the TEC coordinate image is better than the Cartesian image.

Figure 16 presents a depth slice extracted from the TEC (top panel) and CC (bottom panel) image volumes. The annular ring, showing the location of the salt body, is apparent in both images; however, the image is sharper in the TEC image indicating improved focussing of energy. Oval A shows an example of an area where the TEC image is better than that generated in Cartesian, including two parts of the right-hand salt flank. Oval B shows the TEC coordinate image improvements in the crossline direction.

The results of the 3D field data application likely could have been improved in a number of aspects. First, a migration velocity model incorporating anisotropy values (e.g. $v_{mig} = v_{vert}(1 + 1.8\delta)$) could have been used instead of the vertical velocity profile. Although this would affect the vertical location of the flat-lying sedimentary reflectors, it likely would have led to more accurate horizontal propagation and imaging of waves reflecting off the target salt flanks. Second, if additional computational resources were made available, migrating the full data set (i.e. every 80 m in crossline source position rather than every 160 m) with a higher frequency content would have led to a more infilled and higher resolution image. Third, extending the generalized RWE theory to incorporate TTI anisotropy likely would have enabled a more consistent imaging of the steep salt flanks. This extension is likely to be a subject for future research.

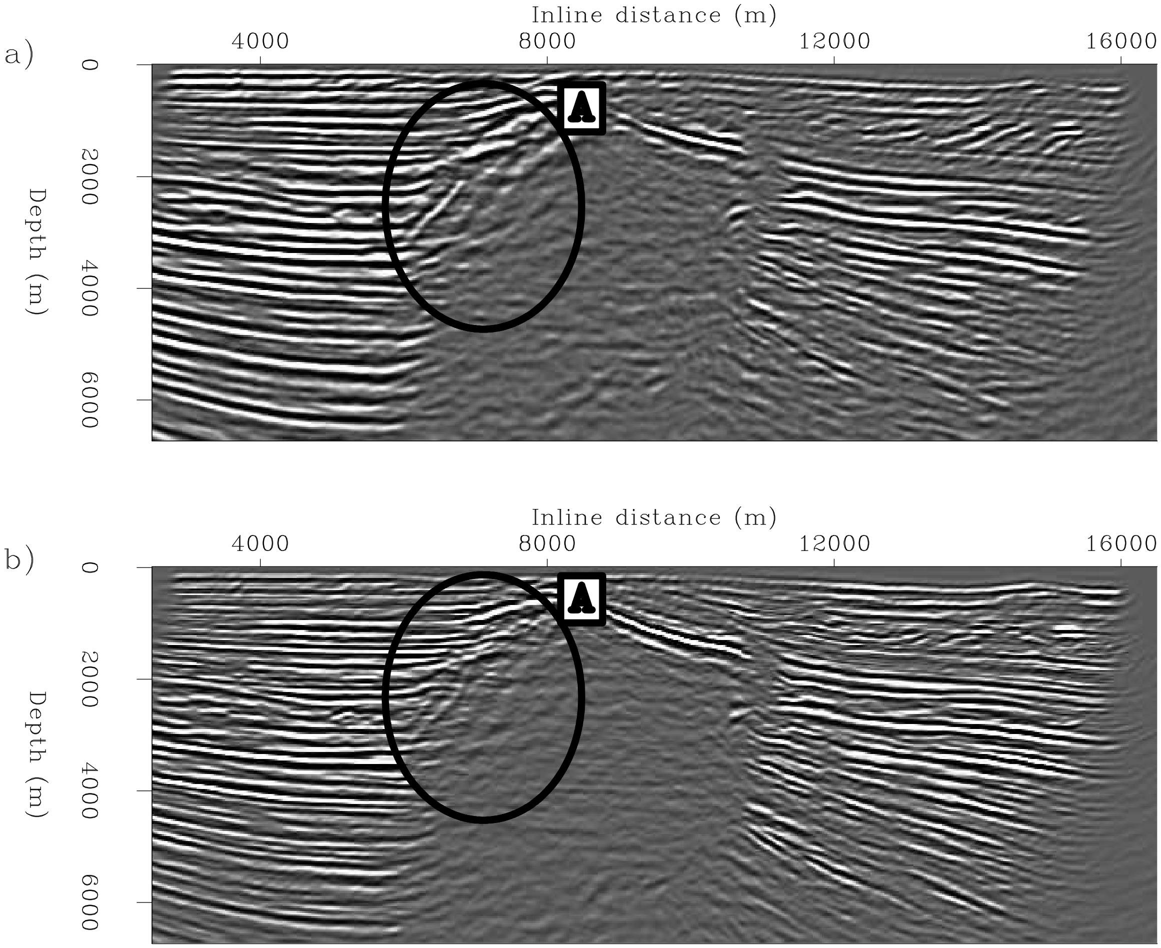


Figure 14: Inline sections through the migration images taken at the 8750 m crossline coordinate location. Top: TEC coordinate migration results. Bottom: Cartesian coordinate migration results. Oval A shows the imaging improvements in TEC coordinates for the left-hand flank. **CR** jeff1/. RFIG1

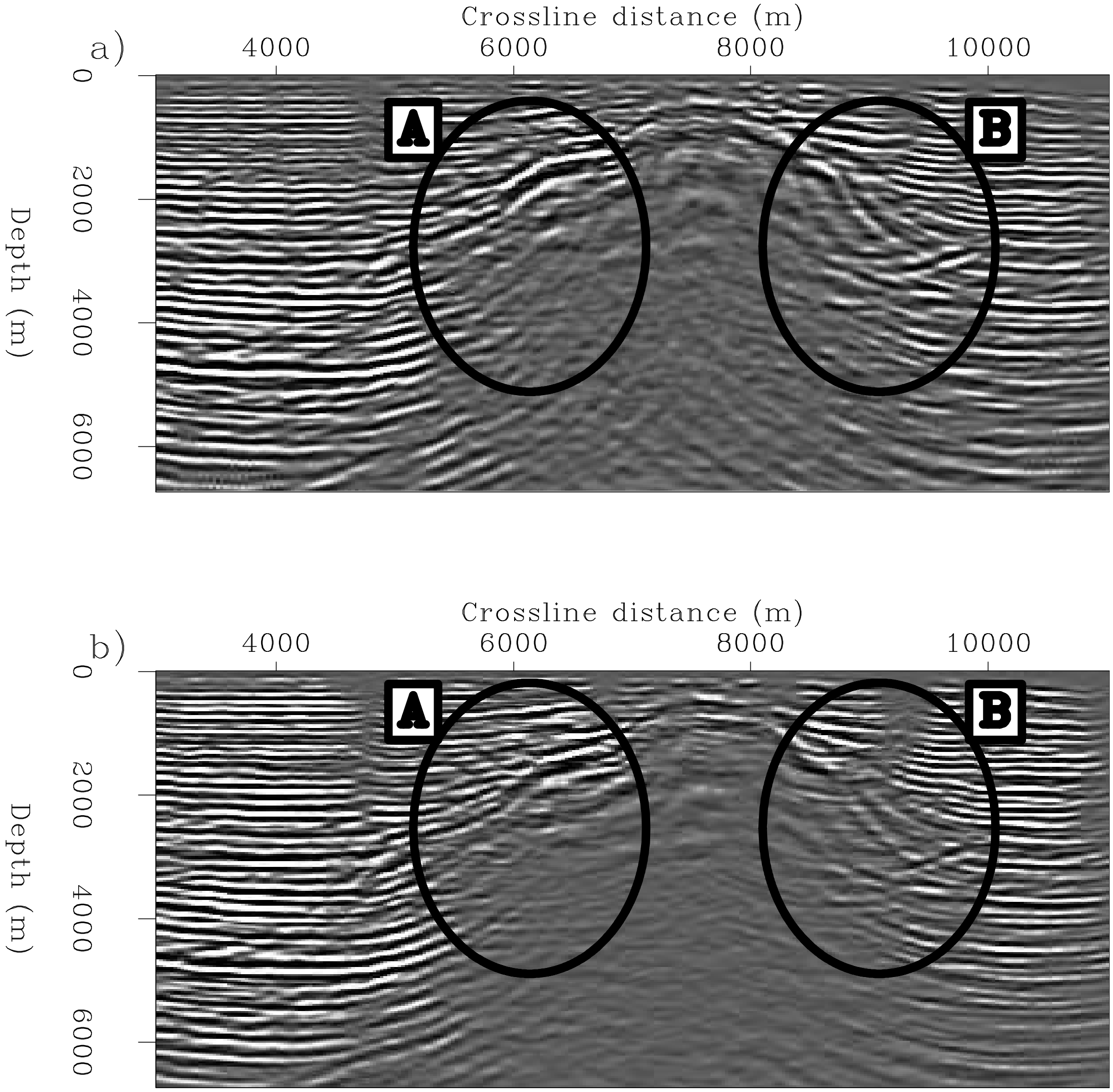


Figure 15: Crossline sections through the migration images taken at the 7100 m crossline coordinate location. Top: TEC coordinate migration results. Bottom: Cartesian coordinate migration results. Oval A shows an example of an area where the TEC coordinate image is better than the Cartesian image. **CR** jeff1/. RFIG3

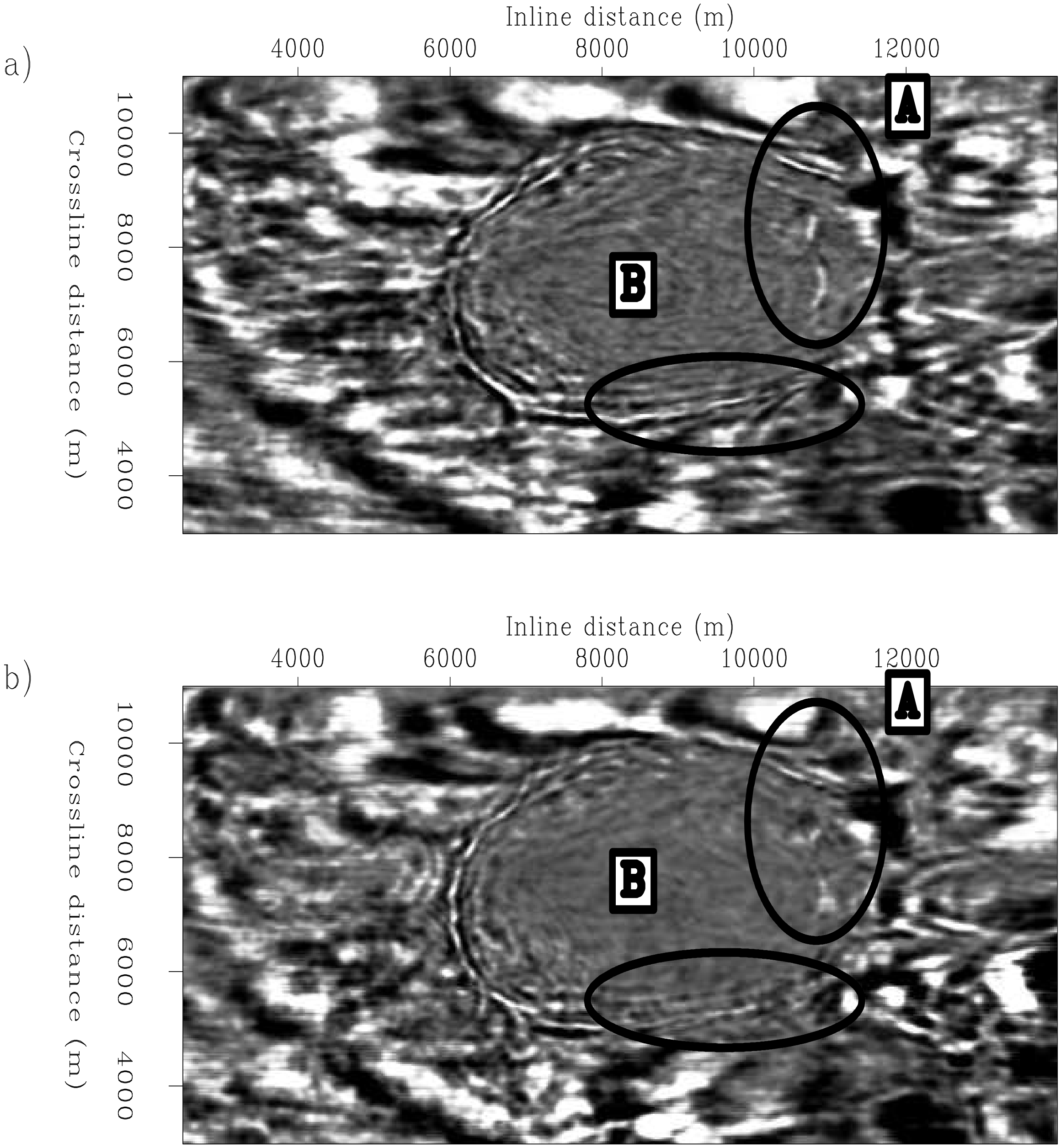


Figure 16: Migration results for the 3D Gulf of Mexico field data set through the sedimentary section. Top: TEC coordinate migration results. Bottom: Cartesian coordinate migration results. Oval A shows an example of an area where the TEC image is better than that generated in Cartesian, including two parts of the right-hand salt flank. Oval B shows the TEC coordinate image improvements in the crossline direction. **CR** jeff1/. RFIG5

## CONCLUSIONS

This paper discusses an inline delayed-shot migration technique in tilted elliptical-cylindrical coordinates. I argue that migration approach, relative to the full 3D plane-wave technique, offers both lower memory requirements (due to small migration aperture), as well as a potential reduction in the number of total migrations needed (by migrating fewer sail lines than crossline plane waves). I demonstrate that the impulse response of inline-source delayed-shot wavefields are well-matched to TEC geometry, and that corresponding extrapolation wavenumber is no more complicated than that of elliptically anisotropic media. This leads to an accurate 3D finite-difference splitting algorithm that both accurately propagates wavefields and handles the associated numerical anisotropy. The 3D synthetic Gulf of Mexico data tests demonstrate the migration technique's ability to generate improved images of steeply dipping structure, relative to Cartesian coordinate migration, at reduced computational cost. Field data tests illustrate the utility of the 3D migration approach in exploration practice.

## ACKNOWLEDGMENTS

I thank Biondi Biondi, Bob Clapp, Alejandro Valenciano, Ben Witten, and Bill Curry for helpful conversations. John Etgen is greatly thanked for donating his wisdom and time on the BP research computer cluster, where the 3D synthetic imaging results were generated. I acknowledge BP for the release of the synthetic data set, and ExxonMobil for releasing the field data set.

## REFERENCES

Arfken, G., 1970, Mathematical methods for physicists, 2nd ed.: Academic Press, New York.

Baumstein, A. and J. Anderson, 2003, Wavefield extrapolation in laterally varying VTI media: 73rd Annual International Meeting, SEG, Expanded Abstracts, 945–948.

Bear, L., T. Dickens, J. Krebs, L. Liu, and P. Traynin, 2005, Integrated velocity model estimation for improved positioning with anisotropic psdm: The Leading Edge, **24**, 622–634.

Biondi, B., 2004, 3-D Seismic Imaging: Stanford University.

Duquet, B. and P. Lailly, 2006, Efficient 3D wave-equation migration using virtual planar sources: Geophysics, **71**, S185–S197.

Duquet, B., P. Laily, and A. Ehinger, 2001, 3D plane wave migration of streamer data: 71st Annual International Meeting, SEG, Expanded Abstracts, 1033–1036.

Jing, X., C. Finn, T. Dicken, and D. Willen, 2000, Encoding multiple shot gathers in prestack migration: 70th Annual International Meeting, SEG, Expanded Abstracts, 786–789.

Lee, M. W. and S. Y. Suh, 1985, Optimization of one-way wave-equations (short note): Geophysics, **50**, 1634–1637.

Li, Z., 1991, Compensating finite-difference errors in 3D migration and modeling: Geophysics, **56**, 1650–1660.

Liu, F., D. Hanson, N. Whitmore, R. Day, and R. Stolt, 2006, Toward a unified analysis for source plane-wave migration: Geophysics, **71**, S129–S139.

Liu, F., H. D. Whitmore, N.D., and R. Day, 2004, The impact of reciprocity on prestack source plane wave migration: 74th Annual International Meeting, SEG, Expanded Abstracts, 1131–1134.

Ma, Z., 1982, Finite-difference migration with higher-order approximation: Oil Geophysical Prospecting, **1**, 6–15.

Morton, S. and C. Ober, 1998, Faster shot record depth migrations using depth encoding: 68th Annual International Meeting, SEG, Expanded Abstracts, 1131–1134.

Mosher, C. and D. Foster, 1998, Offset plane wave propagation in laterally varying media: *in* S. Hassanzadeh, ed. Mathematical methods in Geophysical imaging, 36–46.

Romero, L., D. Ghiglian, C. Ober, and S. Morton, 2000, Phase Encoding of Shot Records in Prestack Migration: Geophysics, **65**, 426–436.

Shan, G., 2008, Image of steep reflectors in anisotropic media by wavefield extrapolation: Ph.D. thesis, Stanford University.

Shan, G. and B. Biondi, 2004, Imaging overturned waves by plane-wave migration in tilted coordinates: 74th Annual International Meeting, SEG, Expanded Abstracts, 969–972.

———, 2005, 3D wavefield extrapolation in laterally-varying tilted TI media: 75th Annual International Meeting, SEG, Expanded Abstracts, 104–107.

Shragge, J., 2008, Riemannian wavefield extrapolation: Non-orthogonal coordinate systems: Geophysics, **73**, T11–T21.

Shragge, J. and G. Shan, 2008, Prestack wave-equation migration in elliptical coordinates: Geophysics, **73**, S169–S175.

Sun, P., S. Zhang, and F. Liu, 2002, Prestack migration of areal shot records with phase encoding: 72nd Annual International Meeting, SEG, Expanded Abstracts, 1172–1175.

Thomsen, L., 1986, Weak elastic anisotropy: Geophysics, **51**, 1954–1966.

Tsvankin, I., 1996, P-wave signatures and notation for transversely isotropic media: An overview: Geophysics, **61**, 467–483.

Whitmore, N. D., 1995, An imaging hierarchy for common angle plane wave seismograms: Ph.D. thesis, University of Tulsa.

Zhang, J., D. Verschuur, and C. A. P. Wapenaar, 2001, Depth migration of shot records in heterogeneous, transversely isotropic media using optimum explicit operators: Geophysical Prospecting, **49**, 287–299.

Zhang, Y., C. Notfors, S. Gray, L. Chernis, and J. Young, 2003, Delayed-shot 3D prestack depth migration: 73rd Annual International Meeting, SEG, Expanded Abstracts, 1027–1030.

*Stanford Exploration Project, Report SEP138, October 19, 2009*

# Reverse time migration with random boundaries

*Robert G. Clapp*

## ABSTRACT

Reading wavefield checkpoints from disk is quickly becoming the bottleneck in Reverse Time Migration. We eliminate the need to write the wavefields to disk by creating an increasingly random boundary around the computational domain when propagating the source function. The wavefield that encounters the boundary region is pseudo-randomized. Reflections off the random layer have minimal coherent correlation with the receiver wavefield but can be reformed by running the wave equation in the reverse direction. This allows the source to first be propagated to the maximum recording time and then to be backward propagated simultaneously with receiver wavefield significantly reducing memory and IO requirements. I demonstrate the methodology on the Sigsbee synthetic and show that it significantly reduces coherent correlation artifacts.

## INTRODUCTION

Reverse Time Migration (RTM) (Baysal et al., 1983) is quickly becoming the standard for high-end imaging. At the core of the RTM algorithm is a modeling kernel. The simplicity of the the modeling kernel has led to high-performance implementation on Field Programmable Gate Arrays (FPGA) (Nemeth et al., 2008), General Purpose Graphics Processing Units (GPGPU) (Micikevicius, 2008), and conventional processors. Of growing significance is the problem that the source field must propagated from $t = 0$ to $t =$ maxtime while the receiver wavefield must be propagated from $t =$ maxtime to $t = 0$ since the fields must be correlated at equivalent time positions. One propagation must be stored either completely or in a check-pointed manner to disk.

Symes (2007) and Dussaud et al. (2008) discuss checkpointing methods to handle the different propagation directions. Dussaud et al. (2008) and Clapp (2008) suggest an alternate approach of propagating the source wavefield to the maximum recording time and then reversing the propagation to make it consistent with the receiver propagation direction. The use of damping schemes around the boundary results in the need to inject energy from the non-damped, forward propagated wavefield, inside the boundary region. The RAM requirement with this scheme is less than conventional checkpointing approaches but still imposes significant disk IO requirements.

In this paper, I discuss an alternate approach. I replace the conventional damped region with an increasingly random velocity region. Rather than eliminate reflections I distort them to minimize coherent correlations with the receiver wavefield that cause artifacts. I begin by describing the construction of the random boundary. I then demonstrate the amplitude behavior of the time reversed wavefield. I conclude by showing the methodology applied in a 2-D synthetic.

## BACKGROUND

The basic idea behind RTM is to propagate a source function within the computational domain from time $t = 0$ to $t =$ maxtime, storing the wavefield $s(t, \mathbf{x})$ at time steps consistent with the time sampling of the data $dt$. The recorded data is injected into a second computational domain and propagated from $t =$ maxtime to $t = 0$ and stored in $r(t, \mathbf{x})$. The final image $I$ is constructed

$$I(\mathbf{x}) = \sum_{t=0}^{\text{maxtime}} s(t, \mathbf{x}) r(t, \mathbf{x}), \tag{1}$$

or some similar imaging condition. The problem is that $s(t, \mathbf{x})$ and $r(t, \mathbf{x})$ are too large to store in memory, and often too large store on disk. The image can be updated while propagating the receiver wavefield but the different propagation directions of the source and receiver wavefields, still introduce a large storage requirement. To reduce the storage requirement, checkpointing schemes are used. The source wavefield is stored at various intervals $d_{check}$ during forward propagation. When propagating the receiver wavefield these checkpoints are read from disk and re-propagated into a buffer to be correlated with the receiver propagation. Algorithm 1 illustrates this approach. There are several undesirable features to this approach. First, the source wavefield must be re-propagated. Second, buffering of the re-propagated source wavefield introduces a large memory requirement. Finally, reading the snapshots from disk, particularly when the propagation is done on an accelerator, can/does make RTM IO bound. A wavefield can be propagated forward or backwards in time. On first glance, an obvious solution to the storage and IO requirements is to propagate the source wavefield to $t =$ maxtime and then reverse the propagation eliminating the need for checkpointing. The flaw in this hypothesis comes from the boundary conditions we conventionally use.

**Algorithm 1** Standard RTM with checkpointing

```
for all shots do
  while $t <$ maxtime do
    Advance source wavefield to check point ($d_{check}$)
    store wavefield
  end while
  $t =$ maxtime
  while $t > 0$ do
    Read source wavefield at $t - d_{check}$
    Advance and buffer source wavefield to $t$
    for $t = 0$ to $t = d_{check}$ do
      Advance receiver wavefield $-dt$
      Correlate source and receiver wavefield at constant $t$
    end for
    $t = t - d_{check}$
  end while
end for
```

## BOUNDARY CONDITIONS

Ideally, we would like to emulate an infinite computational domain when propagating our seismic wavefield. Computationally, this is an unachievable goal; instead we try to minimize artifacts, reflections from the boundaries, caused by our limited computational domain. The most effective method is the Perfectly Matched Boundary method from the electromagentic community (Berenger, 1994). This method amounts to mapping the coordinate system into the complex domain and changing the propagating wavefield into a decaying wavefield. A second common technique is to introduce a damping region at the edge of the computation domain (Cerjan et al., 1985; Baysal et al., 1984). This is often combined with techniques to kill plane waves that are perpendicular to the computational boundary. All of these techniques have proven effective for modeling seismic data, but force propagation in a single direction.

One technique to allow time reversal is to store the wavefield that has not hit the boundary region at each time step and then re-inject when reversing the propagation direction (Clapp, 2008). This technique has the advantage of eliminating the need for buffering, but still requires a large volume of data to be stored on disk.

Another approach is to rethink what we are attempting to accomplish with our boundary conditions for RTM. Ideally, we would like to eliminate all reflections from the edge of our computational domain, but what we are really concerned with is *coherent* reflections. If we can distort the wavefield coming from the boundaries so it does not coherently correlate with the receiver wavefield, we will have accomplished our goal.

In acoustic modeling, one way to manipulate the boundaries while still allowing time reversal, is to introduce a random component to the velocity field at the boundary. We must be careful in how we modify the boundary, both by staying within the stability constraint of our finite difference method, and by slowly introducing random numbers to avoid immediate reflections off the randomized zone. The basic algorithm for constructing the boundary can be found below. Panel 'A' of Figure 1 shows a velocity model constructed with a random boundary and panel 'B' shows a cross section through that boundary. Note how the variability of the velocity increases near the edge of the computational domain.

**Algorithm 2** Creating random boundary

```
for all x,y,z do
  if within boundary region then
    $d$=distance within boundary
    found=false
    while found==false do
      select random number $r$
      $vtest = v(x,y,z) + r * d$
      if $vtest$ meets stability constraint then
        $v(x,y,z) = vtest$
        found=true
      end if
    end while
  end if
end for
```

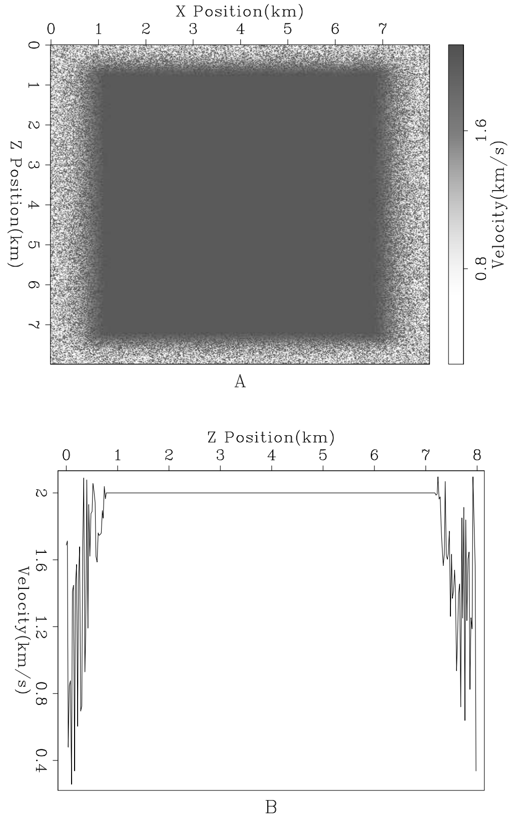


Figure 1: Panel A shows the constant velocity model with a random boundary. Panel B shows a cross section through the velocity model. Note the increasing random nature of the boundary. [**ER**] bob1/. vel

Panel ‘A’ of Figure 2 shows the wavefield at $t = .63$ after injecting a source in the center of the computational domain. Panel ‘B’ shows the wavefield at $t = 2.205$ after the wavefield has hit the boundary. Note the absence of a coherent reflection off the boundary. Panel ‘C’ shows the wavefield at $t = 3.906$, when no coherent energy is present. The wavefield was then propagated to $t = 5.0$ and the computation was reversed. Panel ‘D’ shows the reversed wavefield at $t = .63$, and panel ‘E’ shows the difference between wavefields scaled by 10,000. The difference between the two images is in the range of the machine precision.

To see if there is a coherent pattern underneath the random looking field, I repeated the experiment 16 times, each with a different random boundary. Panels ‘F-J’ show the average of these 16 experiments at the same times as panels ‘A-E’. Note in panel ‘G’ how the average of the 16 experiments has greatly reduced the energy reflected from the boundary. In panel ‘H’ we can see that there is a low energy, low spatial frequency reflection from the boundary. Panel ‘J’ shows that even the machine precision noise tends to cancel.

Panels ‘A’ and ‘B’ of Figure 3 show the results of Fourier transforming the data shown in panels ‘C’ and ‘H’ of Figure 2. Panel ‘A’ demonstrates that the wave field is dominated by low-spatial wave-number features. In panel ‘B’ we see that almost all of the energy is concentrated at the low spatial wave-numbers. This is not surprising. At low enough frequencies, our random boundary does not affect our propagating wavefield. By increasing the size of the random zone, we can damp lower frequencies.

Summing multiple experiments is directly applicable to RTM. In RTM we are summing the result from many migrated shots to form our final image. As a result we will get a

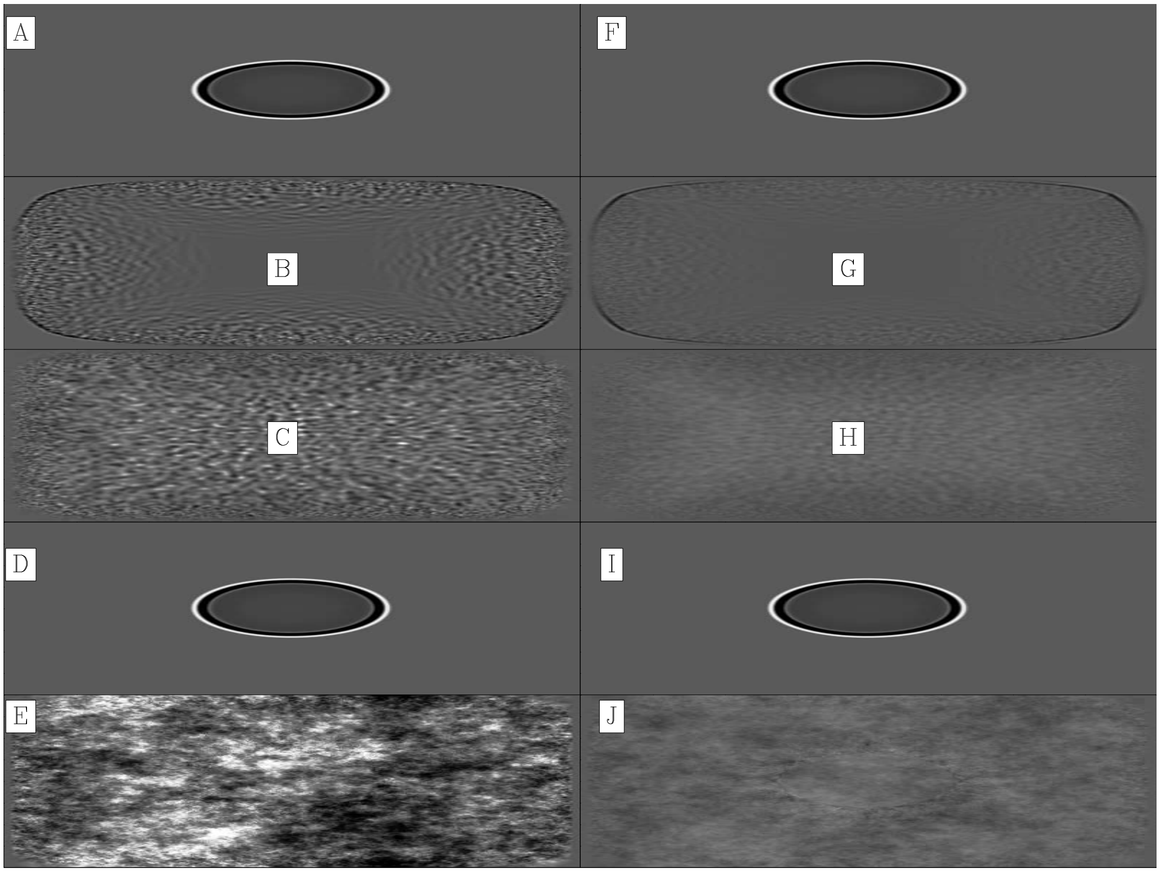


Figure 2: The left column shows snapshots from a single modeling experiment at .63, 2.205, and 3.906. The fourth panel shows the result of time reversing the computation to .63 seconds. The bottom panel shows the difference between first and fourth panel scaled by 10000. The right column shows the summing of 16 modeling experiments each with different random boundaries. [**ER**] bob1/. combo

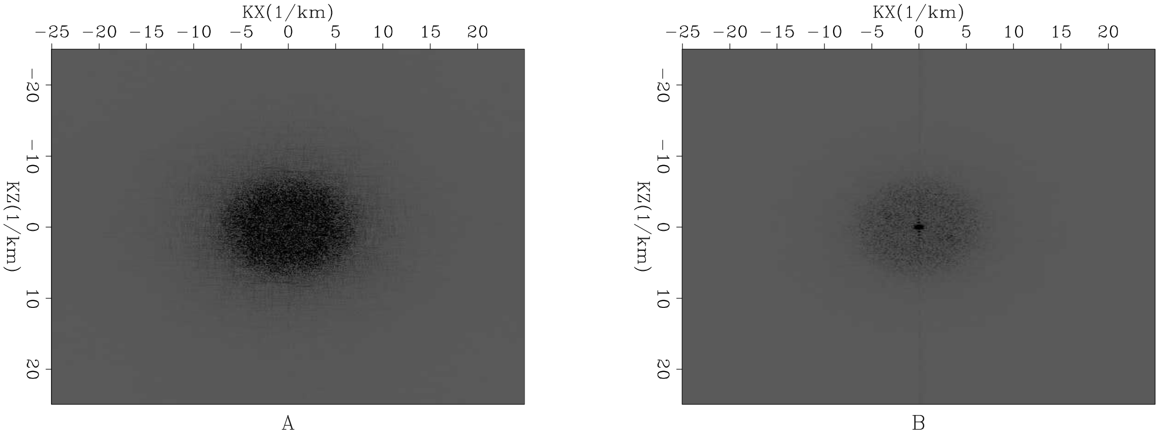


Figure 3: Panel 'A' is the result of Fourier transforming the data shown in panels 'C' and 'H' of Figure 2. Note how the summing of multiple experiment reduces the energy in higher spatial wave-number events. [**ER**] bob1/. spat

signal-to-noise boost from neighboring shots having different random patterns.

By making these changes, the RTM algorithm simplifies to the following template.

**Algorithm 3** Time reverse RTM with random boundaries

```
for all shots do
  Create random boundary around computational domain
  Advance source wavefield to $t$ = maxtime
  for $t$ = maxtime to $t = 0$ do
    Advance source wavefield $-dt$
    Advance receiver wavefield $-dt$
    Correlate source and receiver wavefield
  end for
end for
```

## SYNTHETIC EXAMPLE

To test the methodology, I implemented both algorithm 1 and 3. I used the first 4.5 seconds of the Sigsbee synthetic, limiting the computational domain to the first 4.5 km. As a baseline I performed RTM using a reflecting boundary condition. Locations 'A' and 'B' in Figure 4 show where reflections from the top of the domain coherently correlated with the receiver wavefield.

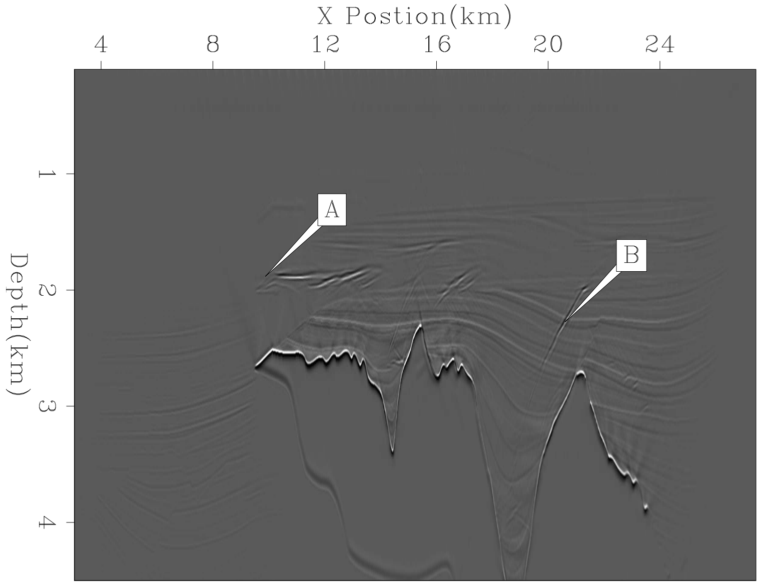


Figure 4: The result of RTM migration using a reflecting boundary. Note the obvious boundary artifacts at 'A' and 'B'. [**CR**] bob1/. none

Figure 5 shows the RTM result using algorithm 1. In this case I applied a damped exponential in a 30 point region around the computational domain. Note how the artifacts at 'A' and 'B' are significantly reduced. Finally, Figure 6 shows the result of algorithm 3. Note how the energy at 'A' and 'B' are further reduced. Some additional coherent correlations are present in the water layer but are not visible lower in the image once sufficient reflected energy is present.

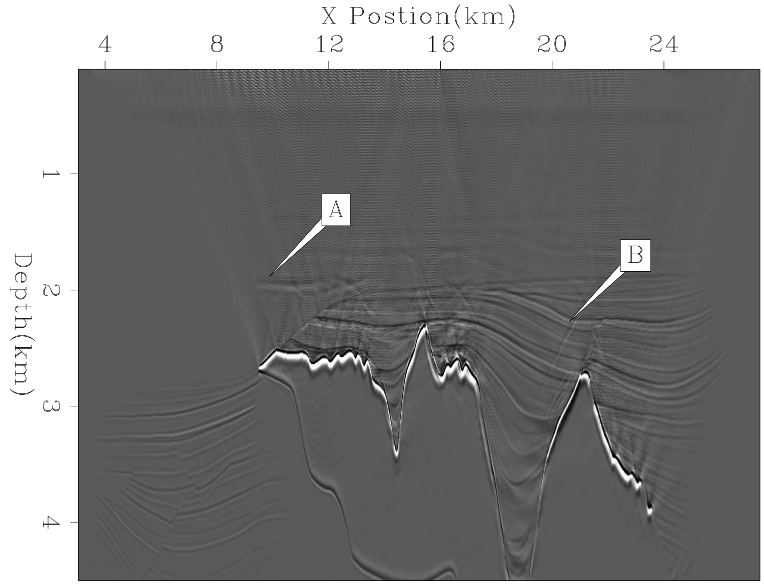


Figure 5: The result of RTM migration using a damping boundary condition following the methodology described by algorithm 1. Note how the boundary artifacts have been reduced at ‘A’ and ‘B’. [**CR**] bob1/. damp

## OTHER APPLICATIONS

There are two notable additional applications to this approach. We can think of the random boundary region as a series of point sources that are excited at different times. By cross-correlating the energy at any two locations we can generate a two-way interferometric Green’s function.

In addition, the random boundaries offer the potential to use implicit rather than explicit propagation. We can treat our medium as a single 1-D trace (helixize (Claerbout, 1998) the computational domain). We can then use 1-D theory to calculate causal-minimum phase filters that can be applied using polynomial division. This technique was applied to downward continuation by Rickett et al. (1998) but suffered from obvious wraparound problems. The random boundaries discussed in this paper would minimize this problem.

## CONCLUSIONS

Pseudo-random boundary conditions effectively distort an incoming wavefield. I use these boundary conditions to propagate the source wavefield in RTM both forward and backwards. The distorted wavefield correlates poorly with the receiver wavefield, minimizing boundary artifacts. I hypothesize that these boundaries have additional applications for implicit finite difference and interferometry.

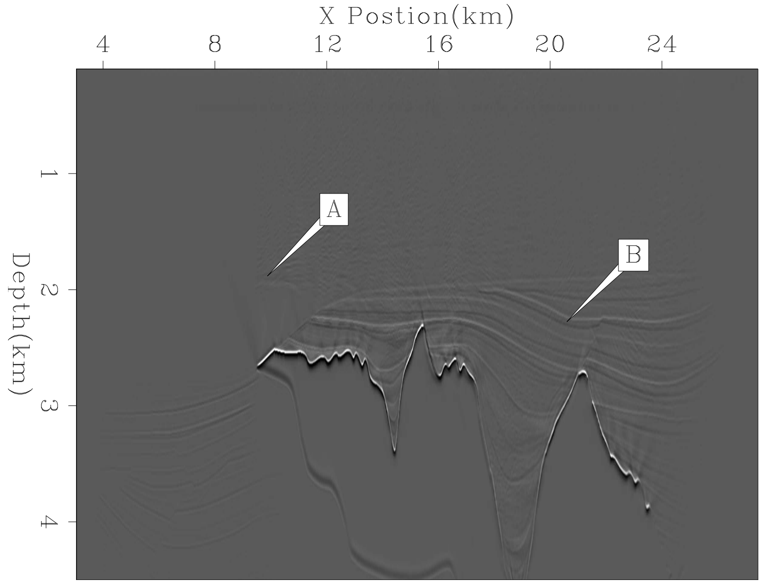


Figure 6: The result of RTM migration using a random boundary condition using the methodology described by algorithm 3. Note the greatly reduced artifacts at 'A' and 'B' [**CR**] bob1/. rand

## ACKNOWLEDGMENTS

We would like to thank the SmartJV for the use of the Sigsbee synthetic. research.

## REFERENCES

Baysal, E., D. D. Kosloff, and J. W. C. Sherwood, 1983, Reverse time migration: Geophysics, **48**, 1514–1524.

———, 1984, A two-way nonreflecting wave equation: Geophysics, **49**, 132–141.

Berenger, J., 1994, A perfectly matched layer for the absorption of electromagnetic waves: Journal of Computational Physics, **114**, 185–200.

Cerjan, C., D. Kosloff, R. Kosloff, and M. Reshef, 1985, A nonreflecting boundary condition for discrete acoustic-wave and elastic-wave equations (short note): Geophysics, **50**, 705–708.

Claerbout, J. F., 1998, Multi-dimensional recursive filtering via the helix: Geophysics, **63**, 1532–1541.

Clapp, R. G., 2008, Reverse time migration: Saving the boundaries: 2008, **136**, 136–144.

Dussaud, E., W. W. Symes, L. Lemaistre, P. Singer, B. Denel, and A. Cherrett, 2008, Computational strategies for reverse-time migration: 78th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, SPMI 3.3.

Micikevicius, P., 2008, 3d finite difference computation on gpus using cuda: 2nd Workshop on General Purpose Processing on Graphics Processing Units, Expanded Abstracts, 79–84.

Nemeth, T., J. Stefani, W. Liu, R. Dimond, O. Pell, , and R. Ergas, 2008, An implementation of the acoustic wave equation on fpgas: 78th Ann. Internat. Meeting, Expanded Abstracts, 2874–2877, Soc. Expl. Geophys.

Rickett, J., J. Claerbout, and S. B. Fomel, 1998, Implicit 3-D depth migration by wavefield extrapolation with helical boundary conditions: 68th Ann. Internat. Mtg, 1124–1127, Soc. of Expl. Geophys.

Symes, W. M., 2007, Reverse time migration with optimal checkpointing: Geophysics, **72**, SM213–SM221.

# Kinematics in Iterated Correlations of a Passive Acoustic Experiment

*Sjoerd de Ridder and George Papanicolaou*

## ABSTRACT

Correlating ambient seismic noise can yield the inter-station Green's function, but only if the energy that is excited by seismic background sources is sufficiently equipartitioned after averaging over all sources. If this requirement is not fulfilled, the reconstructed Green's function is imperfect. Secondary scattering can mitigate the directivity of the primary wave field emitted by the sources. To extract and utilize secondary scattering for Green's function reconstruction, we introduce a second correlation using an auxiliary station. We investigate the kinematics of the reconstructed Green's functions to understand the role of the positions of source, scatterer and auxiliary stations. Iterated correlations can use secondary scattering to mitigate the directivity in the background seismic wave field. In general, there will be additional spurious events in the retrieved Green's functions. Averaging the results of several sources and using a network of randomly distributed auxiliary stations can minimize these spurious events with respect to the correct events in the retrieved Green's functions.

## INTRODUCTION

It has long been known that correlations of seismic background noise recorded at two stations can yield the Green's function between the two stations (Aki, 1957; Claerbout, 1968; Lobkis and Weaver, 2001; Wapenaar, 2004), hereafter referred to as the estimated Green's function (EGF). A variety of proofs exist for this relation, including many based upon diffusivity of the wave fields (Weaver and Lobkis, 2001; Roux et al., 2005; Sánchez-Sesma et al., 2006; Sánchez-Sesma and Campillo, 2006), stationary-phase analysis (Schuster et al., 2004; Snieder, 2004; Snieder et al., 2006), and propagation invariants and reciprocity theorems (Claerbout, 1976; Weaver and Lobkis, 2004; Wapenaar, 2004; Wapenaar and Fokkema, 2006; van Manen et al., 2005). In general, these proofs require energy equipartitioning in the background seismic field; i.e., the energy flow must be equal in all directions. It is generally assumed that energy equipartitioning should be obtained after averaging over sources that excite the background field (Snieder et al., 2007). If the background noise field does not satisfy this condition, we expect the field correlations to recover imperfect EGFs (Malcolm et al., 2004; Paul et al., 2005).

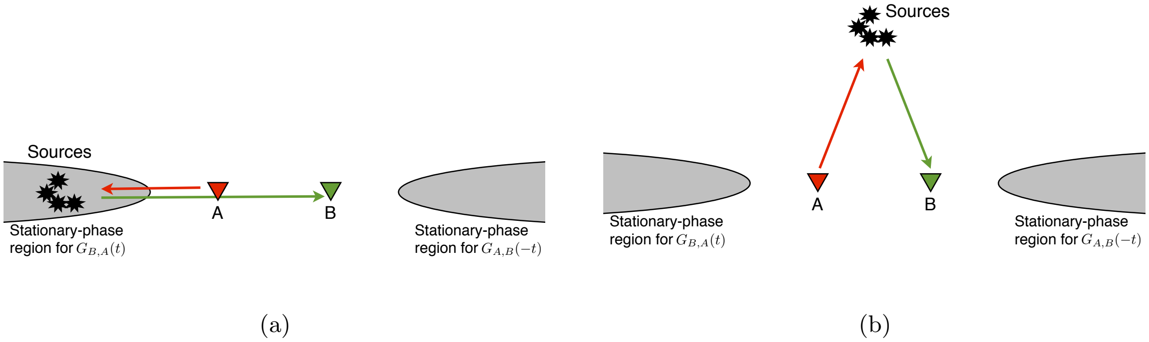
Recently it has been argued that multiple scattering by random inhomogeneities can excite a secondary wave field that satisfies the assumption of equipartitioning, even if the primary wave field does not (Stehly et al., 2008). It is also known that correlation of coda waves can yield the Green's function (Snieder, 2004; Malcolm et al., 2004; Paul et al., 2005; de Ridder, 2008). Stehly et al. (2008) describes a way to use the coda waves of background noise to improve the quality of EGFs (Stehly et al., 2008). Garnier and Papanicolaou

(2009) give a proof for this procedure to enhance Green's function estimation in random media based upon stationary-phase analysis of the four leading terms in the higher order correlation.

This paper discusses the problems associated with Green's function retrieval in directional wave fields. Then we proceed to briefly repeat the stationary-phase analysis of Garnier and Papanicolaou (2009) in the case of a wave field excited by one source in a homogeneous medium with the addition of one scatterer. The kinematics of the four leading terms are investigated using correlation gathers of auxiliary station position and source positions. Our examples show the basic procedure for reconstructing a Green's function by iterated correlations and provides a physical understanding of the elementary requirements for the positions of sources, random inhomogeneities, and auxiliary stations. This study has implications for seismic exploration using ambient seismic noise for different acquisition geometries, as in a network of stations only on the surface recording the ambient field above a reservoir, or a borehole survey with stations both down-hole and on the surface.

## CONVENTIONAL VERSUS ITERATIVE INTERFEROMETRY

Conventional seismic interferometry (SI) retrieves the Green's function between two stations by correlating, $C^{(2)}$, records of an ambient field, in which the energy is equipartitioned, recorded at both stations. It is generally assumed that energy equipartitioning should be obtained after averaging over sources that excite the background field (Snieder et al., 2007). Sources located at stationary phases are necessary to retrieve high-quality EGFs. For example, the stationary-phase region of the Green's function between stations $A$ and $B$ in Figure 1(a) is located on a ray path from station $B$ extending to and beyond station $A$ [gray shading on left side of Figure 1(a)]. Correlating responses from these sources recorded at $A$ and $B$ will retrieve a high-quality EGF. However, because the sources in Figure 5(b) are not located in the stationary-phase region, correlating responses from these sources recorded at $A$ and $B$ will retrieve a low-quality EGF.


Sources
Stationary-phase region for $G_{B,A}(t)$
A
B
Stationary-phase region for $G_{A,B}(-t)$
Sources
Stationary-phase region for $G_{B,A}(t)$
A
B
Stationary-phase region for $G_{A,B}(-t)$

(a) (b)

Figure 1: Source positions for respectively (a) high-quality and (b) poor quality Green's function estimation by conventional SI. Stationary-phase regions are indicated by gray areas. [**NR**] sjoerd1/. mainC2drawnew1,mainC2drawnew2

Some proposed methods to compensate for anisotropic illuminations include: (a) Beam forming and weighting (Stork and Cole, 2007) or $\tau - p$ filtering (Ruigrok et al., 2008) the data for different directionality components. (b) Estimating a radiation pattern by autocorrelating the down-going wave field and correcting by deconvolution (van der Neut et al., 2008; van der Neut and Bakulin, 2008). (c) Multidimensional deconvolution after the identification of individual responses (Wapenaar et al., 2008). Finally (d), Stehly et al. (2008) propose a novel procedure to improve EGFs by using scatterers positioned at the stationary-phase positions that act as secondary Huygens' sources, as illustrated in Figure 2. Their method requires three steps: First, the recordings at two main stations are correlated with a network of auxiliary stations. Each correlation yields an EGF. Second, each EGF is muted for times prior to an estimated arrival time. Third, a correlation, $C^3$, is evaluated between the muted EGF pairs estimated for each auxiliary station. That correlation is subsequently averaged across the network of auxiliary stations.

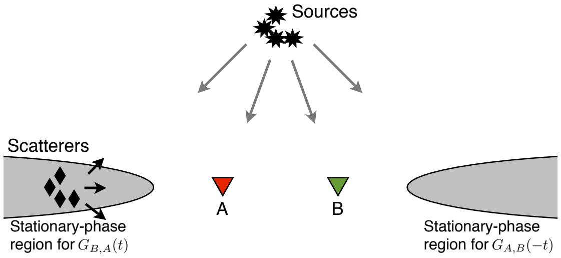

Sources
Scatterers
A
B
Stationary-phase region for $G_{B,A}(t)$
Stationary-phase region for $G_{A,B}(-t)$

Figure 2: Illustration of how scatterers acting as secondary Huygens' sources can illuminate stations $A$ and $B$ from a stationary-phase region, while the primary sources are located outside the stationary-phase regions. [**NR**] sjoerd1/. mainC4drawnew0

## GREEN'S FUNCTION RETRIEVAL BY CORRELATION

We define the temporal correlation function between two time signals $F_A(t)$ and $F_B(t)$ measured at stations A and B as

$$C^{(2)}_{B,A}(t) = \int_{-\infty}^{\infty} F_B(\tau + t)F_A(\tau)\mathrm{d}\tau = \frac{1}{2\pi}\int_{-\infty}^{\infty} F_B(\omega)F_A^*(\omega)\ \exp\{i\omega t\}\,\mathrm{d}\omega, \tag{1}$$

where $\omega$ denotes angular frequency. The right-hand side of equation 1 shows that through the inverse Fourier transformation of equation A-3, a correlation integral in the time domain is a direct product in the frequency domain. We can retrieve the Green's function between two stations $A$ and $B$ by independently measuring responses of sources positioned on a boundary surrounding the two stations, and summing the correlation between the measurements at the two stations. This property can be expressed as* (Wapenaar and Fokkema, 2006):

$$G(\mathbf{x}_B, \mathbf{x}_A, \omega) - G^*(\mathbf{x}_A, \mathbf{x}_B, \omega) = -\frac{2i\omega}{c_0}\oint_{\partial \mathbf{D}} G(\mathbf{x}_B, \mathbf{x}_s, \omega)G^*(\mathbf{x}_A, \mathbf{x}_s, \omega)\mathrm{d}\mathbf{x}_s, \tag{2}$$

where $\mathbf{x}_A$, $\mathbf{x}_B$ and $\mathbf{x}_s$ denote positions of stations $A$ and $B$ and the sources respectively.

We investigate the terms within this this integral for a medium containing a scatterer. The Green's function under the Born approximation in a scattering medium is composed of two terms:

$$G(\mathbf{x}, \mathbf{x}_s, \omega) = G_0(\mathbf{x}, \mathbf{x}_s, \omega) + G_1(\mathbf{x}, \mathbf{x}_s, \omega), \tag{3}$$

*We employ a different definition of the Green's function with respect to equation 31 of Wapenaar and Fokkema (2006), $G' = \frac{\rho}{i\omega}G$, where $\rho$ is density.

where $G_0$ is the contribution of the direct wave, and $G_1$ is the contribution of the scattered wave. In the Born approximation, the contribution of the scatterer is included to order $\alpha$. The correlation product between measurements made at stations $A$ and $B$ therefore is composed of $2^2 = 4$ terms

$$C^{(2)}_{B,A}(\omega) = G_0(\mathbf{x}_B, \mathbf{x}_s, \omega)G^*_0(\mathbf{x}_A, \mathbf{x}_s, \omega) + G_0(\mathbf{x}_B, \mathbf{x}_s, \omega)G^*_1(\mathbf{x}_A, \mathbf{x}_s, \omega) + G_1(\mathbf{x}_B, \mathbf{x}_s, \omega)G^*_0(\mathbf{x}_A, \mathbf{x}_s, \omega) + G_1(\mathbf{x}_B, \mathbf{x}_s, \omega)G^*_1(\mathbf{x}_A, \mathbf{x}_s, \omega), \tag{4}$$

which will be referred to as 4.1, 4.2, 4.3 and 4.4 respectively. The second and third terms are of order $\alpha$, and the fourth term is of order $\alpha^2$. Therefore, we should exclude the fourth term when we evaluate the right-hand side of equation 2 and compare it to the left-hand side of equation 2. See Snieder et al. (2008) for a more general discussion of the fourth term for exact Green's functions (without Born approximation). We denote the integration of $C^{(2)}_{B,A}$ over the source coordinate and multiplication by the phase-modifying factor as follows:

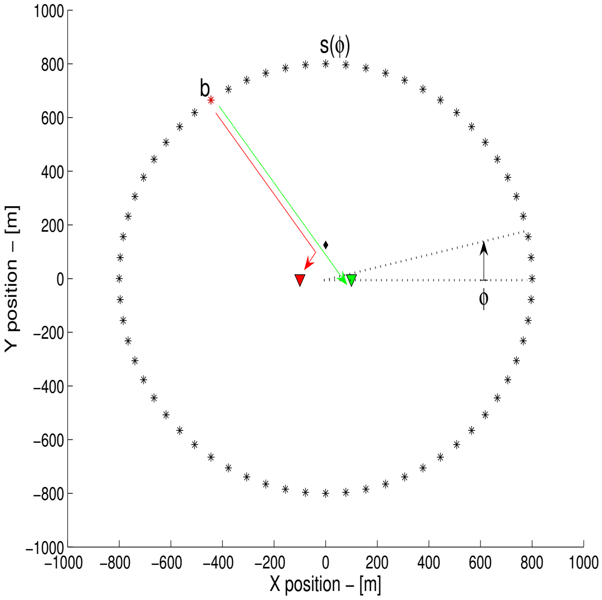


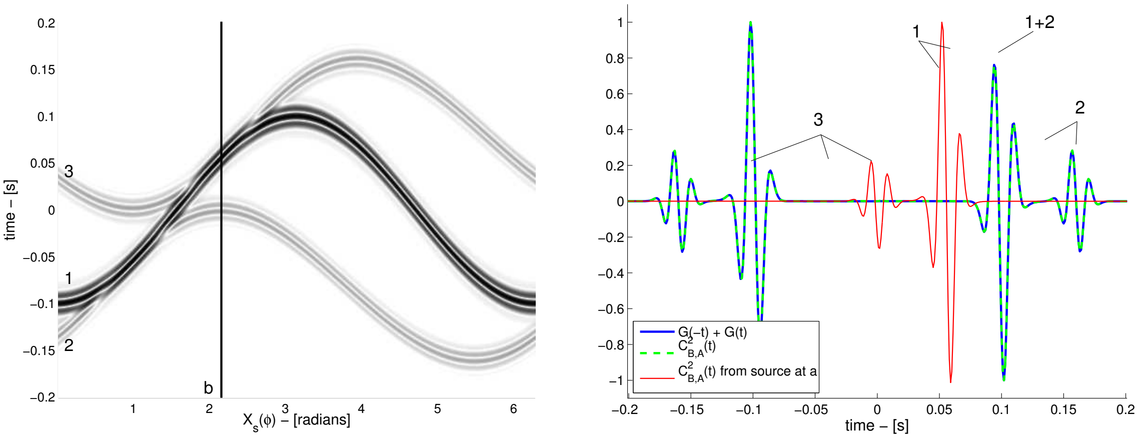
Figure 3: Geometry for the evaluation of $C^{(2)}_{B,A}$ in a homogeneous medium containing one scatterer. For three source positions, $a$, $b$ and $c$, two ray paths are shown for stationary phases; see text. [**ER**] sjoerd1/. geomC2

$$\tilde{C}^{(2)}_{B,A}(\omega) = -\frac{2i\omega}{c_0}\oint_{\partial\mathbf{D}} C^{(2)}_{B,A}(\omega)\mathrm{d}\mathbf{x}_s, \tag{5}$$

where $C^{(2)}_{B,A}(\omega)$ is an implicit function of source position $\mathbf{x}_s$, according to equation 4.

## STATIONARY-PHASE ANALYSIS OF CONVENTIONAL INTERFEROMETRY

The phase of the correlation under the integral on the right-hand side of equation 2 changes rapidly as a function of source position. The dominant contribution to the integral comes from points at which the phase is stationary. Physically these positions correspond to source points from where the ray paths to both stations align. To analyze the stationary phases in the presence of a scatterer, we consider a homogeneous medium and study the time-domain



(a)

(b)

Figure 4: a) Correlogram displaying correlations of source responses measured at stations A and B for sources as a function of position angle. b) Comparison of retrieved and true Green's functions. [**ER**] sjoerd1/. corrC2,resultC2

expression of equation 2 using the first three terms in $C^{(2)}_{B,A}$ of equation 4:

$$\begin{aligned} G(\mathbf{x}_B, \mathbf{x}_A, t) - G(\mathbf{x}_A, \mathbf{x}_B, -t) = & \qquad (6) \\ \tilde{C}^{(2)}_{B,A}(t) = -\int_{-\infty}^{\infty} \frac{2i\omega}{2\pi c_0} \oint_{\partial \mathbf{D}} \quad & a_0(\mathbf{x}_B, \mathbf{x}_s, \omega) a_0^*(\mathbf{x}_A, \mathbf{x}_s, \omega) \exp\{i\Omega_1\} \quad + \\ & a_0(\mathbf{x}_B, \mathbf{x}_s, \omega) a_1^*(\mathbf{x}_A, \mathbf{x}_s, \omega) \exp\{i\Omega_2\} \quad + \\ & a_1(\mathbf{x}_B, \mathbf{x}_s, \omega) a_0^*(\mathbf{x}_A, \mathbf{x}_s, \omega) \exp\{i\Omega_3\} \quad \mathrm{d}\mathbf{x}_s \mathrm{d}\omega, \end{aligned}$$

where $a_0$ and $a_1$ are amplitude factors. The rapid phases, $\Omega_1 = \Omega_1(\mathbf{x}_s, \mathbf{x}_A, \mathbf{x}_B, \omega)$, $\Omega_2(\mathbf{x}_s, \mathbf{x}_A, \mathbf{x}_B, \omega)$ and $\Omega_3(\mathbf{x}_s, \mathbf{x}_A, \mathbf{x}_B, \omega)$, of the three terms are found using equations A-6 and A-11:

$$\begin{aligned} \Omega_1 &= \omega \left[ t - c_0^{-1} \left\{ |\mathbf{x}_B - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_s| \right\} \right], & (7) \\ \Omega_2 &= \omega \left[ t - c_0^{-1} \left\{ |\mathbf{x}_B - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_c| - |\mathbf{x}_c - \mathbf{x}_s| \right\} \right], & (8) \\ \Omega_3 &= \omega \left[ t - c_0^{-1} \left\{ |\mathbf{x}_B - \mathbf{x}_c| + |\mathbf{x}_c - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_s| \right\} \right], & (9) \end{aligned}$$

where $\mathbf{x}_c$ is the position of the scatterer.

We analyze these rapid phases using the stationary-phase method, keeping $\mathbf{x}_A$ and $\mathbf{x}_B$ fixed and varying $\mathbf{x}_s$. According to the stationary-phase method, the dominant contribution comes from stationary phases where

$$\partial_\omega \Omega = 0 \quad \text{and} \quad \nabla_{\mathbf{x}_s} \Omega = \mathbf{0}. \tag{10}$$

From the rapid phase, $\Omega_1$, of the first term in equation 6, we find stationary points for which

$$t = c_0^{-1} \left\{ |\mathbf{x}_B - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_s| \right\} \quad \text{and} \quad \nabla_{\mathbf{x}_s} |\mathbf{x}_B - \mathbf{x}_s| = \nabla_{\mathbf{x}_s} |\mathbf{x}_A - \mathbf{x}_s|. \tag{11}$$

The second condition requires the points $\mathbf{x}_A$ and $\mathbf{x}_B$ to be aligned along a line issuing from $\mathbf{x_s}$. When the stations and source are aligned as $\mathbf{x}_s \to \mathbf{x}_A \to \mathbf{x}_B$, the first condition gives $t = c_0^{-1}\{|\mathbf{x}_B - \mathbf{x}_A|\}$. When the stations and source are aligned as $\mathbf{x}_s \to \mathbf{x}_B \to \mathbf{x}_A$, the first condition gives $t = -c_0^{-1}\{|\mathbf{x}_B - \mathbf{x}_A|\}$.

From rapid phase $\Omega_2$ of the second terms in equation 6, we find stationary points for which

$$t = c_0^{-1}\{|\mathbf{x}_B - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_c| - |\mathbf{x}_c - \mathbf{x}_s|\} \quad \text{and} \quad \nabla_{\mathbf{x}_s}|\mathbf{x}_B - \mathbf{x}_s| = \nabla_{\mathbf{x}_s}|\mathbf{x}_c - \mathbf{x}_s|. \tag{12}$$

The second condition requires the points $\mathbf{x}_B$ and $\mathbf{x}_c$ to be on a line issuing from $\mathbf{x}_s$. When station B, as well as the scatterer and sources are aligned as $\mathbf{x}_s \to \mathbf{x}_B \to \mathbf{x}_c$, then $|\mathbf{x}_c - \mathbf{x}_s| = |\mathbf{x}_c - \mathbf{x}_B| + |\mathbf{x}_B - \mathbf{x}_s|$, and the first condition states that $t = -c_0^{-1}\{|\mathbf{x}_B - \mathbf{x}_c| + |\mathbf{x}_c - \mathbf{x}_A|\}$. When station B, the scatterer and the source are aligned as $\mathbf{x}_s \to \mathbf{x}_c \to \mathbf{x}_B$, then $|\mathbf{x}_B - \mathbf{x}_s| = |\mathbf{x}_c - \mathbf{x}_s| + |\mathbf{x}_B - \mathbf{x}_c|$, and the first condition states that $t = c_0^{-1}\{|\mathbf{x}_B - \mathbf{x}_c| - |\mathbf{x}_A - \mathbf{x}_c|\}$.

From rapid phase $\Omega_3$ of the second terms in equation 6, we find stationary points for which

$$t = c_0^{-1}\{|\mathbf{x}_B - \mathbf{x}_c| + |\mathbf{x}_c - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_s|\} \quad \text{and} \quad \nabla_{\mathbf{x}_s}|\mathbf{x}_A - \mathbf{x}_s| = \nabla_{\mathbf{x}_s}|\mathbf{x}_c - \mathbf{x}_s|. \tag{13}$$

The second condition requires the points $\mathbf{x}_A$ and $\mathbf{x}_c$ to be on a line issuing from $\mathbf{x}_s$. When station A, as well as the scatterer and sources are aligned as $\mathbf{x}_s \to \mathbf{x}_A \to \mathbf{x}_c$, then $|\mathbf{x}_c - \mathbf{x}_s| = |\mathbf{x}_c - \mathbf{x}_A| + |\mathbf{x}_A - \mathbf{x}_s|$, and the first condition states that $t = c_0^{-1}\{|\mathbf{x}_B - \mathbf{x}_c| + |\mathbf{x}_c - \mathbf{x}_A|\}$. When station A, the scatterer and sources are aligned as $\mathbf{x}_s \to \mathbf{x}_c \to \mathbf{x}_A$, then $|\mathbf{x}_A - \mathbf{x}_s| = |\mathbf{x}_A - \mathbf{x}_c| + |\mathbf{x}_c - \mathbf{x}_s|$, and the first condition states that $t = c_0^{-1}\{|\mathbf{x}_B - \mathbf{x}_c| - |\mathbf{x}_A - \mathbf{x}_c|\}$. For a more extensive treatment of stationary-phase positions in conventional interferometry, see Schuster et al. (2004); Snieder (2004); Snieder et al. (2006) and Garnier and Papanicolaou (2009).

## EXAMPLE OF GREEN'S FUNCTION RETRIEVAL BY CONVENTIONAL INTERFEROMETRY

To aid interpretation of iterated interferometry in later sections, we study the kinematics of conventional interferometry for a medium containing a scatterer. The background velocity is $c_0 = 2000$ m/s. Stations $A$ and $B$ are positioned 200 m distant from each other, and the scatterer is positioned 125 m above and in between the stations. The stations and scatterer are surrounded by 512 sources on a circle with a radius of 800 m, centered between the two stations; see Figure 3.

We simulate the measurements at stations $A$ and $B$ using the single-scatterer Born approximation (see appendix  equation A-11). We assume all sources emit a zero-phase Ricker wavelet, $s(t)$ (see appendix equation A-8). For each source location separately, the responses recorded at stations A and B are cross-correlated, and their contribution to the integral on the right-hand side of equation 2 is shown as a function of angle in the correlogram in Figure 4(a).

The correlogram contains three events labeled 1, 2 and 3. These correspond to the first three terms respectively in the correlation product in equation 4. Term 1 is associated with the direct event between stations $A$ and $B$. It has two stationary points at angles of $\phi = 0$

and $\phi = \pi$ radians, where the stations and source are aligned on a line as $\mathbf{x}_s \rightarrow \mathbf{x}_B \rightarrow \mathbf{x}_A$ and $\mathbf{x}_s \rightarrow \mathbf{x}_A \rightarrow \mathbf{x}_B$, respectively. For all other angles, the correlation peak resides at a lag that is smaller than the actual travel time between the stations. The second and third terms correspond to correlations of recorded events that are either scattered at A and direct at B or vice versa. Both events have two stationary phases. Event 2, for example, has a stationary phase for a source positioned close to $\phi = 3/4\pi$ where $\mathbf{x}_s \rightarrow \mathbf{x}_c \rightarrow \mathbf{x}_B$, and at approximately $\phi = 4/3\pi$ where $\mathbf{x}_s \rightarrow \mathbf{x}_B \rightarrow \mathbf{x}_c$. The total correlogram is summed over all angles and multiplied by a factor $-\frac{2i\omega}{c_0}$, according to equation 5, to match the asymmetrized true Green's function on the left-hand side of equation 2. The asymmetrized Green's function is multiplied with the auto-correlation of the Ricker wavelet to match the source function after correlations. Although the calculation matches before normalization, the Green's functions are normalized to have a peak value of 1. The comparison between the retrieved result (dashed green line) found by evaluating the right-hand side of equation 2 and the directly modeled result (solid blue curve) found by computing the left-hand side of equation 2 is shown in Figure 4(b), they match exactly.

Three contributions of stationary angles are isolated from all other source contributions and compared to the fully retrieved result. These stationary angles have events arriving with the correct travel time but incorrect phase. They also have events with incorrect travel times. However the contribution from a source positioned close to $\phi = 4/3\pi$ radians seems to have an event with a travel time approximately corresponding to the acausal direct event. It is non-stationary and associated with the acausal scattered events as can be seen in Figure 4(a).

## GREEN'S FUNCTION RETRIEVAL BY ITERATED CORRELATIONS

In the absence of complete source coverage, we can make use of the scattering properties of the medium to mitigate the directivity of the wave field. The iterated correlation between stations $B$ and $A$ is defined using auxiliary station X as follows:

$$\begin{aligned} C^{(3)}_{B,A}(t) &= \int_{-\infty}^{\infty} C^{(2)}_{B,X}(\tau' + t) C^{(2)}_{A,X}(\tau') \mathrm{d}\tau' \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F_B(\tau + \tau' + t) F_X(\tau) F_X(s) F_A(s + \tau') \mathrm{d}s \mathrm{d}\tau \mathrm{d}\tau' \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty} F_B(\omega) F^*_X(\omega) F_X(\omega) F^*_A(\omega) \exp\{i\omega t\} \, \mathrm{d}\omega \qquad (14) \end{aligned}$$

The Green's function in the Born approximation for a scattering medium is composed of two terms; $C^{(3)}_{B,A}$ therefore contains $2^4 = 16$ terms

$$
\begin{array}{rll}
C^{(3)}_{B,A} = & G_0(\mathbf{x}_B,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_0(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.1)\\
& G_0(\mathbf{x}_B,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_1(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.2)\\
& G_0(\mathbf{x}_B,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_0(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.3)\\
& G_0(\mathbf{x}_B,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_1(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.4)\\
& G_0(\mathbf{x}_B,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_0(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.5)\\
& G_0(\mathbf{x}_B,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_1(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.6)\\
& G_0(\mathbf{x}_B,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_0(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.7)\\
& G_0(\mathbf{x}_B,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_1(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.8)\\
& G_1(\mathbf{x}_B,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_0(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.9)\\
& G_1(\mathbf{x}_B,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_1(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.10)\\
& G_1(\mathbf{x}_B,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_0(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.11)\\
& G_1(\mathbf{x}_B,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_1(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.12)\\
& G_1(\mathbf{x}_B,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_0(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.13)\\
& G_1(\mathbf{x}_B,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_1(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.14)\\
& G_1(\mathbf{x}_B,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_0(\mathbf{x}_X,\mathbf{x}_s,\omega)\,+ & (15.15)\\
& G_1(\mathbf{x}_B,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_X,\mathbf{x}_s,\omega)G_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega)G_1(\mathbf{x}_X,\mathbf{x}_s,\omega). & (15.16)
\end{array}
\tag{15}
$$

Three groups of terms can be distinguished; group 1 includes terms 15.1, 15.2, 15.3, 15.4, 15.5, 15.9 and 15.13, which are terms correlating with the dominant contribution in $C^{(2)}$; $G_0G_0^*$. Group 2 contains the terms of interest in this paper; 15.6, 15.7, 15.10 and 15.11; see the stationary-phase analysis below. The third group contains events that are of order $\alpha^3$ and includes terms 15.8, 15.12, 15.14, 15.15 and 15.16. The leading term in $C^{(2)}$ contributes to a spurious term, because the source is not located at a stationary angle of the event between stations A and B. To exclude the terms of group 1, we remove the dominant term after forming $C^{(2)}_{B,X}$ and $C^{(2)}_{A,X}$. This is done by muting the correlation in the time domain to suppress all times smaller than $\tau_{coda}$:

$$
\begin{aligned}
C^{(3)}_{B,A}(t) \quad &= \quad \int_{-\infty,-\tau'_{coda}}^{\tau'_{coda},\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F_B(\tau+\tau'+t)F_X(\tau)F_X(s)F_A(s+\tau')\mathrm{d}s\mathrm{d}\tau\mathrm{d}\tau' \qquad (16)\\
&= \quad \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\beta(\tau')F_B(\tau+\tau'+t)F_X(\tau)F_X(s)F_A(s+\tau')\mathrm{d}s\mathrm{d}\tau\mathrm{d}\tau', \qquad (17)
\end{aligned}
$$

where $\tau_{coda}$ is defined as an estimated traveltime between the main stations and the auxiliary stations, $\beta(\tau)$ is a muting function that is zero for $\beta(\tau) = 0$ for $\tau\ :\ [-\tau_{coda} : \tau_{coda}]$ and otherwise $\beta(\tau) = 1$.

We learned from Figure 3 that the dominant term always arrives within that time window. We average the iterated correlations over a network of $A$ auxiliary stations and include a phase-modifying term,

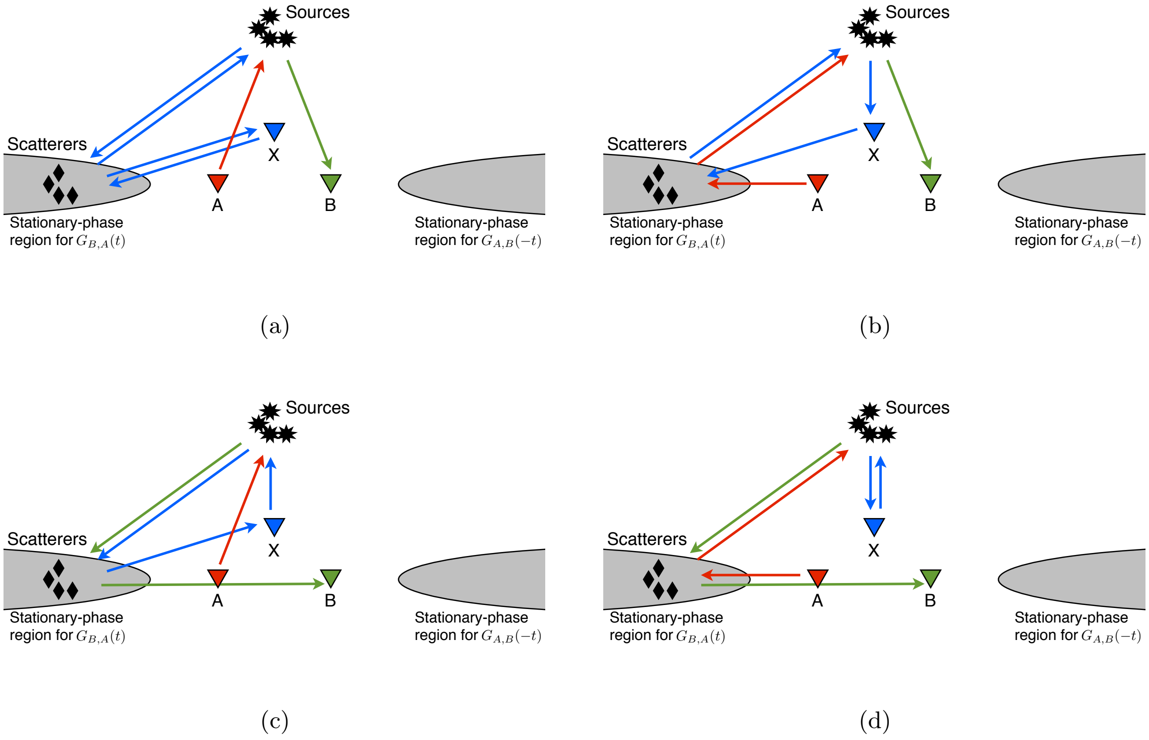
$$
\tilde{C}^{(3)}_{B,A}(\omega) = \frac{2c_0}{i\omega A}\sum_{a=1}^{A} C^{(3)}_{B,A}(\omega), \tag{18}
$$

where $\tilde{C}^{(3)}_{B,A}(\omega)$ is an implicit function of auxiliary-station position $\mathbf{x}_{X,a}$, according to equation 15. The phase-modifying proportionality factor is chosen such that the $\frac{\omega^2}{c_0^2}$ factor in the Born approximation (see equation A-10) is matched to the $\frac{-2i\omega}{c_0}$ factor in conventional interferometry (equation 2).

## STATIONARY PHASES IN ITERATED CORRELATIONS

We proceed by studying the stationary phases of terms 15.6, 15.7, 15.10 and 15.11 in the iterated correlation. All terms correspond to particular combinations of ray paths.

Figure 5 shows for each term a graphical illustration of the combination of ray paths. Ray paths towards the source are subtracted from the ray paths emitting from the source, as in the correlation process (a convolution of one Green's function with the time reverse of another Green's function). The time domain of equation 15, including only the terms


Sources
Scatterers
X
A
B
Stationary-phase region for $G_{B,A}(t)$
Stationary-phase region for $G_{A,B}(-t)$
Sources
Scatterers
X
A
B
Stationary-phase region for $G_{B,A}(t)$
Stationary-phase region for $G_{A,B}(-t)$
Sources
Scatterers
X
A
B
Stationary-phase region for $G_{B,A}(t)$
Stationary-phase region for $G_{A,B}(-t)$
Sources
Scatterers
X
A
B
Stationary-phase region for $G_{B,A}(t)$
Stationary-phase region for $G_{A,B}(-t)$

(a)

(b)

(c)

(d)

Figure 5: Geometrical interpretation of the correlations in terms 15.6, 15.7, 15.10 and 15.11 respectively in a) b) c) and d). [**NR**]
sjoerd1/. mainC4drawnew1,mainC4drawnew2,mainC4drawnew3,mainC4drawnew4

of group 2, is given as

$$
\begin{aligned}
\tilde{C}^{(3)}_{B,A}(t) = \int_{-\infty}^{\infty} \frac{2c_0}{2\pi i \omega A} \sum_{a=1}^{A} & \\
& a_0(\mathbf{x}_B,\mathbf{x}_s,\omega) a_1^*(\mathbf{x}_{X,a},\mathbf{x}_s,\omega) a_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega) a_1(\mathbf{x}_{X,a},\mathbf{x}_s,\omega) \mathrm{exp}\left\{i\Omega_1\right\} \quad + \\
& a_0(\mathbf{x}_B,\mathbf{x}_s,\omega) a_1^*(\mathbf{x}_{X,a},\mathbf{x}_s,\omega) a_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega) a_0(\mathbf{x}_{X,a},\mathbf{x}_s,\omega) \mathrm{exp}\left\{i\Omega_2\right\} \quad + \\
& a_1(\mathbf{x}_B,\mathbf{x}_s,\omega) a_0^*(\mathbf{x}_{X,a},\mathbf{x}_s,\omega) a_0^*(\mathbf{x}_A,\mathbf{x}_s,\omega) a_1(\mathbf{x}_{X,a},\mathbf{x}_s,\omega) \mathrm{exp}\left\{i\Omega_3\right\} \quad + \\
& a_1(\mathbf{x}_B,\mathbf{x}_s,\omega) a_0^*(\mathbf{x}_{X,a},\mathbf{x}_s,\omega) a_1^*(\mathbf{x}_A,\mathbf{x}_s,\omega) a_0(\mathbf{x}_{X,a},\mathbf{x}_s,\omega) \mathrm{exp}\left\{i\Omega_4\right\} \quad \mathrm{d}\mathbf{x}_x \mathrm{d}\omega,
\end{aligned}
\tag{19}
$$

where $a_0$ and $a_1$ are amplitude factors. The rapid phases, $\Omega_1$, $\Omega_2$, $\Omega_3$ and $\Omega_4$ are found using equations A-6 and A-11; for a particular auxiliary station $\mathbf{x}_X$, we find

$$
\begin{aligned}
\Omega_1 &= \omega\left[t - c_0^{-1}\left\{|\mathbf{x}_B - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_s|\right\}\right], && (20)\\
\Omega_2 &= \omega\left[t - c_0^{-1}\left\{|\mathbf{x}_B - \mathbf{x}_s| - |\mathbf{x}_X - \mathbf{x}_c| - 2|\mathbf{x}_c - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_c| + |\mathbf{x}_X - \mathbf{x}_s|\right\}\right], && (21)\\
\Omega_3 &= \omega\left[t - c_0^{-1}\left\{|\mathbf{x}_B - \mathbf{x}_c| + 2|\mathbf{x}_c - \mathbf{x}_s| - |\mathbf{x}_X - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_s| + |\mathbf{x}_X - \mathbf{x}_c|\right\}\right], && (22)\\
\Omega_4 &= \omega\left[t - c_0^{-1}\left\{|\mathbf{x}_B - \mathbf{x}_c| - |\mathbf{x}_A - \mathbf{x}_c|\right\}\right]. && (23)
\end{aligned}
$$

We analyze these rapid phases using the stationary-phase method, keeping $\mathbf{x}_A$ and $\mathbf{x}_B$ fixed and varying $\mathbf{x}_X$, $\mathbf{x}_c$ and $\mathbf{x}_s$. According to the stationary-phase method, the dominant contribution to the integral and sum in equation 19 comes from positions of $\mathbf{x}_X$, $\mathbf{x}_c$ and $\mathbf{x}_s$ where

$$
\partial_\omega \Omega = 0, \qquad \nabla_{\mathbf{x}_s}\Omega = \mathbf{0}, \qquad \nabla_{\mathbf{x}_X}\Omega = \mathbf{0} \quad \text{and} \quad \nabla_{\mathbf{x}_c}\Omega = \mathbf{0} \tag{24}
$$

From the rapid phase, $\Omega_1$, of the first term in equation 19 we find stationary points for which

$$
\begin{aligned}
t &= c_0^{-1}\left\{|\mathbf{x}_B - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_s|\right\}, && (25)\\
\nabla_{\mathbf{x}_s}|\mathbf{x}_B - \mathbf{x}_s| &= \nabla_{\mathbf{x}_s}|\mathbf{x}_A - \mathbf{x}_s|, && (26)\\
\nabla_{\mathbf{x}_X}|\mathbf{x}_B - \mathbf{x}_s| &= \nabla_{\mathbf{x}_X}|\mathbf{x}_A - \mathbf{x}_s|, && (27)\\
\nabla_{\mathbf{x}_c}|\mathbf{x}_B - \mathbf{x}_s| &= \nabla_{\mathbf{x}_c}|\mathbf{x}_A - \mathbf{x}_s|. && (28)
\end{aligned}
$$

Conditions 27 and 28 are always satisfied. Condition 26 requires the stations to be on a line and the source to be on a line issuing from $\mathbf{x_s}$. When the stations and source are aligned as $\mathbf{x}_s \rightarrow \mathbf{x}_A \rightarrow \mathbf{x}_B$, condition 25 gives $t = c_0^{-1}|\mathbf{x}_B - \mathbf{x}_A|$. When the stations and source are aligned as $\mathbf{x}_s \rightarrow \mathbf{x}_B \rightarrow \mathbf{x}_A$, the first condition gives $t = -c_0^{-1}|\mathbf{x}_B - \mathbf{x}_A|$.

From the rapid phase, $\Omega_2$, of the first term in equation 19 we find stationary points for which

$$
\begin{aligned}
t &= c_0^{-1}\left\{|\mathbf{x}_B - \mathbf{x}_s| - |\mathbf{x}_X - \mathbf{x}_c| - 2|\mathbf{x}_c - \mathbf{x}_s| - |\mathbf{x}_A - \mathbf{x}_c| + |\mathbf{x}_X - \mathbf{x}_s|\right\}, && (29)\\
\nabla_{\mathbf{x}_s}\left\{|\mathbf{x}_B - \mathbf{x}_s| - |\mathbf{x}_c - \mathbf{x}_s|\right\} &= \nabla_{\mathbf{x}_s}\left\{|\mathbf{x}_c - \mathbf{x}_s| - |\mathbf{x}_X - \mathbf{x}_s|\right\}, && (30)\\
\nabla_{\mathbf{x}_X}|\mathbf{x}_X - \mathbf{x}_c| &= \nabla_{\mathbf{x}_X}|\mathbf{x}_X - \mathbf{x}_s|, && (31)\\
-\nabla_{\mathbf{x}_c}\left\{|\mathbf{x}_c - \mathbf{x}_s| + |\mathbf{x}_X - \mathbf{x}_c|\right\} &= \nabla_{\mathbf{x}_c}\left\{|\mathbf{x}_A - \mathbf{x}_c| + |\mathbf{x}_c - \mathbf{x}_s|\right\}. && (32)
\end{aligned}
$$

Condition 30 requires that station B, auxiliary station, the scatterer are on a line issuing from the source. Condition 31 requires that the auxiliary station and the scatterer are on a line issuing from the source. Condition 32 requires that station $A$, an auxiliary station and a scatterer are on a line issuing from the source. In short, stations $A$ and $B$, an auxiliary station, and the scatterer all align on a line issuing from the source. When these are aligned as $\mathbf{x_s} \rightarrow \mathbf{x}_c \rightarrow \mathbf{x}_X \rightarrow \mathbf{x}_A \rightarrow \mathbf{x}_B$, then $|\mathbf{x}_X - \mathbf{x}_s| = |\mathbf{x}_X - \mathbf{x}_c| + |\mathbf{x}_c - \mathbf{x}_s|$, $|\mathbf{x}_c - \mathbf{x}_s| + |\mathbf{x}_A - \mathbf{x}_c| = |\mathbf{x}_A - \mathbf{x}_s|$, and condition 29 gives $t = c_0^{-1}|\mathbf{x}_B - \mathbf{x}_A|$. When stations $A$ and $B$ are reversed, condition 29 gives $t = -c_0^{-1}|\mathbf{x}_B - \mathbf{x}_A|$.

The rapid phase, $\Omega_3$, of the third term in equation 19 is similar to the rapid phase, $\Omega_2$, of the second term in equation 19. If stations $A$, $B$, an auxiliary station, and the scatterer are located on a line issuing from the source, aligned as $\mathbf{x_s} \rightarrow \mathbf{x}_c \rightarrow \mathbf{x}_X \rightarrow \mathbf{x}_A \rightarrow \mathbf{x}_B$, the dominant contribution resides at $t = c_0^{-1}|\mathbf{x}_B - \mathbf{x}_A|$. When stations $A$ and $B$ are interchanged, the dominant contribution of the third term in equation 19 resides at $t = -c_0^{-1}|\mathbf{x}_B - \mathbf{x}_A|$. Last we analyze the rapid phase, $\Omega_4$, of the fourth term in equation 19, and we find stationary points for which

$$t = c_0^{-1}\left\{|\mathbf{x}_B - \mathbf{x}_c| - |\mathbf{x}_A - \mathbf{x}_c|\right\}, \tag{33}$$

$$\nabla_{\mathbf{x}_s}|\mathbf{x}_B - \mathbf{x}_c| = \nabla_{\mathbf{x}_s}|\mathbf{x}_A - \mathbf{x}_c|, \tag{34}$$

$$\nabla_{\mathbf{x}_X}|\mathbf{x}_B - \mathbf{x}_c| = \nabla_{\mathbf{x}_X}|\mathbf{x}_A - \mathbf{x}_c|, \tag{35}$$

$$\nabla_{\mathbf{x}_c}|\mathbf{x}_B - \mathbf{x}_c| = \nabla_{\mathbf{x}_c}|\mathbf{x}_A - \mathbf{x}_c|. \tag{36}$$

Conditions 34 and 35 are always satisfied. Condition 36 is satisfied when the scatterer lies on a line through stations $A$ and $B$. When stations $A$, $B$ and the scatterer are aligned as $\mathbf{x}_c \rightarrow \mathbf{x}_A \rightarrow \mathbf{x}_B$, condition 33 gives $t = c_0^{-1}|\mathbf{x}_B - \mathbf{x}_A|$. When stations $A$, $B$ and the scatterer align as $\mathbf{x}_c \rightarrow \mathbf{x}_B \rightarrow \mathbf{x}_A$, condition 33 gives $t = -c_0^{-1}|\mathbf{x}_B - \mathbf{x}_A|$.

## EXAMPLE OF GREEN'S FUNCTION ITERATED CORRELATION

We next study how forming $C^{(3)}_{B,A}$ of a wave field excited by a single source can improve the retrieved Green's function in the presence of an auxiliary scatterer. We study a geometry where the main stations are located 200 m distant from each other (see Figure 6). We use 512 auxiliary stations located on a circle with radius 300 m centered between the two main stations. The source is located at $s'$, with a distance of 800 m from the center and at an angle of $\phi = 3/4\pi$ radians. There is a scatterer positioned at a distance of 550 m from the center at an angle of $\phi = \pi$ radians. We omit the terms of group 3 in equation 15, because their contribution is at least of order $\alpha$ weaker than those in group 2. $C^{(2)}$ is evaluated between stations $A$ or $B$ and all the auxiliary stations $X$, yielding $C^{(2)}_{A,X}$ and $C^{(2)}_{B,X}$; the obtained correlograms are shown in Figures 7(a) and 7(b). We evaluate $C^{(3)}_{B,A}$ for each auxiliary station, including all terms of groups 1 and 2, and compile the result in a correlogram shown in Figure 8(a).

The contribution of each term is labeled according to the numbering of equation 15. We sum $C^{(3)}_{B,A}$ over the auxiliary stations, according to equation 18, to obtain the retrieved signal in Figure 8(b). We compare this signal to the true result, convolved with the square of the autocorrelation of the wavelet $S(\omega)$, and the result retrieved by correlating stations $B$ and $A$ directly ($C^{(2)}_{B,A}$) weighted by $-i\omega S(\omega)$. It is clear that the dominant contribution in $C^{(3)}_{B,A}$, without muting $C^{(2)}_{A,X}$ and $C^{(2)}_{B,X}$, does not correspond to the direct event between the stations $A$ and $B$. If we assume we can perfectly mute only the dominant term 4.1 from $C^{(2)}_{A,X}$ and $C^{(2)}_{B,X}$, this would leave the terms of group 2.

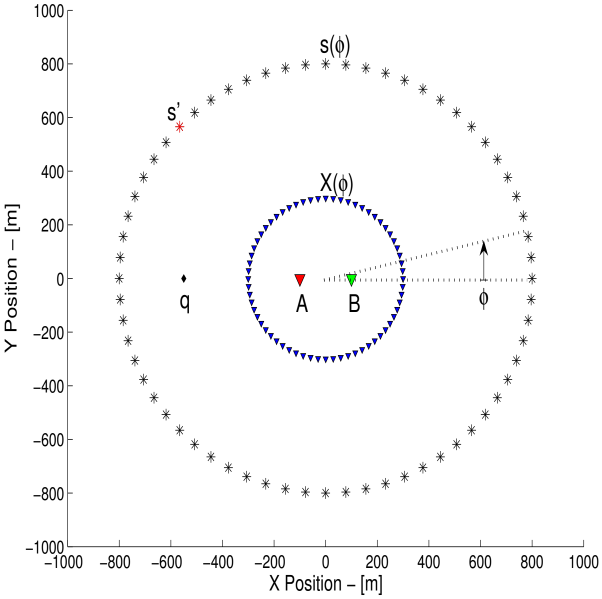


Figure 6: Experiment geometry for the evaluation of $C^{(3)}_{B,A}$. [**ER**] sjoerd1/. geomC4

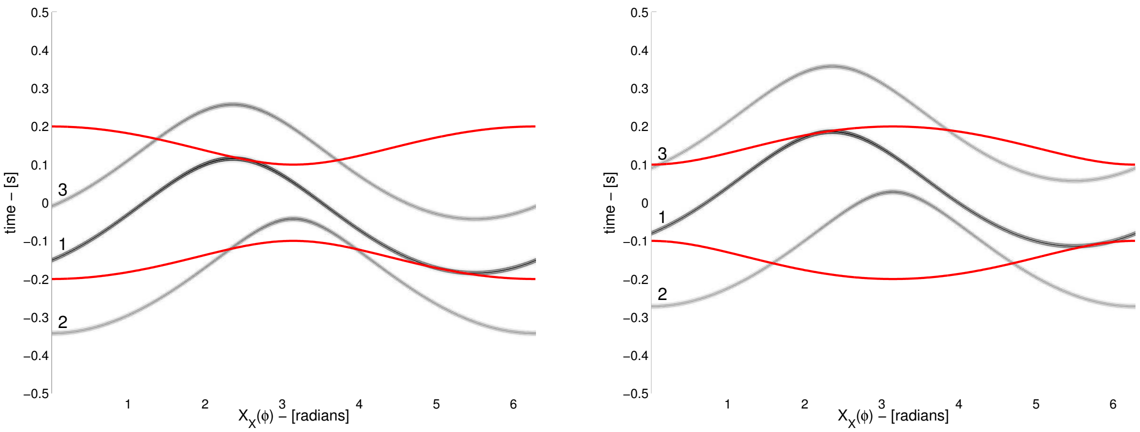
A correlogram of their contributions to $C^{(3)}_{B,A}$ is shown in Figure 9(a), summing this panel and multiplying with a phase-modifying according to equation 18, leads to the signal in Figure 9(b). We now see that there is a dominant term coinciding with the causal direct event between stations $A$ and $B$ in the true result; this event comes from term 15.11.

## ITERATED CORRELATION AFTER MUTING

The terms in group 2 cannot uniquely be separated from those of groups 1. Time-domain muting of $C^{(2)}_{B,X}$ and $C^{(2)}_{B,X}$ can exclude the leading order event 4.1, but would also exclude parts of terms 4.2 and 4.3. The black lines in Figures 7(a) and 7(b) indicate the travel time of an event between station $A$ or $B$ and each auxiliary station. The dominant term in $C^{(2)}$ will always reside in this window, see Figure 4(a). We now mute each $C^{(2)}$ according to these limits to obtain the two correlograms in Figures 10(a) and 10(b).

The $C^{(3)}_{B,A}$ is evaluated for each auxiliary station to obtain the correlogram in Figure 11(a), this panel is summed and multiplied with a phase-modifying according to equation 18 to retrieve the signal in Figure 11(b).

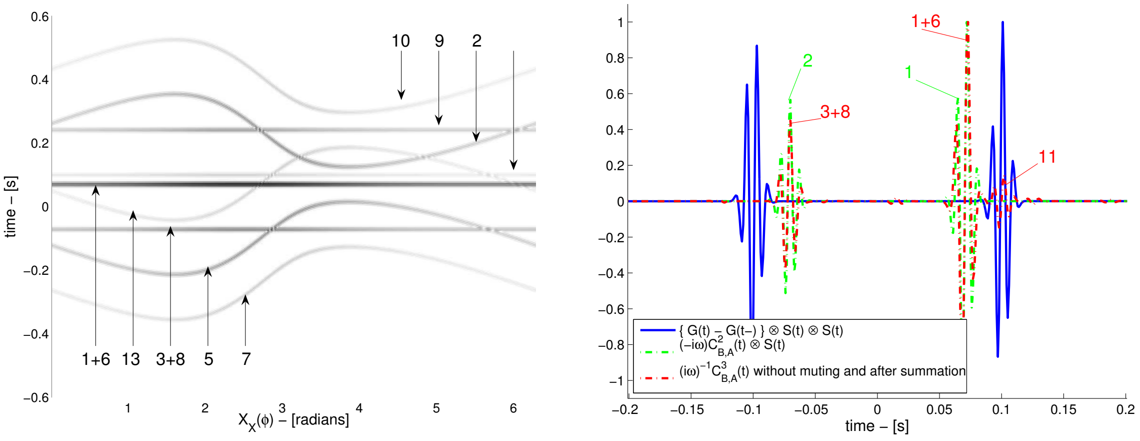
The resulting signal resembles the true result slightly better than evaluating the terms of equation 15 group 2 without muting; the spurious event arriving at $t = .8$s is slightly smaller. This is because, for the present geometry, the auxiliary stations where the spurious event is absent, would have contributed more strongly to the spurious event without muting before evaluating $C^{(3)}_{B,A}$. (The geometrical spreading factors vary for the contribution of each auxiliary station.)



(a)

(b)

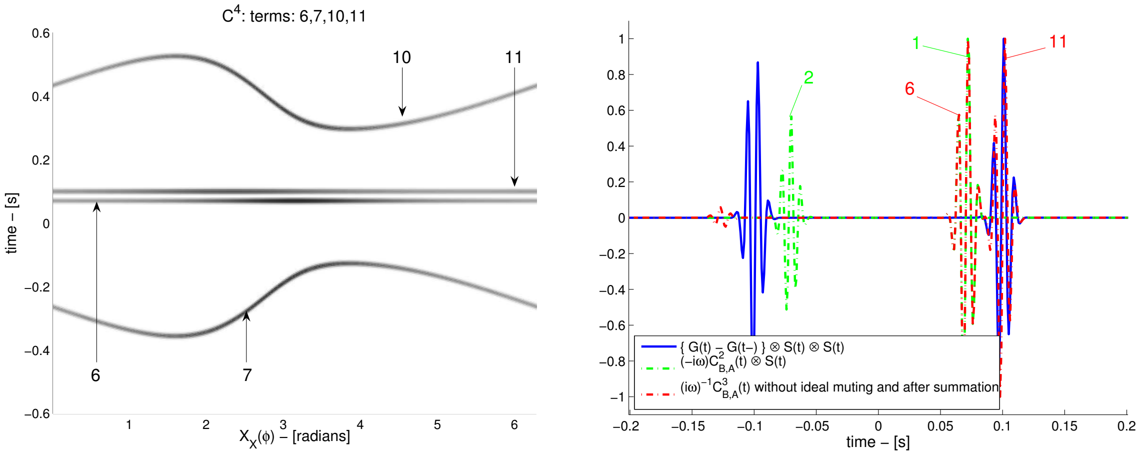
Figure 7: a) Correlogram for correlations between station $A$ and all auxiliary stations as a function of auxiliary station-position angle. Black lines indicates traveltime of a wave from station $A$ to each auxiliary station. b) Correlogram for correlations between station $B$ and all auxiliary stations as a function of auxiliary station-position angle. Black line indicates traveltime of a wave from station or $B$ to each auxiliary station. [**ER**] sjoerd1/. corrC2a,corrC2b



(a)

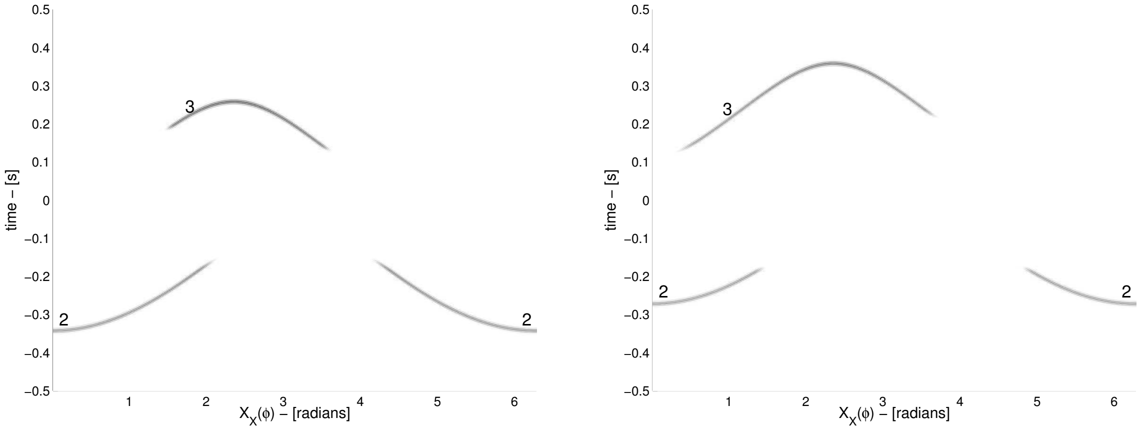
(b)

Figure 8: a) Correlogram of $C^{(3)}_{B,A}$ for each auxiliary station, including all 11 terms in groups 1 and 2 of equation 15. b) Correlogram of $C^{(3)}_{B,A}$ for each auxiliary station, including only the 4 terms from groups 2 of equation 15. [**ER**] sjoerd1/. corrC4ABa,resultC4a



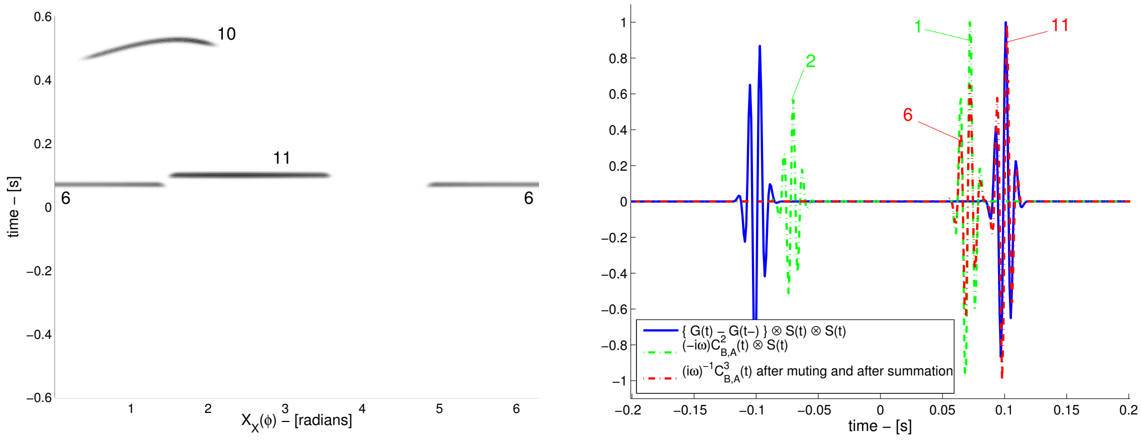
(a) (b)

Figure 9: a) Comparison of reconstructed Green's function with the true result, after summation of all 11 terms of groups 1 and 2 over auxiliary station. b) Comparison of reconstructed Green's function with true result, after summation of 4 terms of group 2 over auxiliary station. [**ER**] sjoerd1/. corrC4ABb,resultC4b



(a) (b)

Figure 10: Muted $C^{(2)}_{A,X}$ in a) and $C^{(2)}_{B,X}$ in b). These are the input for the evaluation of $C^{(3)}_{B,A}$. [**ER**] sjoerd1/. corrC2c,corrC2d



(a)

(b)

Figure 11: a) Correlogram of $C^{(3)}_{B,A}$ evaluated after muting $C^{(2)}$, as a function of each auxiliary station-position angle $\phi$. b) Comparison of retrieved Green's function with the true result, after summation of $C^{(3)}_{B,A}$ over all auxiliary stations. [**ER**] sjoerd1/. corrC4ABc,resultC4c

## ITERATED CORRELATION DEPENDANCE ON SOURCE POSITION

For the geometry in Figure 6, the source, stations A and B, auxiliary stations and scatterer are not aligned at a stationary phase of terms 15.6, 15.7 and 15.10. We will investigate the retrieved result of evaluating $C^{(3)}_{B,A}$ after muting $C^{(2)}_{A,X}$ and $C^{(2)}_{B,X}$, and summing over all auxiliary stations and multiplying with the phase-modifying factor as in equation 18. The sources are positioned on a circle with radius 800 m centered between stations $A$ and $B$ in the geometry described as before; see Figure 6. Evaluating $C^{(3)}_{B,A}$ and summation over the auxiliary stations for terms 15.6 15.7, 15.10 and 15.11, and then evaluating equation 18 for each source contribution gives the correlogram in Figure 12(a).

This correlogram confirms that when the source, stations $A$ and $B$, an auxiliary station, and the scatterer are aligned, each term has a stationary phase. We also see how term 15.11 is stationary with respect to source postion. The behavior of term 15.6 is similar to that of the leading term in $C^{(2)}_{B,A}$; see Figure 4(a). This can be expected from the constraint on $t$ in condition 25, which is equal to condition 11 on $t$ for $C^{(2)}_{B,A}$. We can expect that when we time-average the $C^{(3)}_{B,A}$ of multiple sources at different angular positions, term 15.6 interferes destructively.

Figure 12(a) also tells us that terms 15.7 and 15.10 are also non-stationary with respect to source position. However, the arrival time of non-stationary positions is dependent upon scatterer position (see condition 29); this implies that in a medium with randomly positioned scatterers, terms 15.7 and 15.10 would interfere destructively. Last we investigate whether muting $C^{(2)}$ before evaluating $C^{(3)}_{B,A}$ can work for the source postions located at stationary phases for terms 15.6 15.7 and 15.10; see Figure 12(b). We see how muting the $C^{(2)}_{A,X}$ and $C^{(2)}_{B,X}$ for source positions at and close to $\phi = \pi$ radians also would remove the energy associated with the scatterer. This is expected, because the scatterer is directly behind the source as seen from both stations $A$ and $B$; thus the contribution arrives simultaneously with the direct event from the source.

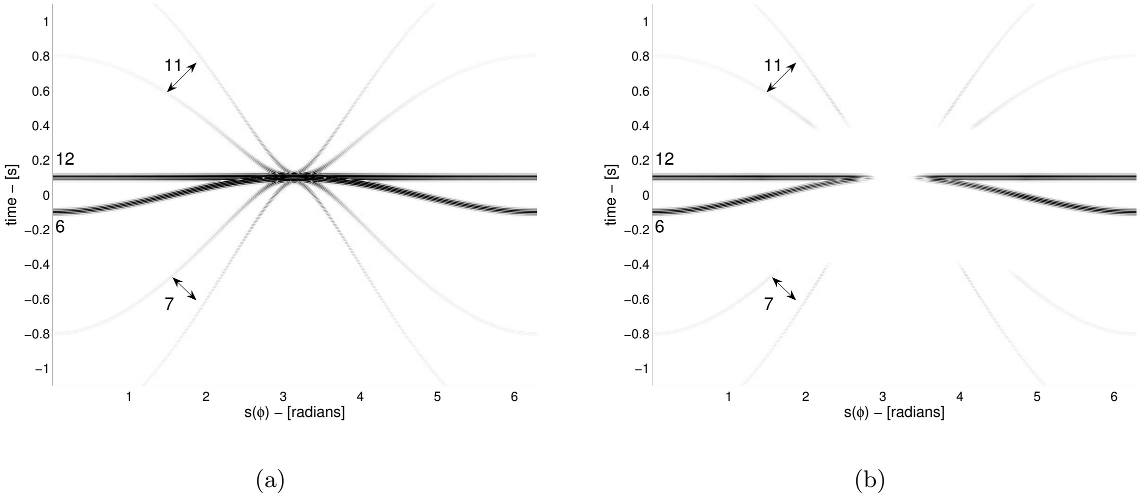


Figure 12: a) Gather showing time-domain equivalents of 15.6, 15.7, 15.10 and 15.11 after summing over auxiliary stations as a function of source position angle $\phi$. b) Gather of $|\tilde{C}^{3}_{B,A}(t)|$ as a function of source position angle $\phi$. [**ER**] sjoerd1/. corrC4ABSa,corrC4ABSb

## CONCLUSIONS

Using Green's functions under the Born approximation in a homogeneous medium with one scatterer, we show that $C^{(3)}_{B,A}$ constitutes 16 terms, that can be divided into 3 groups. The leading-order terms, group 1, are associated with the correlation of the direct waves recorded at the stations from a source that is generally not at a stationary-phase position. Thus evaluating $C^{(3)}_{B,A}$ directly does not improve the Green's function estimation. Instead we can remove the terms in group 1 from $C^{(3)}_{B,A}$ by muting $C^{(2)}_{A,X}$ and $C^{(2)}_{B,X}$. Group 2 contains the 4 leading-order terms in $C^{(3)}_{B,A}$ after muting $C^{(2)}_{B,X}$ and $C^{(2)}_{B,X}$. A stationary-phase ananysis of the 4 terms tells us that the scatterer must be aligned on ray paths between two stations, outside the station span.

Term 15.6 is non-stationary for all source postitions not aligned with the scatterer and stations $A$ and $B$. The non-stationarity is a function only of source position, not of auxiliary-station position. When we evaluate an ensemble average of multiple sources from different locations, term 15.6 will, in general, interfere destructively. Terms 15.7 and 15.10 are stationary when the source aligns with the scatterer, an auxiliary station and stations A and B. The non-stationarity is a function of source position and of auxiliary-station position. We can exploit this fact by using a network of auxiliary stations positioned randomly, such that if the source position is not at the stationary phase, the contribution from different auxiliary stations stack incoherently. Only term 15.11 remains stationary no matter where the source or auxiliary stations are positioned, so that any stacking of $C^{(3)}_{B,A}$ over auxiliary stations will enhance the contribution of this term.

An additional problem for the utilization of terms 15.6, 15.7 and 15.10 for the improvement of Green's function reconstruction is that for the source position for which these terms have stationary contributions at the correct traveltime, the contribution becomes indistinguishable from the leading-order contribution in $C^{(2)}_{A,X}$ and $C^{(2)}_{B,X}$ that must be removed. This means that stacking is the key to enhancing the contribution of term 15.11 and diminishing the contributions of terms 15.6, 15.7 and 15.10 to the EGF from $C^{(3)}_{B,A}$.

## APPENDIX

## WAVE EQUATION AND GREEN'S FUNCTION

We study the wave equation in an acoustic, linear, isotropic, time-invariant, sourceless, constant-density medium. The familiar wave equation for pressure $P = P(\mathbf{x}, t)$ is

$$\partial_i^2 P - c^{-2} \partial_t^2 P = 0, \tag{A-1}$$

where Einstein's summation convention is applied to lower-case subscripts; for 2D they are summed over 1 and 2. Temporal and spatial derivatives are denoted $\partial_t$ and $\partial_i$ respectively, where the subscripts denote time and spatial directions respectively. Under the constant-density assumption, the characteristic wave velocity $c = c(\mathbf{x})$ fully determines the medium.

### Fourier Transformations

The temporal Fourier transformation pairs of a time-domain function $F(t)$ and frequency-domain function $F(\omega)$ are defined as

$$F(\omega) = \int_{\infty}^{-\infty} F(t) \exp(-i\omega t)\, \mathrm{d}t, \tag{A-2}$$

$$F(t) = \frac{1}{2\pi} \int_{\infty}^{-\infty} F(\omega) \exp(i\omega t)\, \mathrm{d}\omega, \tag{A-3}$$

the particular Fourier-domain of the function $F$ is specified by the argument only.

### Frequency-domain Green's function in homogeneous media

Using the forward Fourier transformation equation A-2, the wave equation for pressure in a homogeneous medium with $c(\mathbf{x}) = c_0$ is written in the frequency-domain as

$$\partial_i^2 P + \frac{\omega^2}{c_0^2} P = 0. \tag{A-4}$$

The frequency-domain Green's function $G = G(\mathbf{x}, \mathbf{x}_s, \omega)$ is defined by introducing an impulsive point source acting at $t = 0$ and $\mathbf{x} = \mathbf{x}_s$ on the right-hand side of equation A-4 as follows:

$$\partial_i^2 G + \frac{\omega^2}{c_0^2} G = -\delta(\mathbf{x} - \mathbf{x}_s). \tag{A-5}$$

The Green's function solution for two-dimensional space, under the far field approximation can be obtained as

$$G(\mathbf{x}, \mathbf{x}_s, \omega) = \frac{1}{\sqrt{8\pi\omega c_0^{-1} |\mathbf{x} - \mathbf{x}_s|}} \exp\left(-i\left[\omega c_0^{-1} |\mathbf{x} - \mathbf{x}_s| + \frac{\pi}{4}\right]\right). \tag{A-6}$$

A source function is easily included by multiplication with the frequency-domain source function. A measurement, $P_A(\omega)$, at a station located at $\mathbf{x}_A$ of a source at $\mathbf{x}_s$ emitting a source function $s(\omega)$ is obtained as follows:

$$P_A = G(\mathbf{x}_A, \mathbf{x}_s, \omega) s(\omega). \tag{A-7}$$

The sources in this paper are simulated emitting zero-phase Ricker wavelets with center frequency $\omega_0$. The frequency-domain expression used is

$$s(\omega) = \frac{2\,\omega^2}{\sqrt{\pi}\,\omega_0^3} \exp\left(-\frac{\omega^2}{\omega_0^2}\right). \tag{A-8}$$

## Green's function in the Born Approximation

We are interested in the Green's function in an inhomogeneous medium. We assume the velocity can be split into a background velocity $c_0$ and a perturbation $\alpha(\mathbf{x})$ as $c^{-2}(\mathbf{x}) = c_0^{-2}\,[1 + \alpha(\mathbf{x})]$. Assuming the perturbation is confined inside some finite domain $\mathbf{D}_s$, the Green's function in the Born approximation can now be computed in terms of a Green's function computed in the background, $G_0$, medium as

$$\begin{aligned} G(\mathbf{x}, \mathbf{x}_s, \omega) &= G_0(\mathbf{x}, \mathbf{x}_s, \omega) + G_1(\mathbf{x}, \mathbf{x}_s, \omega), \quad \text{with} && \text{(A-9)} \\ G_1(\mathbf{x}, \mathbf{x}_s, \omega) &= \oint_{\mathbf{D}_s} G_0(\mathbf{x}, \mathbf{x}', \omega) \frac{\omega^2}{c_0^2} \alpha(\mathbf{x}') G_0(\mathbf{x}', \mathbf{x}_s, \omega) \mathrm{d}\mathbf{x}'. && \text{(A-10)} \end{aligned}$$

The Green's function in the background medium is computed using equation A-5 with $c = c_0$. When the medium consists of a homogeneous background with a series of $N$ scatters positioned at $\mathbf{x}_{c,1}, \mathbf{x}_{c,2}, \mathbf{x}_{c,3} ... \mathbf{x}_{c,N}$ with strength $\alpha_1, \alpha_2, \alpha_3, ...\alpha_N$, then $\alpha(\mathbf{x}) = \sum_{i=1}^{N} \delta(\mathbf{x} - \mathbf{x}_{c,i})\alpha_i$. Hence the Green's function $G_1$ in equation A-10 can be written as

$$G_1(\mathbf{x}, \mathbf{x}_s, \omega) = \sum_{i=1}^{N} G_0(\mathbf{x}, \mathbf{x}_{c,i}, \omega) \frac{\omega^2}{c_0^2} \alpha_i G_0(\mathbf{x}_{c,i}, \mathbf{x}_s, \omega). \tag{A-11}$$

# REFERENCES

Aki, K., 1957, Space and time spectra of stationary stochastic waves, with special reference to microtremors: Bulletin of the Earthquake Research Institute, **35**, 415–456.

Claerbout, J. F., 1968, Synthesis of a layered medium from its acoustic transmission response: Geophysics, **33**, 264–269.

———, 1976, Fundamentals of Geophysical Data Processing; with Applications to Petroleum Prospecting: Blackwell Scientific Publications.

de Ridder, S., 2008, Seismic interferometry versus spatial auto-correlation method on the regional coda of the NPE: Technical report, SEP-136.

Garnier, J. and G. Papanicolaou, 2009, Passive sensor imaging using cross correlations of noisy signals in a scattering medium: SIAM J. Imaging Sci., **2**, 396–437.

Lobkis, O. I. and R. L. Weaver, 2001, On the emergence of the greens function in the correlations of a diffuse field: J. Acoust. Soc. Am., **110**, 3011–3017.

Malcolm, A. E., J. A. Scales, and B. A. van Tiggelen, 2004, Retrieving the green function from diffuse, equipartitioned waves: Phys. Rev. E., **70**, 015601–1–015601–4.

Paul, A., M. Campillo, L. Margerin, E. Larose, and A. Derode, 2005, Empirical synthesis of time-asymmetrical Green functions from the correlation of coda waves: Journal of Geophysical Research (Solid Earth), **110**, B08302.1–B08302.13.

Roux, P., K. G. Sabra, W. A. Kuperman, and A. Roux, 2005, Ambient noise cross correlation in free space: Theoretical approach: JASA, **117**, 79–84.

Ruigrok, E. N., D. S. Dragonov, J. Thorbecke, J. R. van der Neut, and K. Wapenaar, 2008, Sampling and illumination aspects of seismic interferometry in horizontally layered media: 70th EAGE annual meeting, Rome, Italy., P277.

Sánchez-Sesma, F. J. and M. Campillo, 2006, Retrieval of the Green's function from cross correlation: The canonical elastic problem: Bull. Seism. Soc. Am., **96**, 1182–1191.

Sánchez-Sesma, F. J., J. A. Párez-Ruiz, M. Campillo, and F. Luzón, 2006, Elastodynamic 2D Green's function retrieval from cross-correlation: Canonical inclusion problem: Geophys. Res. Let., **33**, L13305–1–L13305–6.

Schuster, G. T., J. Yu, J. Sheng, and J. Rickett, 2004, Interferometric/daylight seismic imaging: Geophys. J. Int., **157**, 838–852.

Snieder, R., K. van Wijk, M. Haney, and R. Calvert, 2008, The cancellation of spurious arrivals in Green's function extraction and the generalized optical theorem: Phys. Rev. E, **78**, 036606–1–036606–8.

Snieder, R., K. Wapenaar, and K. Larner, 2006, Spurious multiples in seismic interferometry of primaries: Geophysics, **71**, SI111–SI124.

Snieder, R., K. Wapenaar, and U. Wegler, 2007, Unified Green's function retrieval by cross-correlation; Connection with energy principles: Phys. Rev. E, **75**, 036103–1–036103–14.

Snieder, R. K., 2004, Extracting the Green's function from the correlation of coda waves: A derivation based on stationary phase: Phys. Rev. E, **69**, 046610–1–046620–8.

Stehly, L., M. Campillo, B. Froment, and R. L. Weaver, 2008, Reconstructing Green's function by correlation of the coda of the correlation ($C^3$) of ambient seismic noise: J. Geophys. Res., **113**, B11306.1–B11306.10.

Stork, C. and S. Cole, 2007, Fixing the nonuniform directionality of seismic daylight interferometry may be crucial to its success: SEG Technical Program Expanded Abstracts, **26**, 2713–2717.

van der Neut, J. and A. Bakulin, 2008, The effects of time?gating and radiation correction on virtual source data: SEG, Expanded Abstracts, **27**, 2922–2926.

van der Neut, J. R., A. Bakulin, and K. Mehta, 2008, The effects of time-gating and radiation correction on virtual source data: 70th EAGE annual meeting, Rome, Italy., P274.

van Manen, D.-J., J. O. A. Robertsson, and A. Curtis, 2005, Modeling of wave propagation in inhomogeneous media: Phys. Rev. Lett., **94**, 164301–1–164301–4.

Wapenaar, K., 2004, Retrieving the elastodynamic Green's function of an arbitrary inhomogeneous medium by cross correlation: Phys. Rev. Lett., **93**, 254301–1 – 254301–4.

Wapenaar, K. and J. Fokkema, 2006, Green's function representations for seismic interferometry: Geophysics, **71**, SI33–SI46.

Wapenaar, K., J. van der Neut, and E. Ruigrok, 2008, Passive seismic interferometry by multidimensional deconvolution: Geophysics, **73**, A51–A56.

Weaver, R. L. and O. I. Lobkis, 2001, Ultrasonics without a source: Thermal fluctuation correlations at mhz frequencies.: Phys. Rev. Let., **87**, 134301.1–134301.4.

———, 2004, Diffuse fields in open systems and the emergence of the Green's function (L): J. Acoust. Soc. Am., **116**, 2731–2734.

# Measuring image focusing for velocity analysis

*Biondo Biondi*

## ABSTRACT

I present a method for extracting velocity information by measuring the focusing and unfocusing of migrated images. It measures image focusing by evaluating coherency across structural dips, in addition to coherency across aperture/azimuth angles. The inherent ambiguity between velocity and reflectors' curvature is directly tackled by introducing a curvature correction into the computation of the semblance functional that estimates image coherency. The resulting velocity estimator provides velocity estimates that are: 1) unbiased by reflectors' curvature, and 2) consistent with the velocity information that we routinely gather by measuring coherency over aperture/azimuth angles.

The application of the method to a 2D synthetic data set and a 2D field data set confirms that it provides consistent and unbiased velocity information. It also suggests that velocity estimates based on the new image-focusing semblance may be more robust and have higher resolution than estimates based on conventional semblance functionals. Preliminary tests on two 2D zero-offset synthetic data sets show that velocity information can be extracted from zero-offset data in presence of reflectors with arbitrary curvature, and not only in presence of point diffractors as previously published methods require.

## INTRODUCTION

The effects of migration velocity on the focusing and unfocusing of seismic images is obvious when observing depth migrated seismic images obtained with different migration velocities. Quantitative measures of image focusing could provide valuable information to velocity estimation. This information is particularly abundant in areas where reflectors have strong curvature or are discontinuous; such as in presence of faults, heavily folded geology, buried channels, uncomformities or rough salt/sediment interfaces. Figure 1 shows three images obtained by migrating the same prestack data set: the top panel (a) shows the image obtained with too low migration velocity, the middle panel (b) shows the image obtained with approximately the correct velocity, and the bottom panel (c) shows the image obtained with too high velocity. An interpreter could easily spot clear signs of undermigration in Figure 1a and of overmigration in Figure 1c. However, the definition of objective quantitative criteria to measure image focusing is challenging. Consequently, current practical methods for exploiting image-focusing information are based on subjective interpretation criteria instead of quantitative measurements (Sava et al., 2005; Wang et al., 2006).

If we were able to extract reliably quantitative focusing-velocity information from migrated images it could supplement the velocity information that we routinely extract by analyzing residual moveout along offsets (after common-offset migration) or aperture-angles (after angle-domain migration) axes. Velocity estimation would be enhanced by increasing resolution and decreasing uncertainties. It would be particularly useful to improve the interpretability of the final image and the accuracy of time-to-depth conversion in areas where the reflection aperture range is narrow, either because of unfavorable depth/offset ratio, or because of the presence of high-velocity geological bodies in the overburden (e.g. salt bodies) that deflect the propagating waves. In practice, velocity analysis based on image focusing is unlikely to replace conventional velocity analysis, but only to supplement it. Therefore, a method that measures image focusing should provide velocity estimates that are consistent with conventional methods.

Figure 1 illustrates some of the challenges of defining quantitative criteria to measure image focusing. The main challenge is related to the ambiguity between reflectors' curvature and their apparent focusing velocity. The section migrated with approximately the correct velocity (Figure 1b) shows several convex reflectors with strong curvature. These reflectors collapse into high-amplitude foci in the overmigrated section (Figure 1c). Criteria that have been previously proposed to measure image focusing, such as maximization of the power of the stack or minimization of image entropy (Harlan et al., 1984; De Vries and Berkhout, 1984; Stinson et al., 2005; Fomel et al., 2007), would wrongly rank the overmigrated image higher than the more accurate image. When in the subsurface we have high-curvature reflectors, but not infinite curvature reflectors, the minimum-entropy criterion would fail because it assumes the presence of point scatterers in the subsurface.

Fomel et al. (2007) propose to separate in the data space the diffractions originated from point scatterers before performing minimum-entropy velocity analysis. However, in complex geology this separation can be unreliable, mostly because reflections from curved reflectors may appear as diffractions. This potential source of errors is also well illustrated by the field-data example. Figure 2 shows the near-offset section of the data set used to generate the images shown in Figure 1. The diffraction-like hyperbolic events visible in this section were generated by the high-curvature reflectors discussed above. An application aimed to separate diffractions from other events could easily classify these events for diffractions and lead to biased velocity estimates.

This paper aims to overcome the shortcomings of current methods used to measure image focusing. It presents a new method that has two important characteristics: 1) it is unbiased by reflectors' curvature, and 2) it provides velocity information from image focusing that is consistent with the velocity information that we routinely extract from migrated images by analyzing their coherency along the reflection-aperture angle axes. The method is based on the *image-focusing* semblance functional I introduced in Biondi (2008b), where I generalized the conventional semblance functional used to measure image coherency along the aperture-angle axes by defining an image-focusing semblance functional that simultaneously measures image coherency along the structural-dip axes and the aperture-angle axes.

To remove the bias caused by reflectors' curvature, I explicitly take into account curvature by correcting its effects on image coherency along structural dips. Making curvature an explicit parameter of the velocity estimation does not necessarily resolves the fundamental problem of the ambiguity between the determination of reflectors' curvature and migration velocity. However, I show that it enables a consistent and unbiased velocity estimation that optimally uses the information contained in the data. In the last section of the paper, I present examples of image-focusing velocity analysis applied to two synthetic zero-offset data sets. These examples indicate that image-focusing analysis could automatically extract

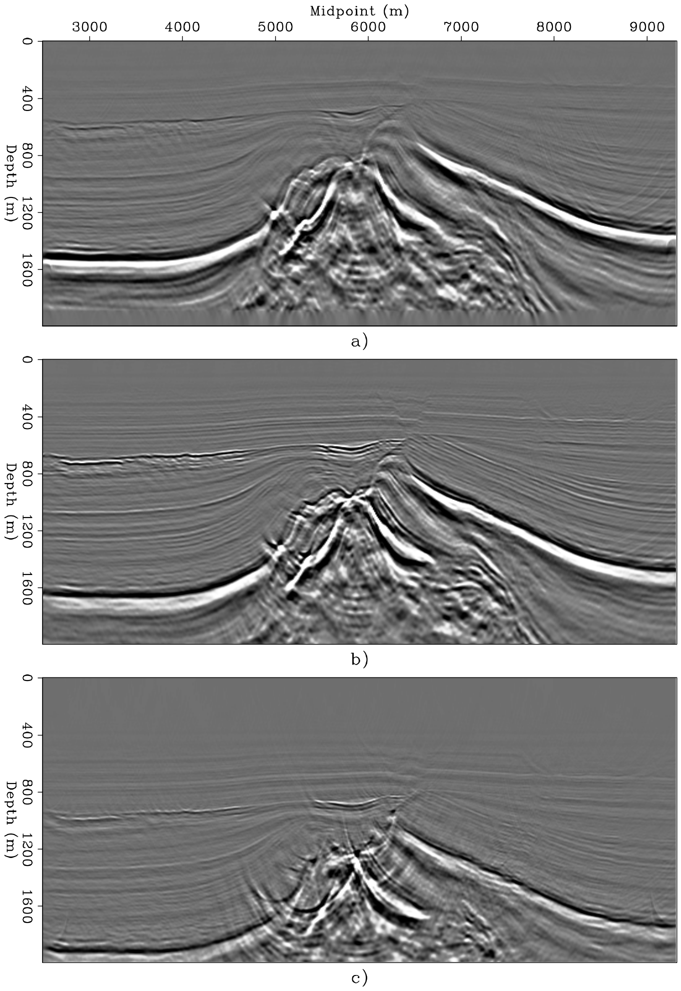


Figure 1: Three images obtained by prestack residual migration applied to the same prestack migration: the top panel (a) is undermigrated ($\rho = 0.8975$), the middle panel (b) s approximately well focused ($\rho = 1.01$), and the bottom panel (c) is overmigrated ($\rho = 1.2725$). [**CR**] biondo1/. ResMig-stack-overn

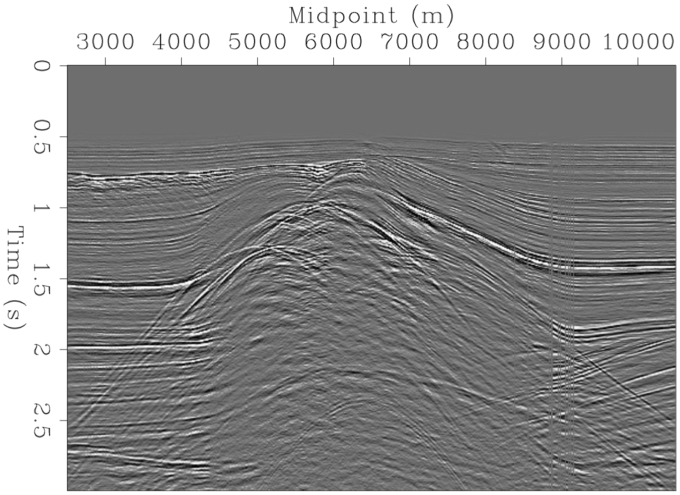


Figure 2: Near-offset section of the data set used to generate the images shown in Figure 1. [**CR**] biondo1/. Near-off-overn

useful velocity information from zero-offset data even when the reflectivity model contains curved reflectors with finite curvature.

The simultaneous image-coherency measurement along both the structural-dip axes and the aperture-angle axes of the curvature-corrected images, assures the consistency of the velocity information provided by the method. This consistency facilitates the interpretation of the results. Furthermore, it may improve the robustness of velocity estimation with respect to conventional angle-domain methods by automatically averaging the coherency computation along reflectors. At each point on a reflector, image coherency is measured for several dips in addition to the stationary dip. The inclusion of non-stationary dips is equivalent to averaging coherency measurements along the reflector, following both its dip and its curvature.

In this paper, I present results of the proposed method applied to 2D data. The computation of the image-focusing semblance functional could be easily generalized from 2D to 3D. In 2D, semblance is computed on 2D patches (structural dip and aperture angle); with 3D full-azimuth data, semblance would be computed on 4D patches (indexed by two structural dips, reflection aperture and reflection azimuth). The curvature correction is also easily generalizable from 2D to 3D. However, three parameters are required to define curvature in 3D: the two main curvatures along the principal axes, and the rotation of the principal axes with respect to the coordinate axes (Al-Dossary and Marfurt, 2006). I expect the nature of the ambiguity between velocity and curvature to be different between 2D and 3D. In both cases velocity is defined by one scalar parameter, whereas curvature is defined by three parameters in 3D.

## UNBIASED MEASURE OF IMAGE FOCUSING

In Biondi (2008b), I introduced a new semblance functional, that I dubbed *Image-focusing semblance*, aimed at quantitatively measuring image focusing simultaneously along the spatial directions and the reflection angle (or offset) axes. The underlying idea is to extend the conventional semblance evaluation by measuring image coherency also along the the structural-dip axes. However, the estimates provided by the image-focusing semblance presented in that report can be biased by reflectors' curvature. In this section, I modify the definition of the image-focusing semblance by explicitly exposing its dependency from the image local curvature. This enables a consistent evaluation of the image focusing across both the reflection-angle axis and the structural-dip axis and improves the interpretability of the results.

The starting point of my method is an ensemble of prestack images, $\mathbf{R}(\mathbf{x}, \gamma, \rho)$; these images are function of a spatial coordinate vector $\mathbf{x} = \{z, x\}$ (with $z$ depth and $x$ the horizontal location), the aperture angle $\gamma$, and a velocity parameter $\rho$. In the numerical examples that follow, the ensemble of prestack images is obtained by residual prestack migration in the angle domain as I presented in Biondi (2008a). The parameter $\rho$ is the ratio between the new migration velocity and the migration velocity used for the initial migration. The proposed method could be easily adapted to the case when residual prestack Stolt migration (Sava, 2003), or any other method that can efficiently generate ensembles of prestack images dependent on a velocity parameter, is used to compute $\mathbf{R}(\mathbf{x}, \gamma, \rho)$. Although, when using other methods to produce the ensemble $\mathbf{R}(\mathbf{x}, \gamma, \rho)$, the corrections equivalent to equations 5, 8 and 9 might be different.

To measure coherency along the structural dip $\alpha$, I first decompose the prestack image and create the dip-decomposed prestack image $\mathbf{R}\left(\mathbf{x},\gamma,\alpha,\rho\right)$. When using either choices of residual prestack migration discussed above, the decomposition can be efficiently performed in the Fourier domain during the residual prestack migration. If other methods are used to produce the ensemble of prestack images $\mathbf{R}\left(\mathbf{x},\gamma,\rho\right)$, the dip decomposition could as efficiently performed in the space domain by applying recursive filters (Fomel, 2002; Hale, 2007). Notice, that the dip-decomposed images I use as input have different kinematic characteristics than the ones described in Reshef and Rüger (2008), Landa et al. (2008), and Reshef (2008). They obtain dip-decomposed images by not performing the implicit summation over dips that is part of angle-domain Kirchoff migration (Audebert et al., 2002), whereas I decompose the migrated images.

In equation 5 in Biondi (2008b) I defined the 2D Image-focusing semblance as:

$$S_{(\gamma,\alpha)}\left(\mathbf{x},\rho\right)=\frac{\left[\sum_{\gamma}\sum_{\alpha}\mathbf{R}\left(\mathbf{x},\gamma,\alpha,\rho\right)\right]^{2}}{N_{\gamma}N_{\alpha}\sum_{\gamma}\sum_{\alpha}\mathbf{R}\left(\mathbf{x},\gamma,\alpha,\rho\right)^{2}},\tag{1}$$

where $N_\gamma$ and $N_\alpha$ are, respectively, the number of aperture angles and the number of dips to be included in the computation. The effective definition of the aperture-angle and the structural-dip ranges to be used in equation 1 is one of the practical challenges when applying the proposed method.

## Image curvature and residual migration

In presence of point diffractors, the semblance functional defined in expression 1 yields unbiased estimates of the velocity parameter $\rho$. However, when the curvature is finite, the dip components would not be aligned for the correct value of $\rho$ and the estimates would be biased. To remove this bias we can correct the dip-decomposed images for the presence of curvature. In Appendix A I show the simple derivation of this correction that amounts to the following spatial shift, $\Delta\mathbf{n}_{\rm Curv}$, along the normal to the structural dip,

$$\Delta\mathbf{n}_{\rm Curv}=\frac{\sin\left(\alpha-\bar{\alpha}\right)\tan\left(\alpha-\bar{\alpha}\right)}{2}R\mathbf{n},\tag{2}$$

where $R$ is the local radius of curvature, $\bar{\alpha}$ is the local dip and $\mathbf{n}$ is the vector normal to the dip $\alpha$ and directed towards increasing depth. Notice that the application of this correction requires the estimation of local dip $\bar{\alpha}$. To estimate the local dips, I used the Seplib program *Sdip* that implements a variant of the algorithms described by Fomel (2002).

Expression 2 can be used directly to create an ensemble to dip-decomposed images that are corrected for the local curvature $\mathbf{R}_{\rm Curv}\left(\mathbf{x},\gamma,\alpha,\rho,R\right)$. The image-focusing semblance can be computed on these images as:

$$S_{(\gamma,\alpha)}\left(\mathbf{x},\rho,R\right)=\frac{\left[\sum_{\gamma}\sum_{\alpha}\mathbf{R}_{\rm Curv}\left(\mathbf{x},\gamma,\alpha,\rho,R\right)\right]^{2}}{N_{\gamma}N_{\alpha}\sum_{\gamma}\sum_{\alpha}\mathbf{R}_{\rm Curv}\left(\mathbf{x},\gamma,\alpha,\rho,R\right)^{2}}.\tag{3}$$

However, the application of correction 2 can be quite expensive unless it is performed together with residual migration. Furthermore, precomputing the curvature-corrected images would further increase the dimensionality of the image space, creating obvious problems for handling the resulting bulky data sets. Fortunately, when the ensemble of the dip-decomposed images $\mathbf{R}\left(\mathbf{x},\gamma,\alpha,\rho\right)$ are the result of residual prestack migration, the curvature correction can be efficiently computed during the evaluation of the semblance functional 3. Correction 2 becomes a simple interpolation along the residual velocity parameter $\rho$, as a function of the aperture angles and dips.

To derive the interpolating function, I first recall the expression of residual migration in Biondi (2008a):

$$\Delta\mathbf{n}_{\mathrm{Rmig}} = \left(\rho_{\mathrm{new}} - \rho_{\mathrm{old}}\right)\frac{\cos\alpha}{\left(cos^2\alpha - \sin^2\gamma\right)} z_0\mathbf{n}, \tag{4}$$

where $\Delta\mathbf{n}_{\mathrm{Rmig}}$ is the normal shift applied by residual migration, $\rho_{\mathrm{new}}$ is the value of $\rho$ after residual migration and $\rho_{\mathrm{old}}$ is the value of $\rho$ before residual migration, which is usually set to be equal to one. The parameter $z_0$ is a constant that is equal to the depth for which the residual migration in 4 is exact.

Equating the normal shift in 4 with the normal shift in 2 and solving for $\rho_{\mathrm{new}}$ we obtain

$$\rho_{\mathrm{new}} = \rho_{\mathrm{old}} + \frac{\sin\left(\alpha-\bar{\alpha}\right)\tan\left(\alpha-\bar{\alpha}\right)\left(\cos^2\alpha - \sin^2\gamma\right)}{2\cos\alpha z_0}R. \tag{5}$$

In this case, $\rho_{\mathrm{new}}$ is the $\rho$ of the images from which the data are interpolated from, and $\rho_{\mathrm{old}}$ is the $\rho$ of the images after correction; that is,

$$\mathbf{R}_{\mathrm{Curv}}\left(\mathbf{x},\gamma,\alpha,\rho_{\mathrm{old}},R\right) = \mathbf{R}\left[\mathbf{x},\gamma,\alpha,\rho_{\mathrm{new}}\left(\rho_{\mathrm{old}},\gamma,\alpha,\bar{\alpha},R\right)\right]. \tag{6}$$

### Image curvature and residual migration in the pseudo-depth domain

The interpolation defined by 5 depends in a non-straightforward manner from both angles $\gamma$ and $\alpha$, as well as from the estimate of the local dip $\bar{\alpha}$. Although, this is the relationship I used in practice for the examples in this paper, I will now analyze one of its variants that is simpler and thus it helps to better understand the relationship between image curvature and residual migration parameter.

I start from redefining residual migration in the pseudo-depth domain $\tilde{z} = z/\rho$ (Sava, 2004). In this domain, the focusing/unfocusing effects of residual migration are better separated from its mapping effects than in the conventional depth domain. In the pseudo-depth domain, normal-incidence images of flat reflectors are not shifted by residual migration. The expression of residual migration 4 becomes:

$$\Delta\mathbf{n}_{\mathrm{Rmig}} = \left(\rho_{\mathrm{new}} - \rho_{\mathrm{old}}\right)\left[\frac{\cos\alpha}{\left(\cos^2\alpha - \sin^2\gamma\right)} - \cos\alpha\right] z_0\mathbf{n}, \tag{7}$$

and the expression of curvature correction 5 becomes:

$$\rho_{\mathrm{new}} = \rho_{\mathrm{old}} + \frac{\sin^2\left(\alpha-\bar{\alpha}\right)\left(\cos^2\alpha - \sin^2\gamma\right)}{2z_0\cos\left(\alpha-\bar{\alpha}\right)\cos\alpha\left(\sin^2\alpha + \sin^2\gamma\right)}R, \tag{8}$$

that also does not provide a straightforward relationship between the input and output $\rho$s. Furthermore, it becomes singular for the flat dip component ($\alpha = 0$) of normal incidence images ($\gamma = 0$). Its use is thus more cumbersome than the use the equivalent expression in the depth domain (equation 5).

However, in the special case of events that are locally flat ($\bar{\alpha} = 0$) and are imaged at normal-incidence (i.e. $\gamma = 0$), this expression simplifies into:

$$\rho_{\text{new}} = \rho_{\text{old}} + \frac{R}{2z_0}. \tag{9}$$

In this case, the curvature correction becomes independent from the dip $\alpha$. It only remaps the image from $\rho_{\text{new}}$ to $\rho_{\text{old}}$ and thus does not affect the coherency along the dip direction of the dip-decomposed images. There is perfect ambiguity between the residual migration parameter $\rho$ and the reflector radius of curvature $R$.

## SYNTHETIC-DATA EXAMPLE

To illustrate the proposed method, I first present its application to a synthetic data set. The model is a medium with constant slowness of .5 s/km and a single reflector with sinusoidal shape. This reflector is shown in Figure 3. I modeled a prestack data set with offsets between -1.5 kilometers and 1.5 kilometers. I then migrated the data with both the correct slowness and a high slowness of .525 s/km; that is, 105% the correct slowness.

I dip-decomposed the image obtained with the correct slowness at zero-subsurface offset and corrected it for curvature according to expression 5. Figure 4a shows the dip-decomposed image at the midpoint of one of the bottoms of the sinusoid ($x$=4.250 km). Because of the curvature, the dips are not aligned and the event is frowning down. Figure 4b shows the panel in Figure 4a corrected for image curvature by applying the shift defined in expression 2. I selected the radius of curvature to be equal to -90 meters. This is consistent with the analytical radius of curvature of the sinusoidal reflector at the same location of -86 meters. I set the reflector local dip to be zero; that is, I set $\bar{\alpha} = 0$ in expression 2.

The panels shown in Figure 5 are equivalent to the panels shown in Figure 4, except that the midpoint location is at one of the tops of the sinusoid ($x$=4.750 km). At this location the curvature is positive and thus the uncorrected dip panel (Figure 5a) smiles upward. The corrected panel (Figure 5b) corresponds to a positive radius of curvature of 90 meters.

I computed the conventional semblance over aperture angle and the proposed image-focusing semblance from the migrated image obtained with the high slowness. Figure 6 compares the semblance fields computed by the conventional semblance functional that measures coherency only over aperture angles (Figure 6a), with the semblance cube computed by the proposed image-focusing semblance functional that measures coherency over both aperture angles and structural dips (Figure 6b). The figure shows the semblance fields at $x$=4.750 km, that is at one of the local top of the sinusoidal reflector. The $\rho$-range is the same ($0.984 \leq \rho \leq 1.134$) for the two panels in the figure. The semblance peak is more sharply defined as a function of the $\rho$ parameter in the result of the new image-focusing functional (right face in Figure 6b) than in the result of conventional method (Figure 6a).

Notice that the semblance peak is located at longer radius of curvature ($R$=125 meters) than the actual radius of curvature of the reflector ($R$=86 meters), because residual migration in the angle domain is not exact and does not fully correct for the reflector curvature.

This error is inconsequential for the proposed method since the aim is to better estimate $\rho$ not $R$.

## FIELD-DATA EXAMPLE

I applied the proposed method to a 2D marine line extracted from a 3D data set. The images shown in Figure 1 were produced from this 2D line. I will focus on the analysis of the results for a small window of the image that contains both convex and concave reflectors. In contrast with the previous synthetic-data example, I performed the curvature correction defined in equation 5 by using a field of local dips estimated numerically. I applied the Seplib program *Sdip* to the ensemble of sections obtained by stacking along the aperture-angle axis the residual migrated images for each value of $\rho$.

Figure 7a shows the migrated stack of the analysis window for a particular choice of the $\rho$ parameter ($\rho$=1.04) that maximizes flatness in the aperture-angle gather at the midpoint location corresponding to the black line superimposed onto the stack; that is for $x$=5.646 km. Figure 7b shows the aperture-angle gather and Figure 7c the corresponding semblance panel.

Starting from the prestack images, I computed dip-decomposed images that are function of both the aperture angle $\gamma$ and the structural dip $\alpha$. Figure 8 shows the 3D cube of the dip-decomposed image at the same midpoint location as the previous figure; that is for $x$=5.646 kilometers. The convex reflector of interest, at depth of 950 meters, shows an upward-smiling moveout in the structural-dips panel, consistently with the result observed when discussing the synthetic-data example in the previous section. Figure 9 displays the image-focusing semblance cube at that same midpoint location. The left panel in the cube displays semblance as a function of depth and radius of curvature ($R$) at $\rho$=1.04; the right panel displays semblance as a function of depth and $\rho$ at $R$=125 meters. The location of the semblance peak in the cube at depth of 950 meters is consistent with the location of the semblance peak in the conventional $\rho$ scan shown in Figure 7c. The semblance peak in the image-focusing cube is slightly tighter than in the conventional scan, but the differences are not substantial.

Figures 10-12 shows similar analysis of the migrated images presented above, but at the midpoint location corresponding to the reflector with negative curvature; that is for $x$=5.539 kilometers. The reflector is locally dipping with negative dip of approximately 45 degrees. The stationary point in the dip-decomposed image shown in the right panel of Figure 11 is located at that value of the structural dip, and it is frowning instead of smiling because of the negative local curvature. The value of $\rho$ for which the reflector is the flattest along the aperture-angle axis ($\rho$=.95), is substantially lower than for the previous reflector ($\rho$=1.04). This substantial difference in apparent velocity, notwithstanding the proximity of the two midpoint locations, is probably related to the fact that the wavefronts that illuminate the two events propagate through different zones of the velocity model due to the dip of the second reflector.

The semblance peak in the image-focusing cube (right panel in Figure 12) is now substantially better defined than in the conventional semblance panel shown in Figure 10c, suggesting a potential resolution benefit for velocity estimation. Further analysis of this potential benefit is needed before drawing definitive conclusions.

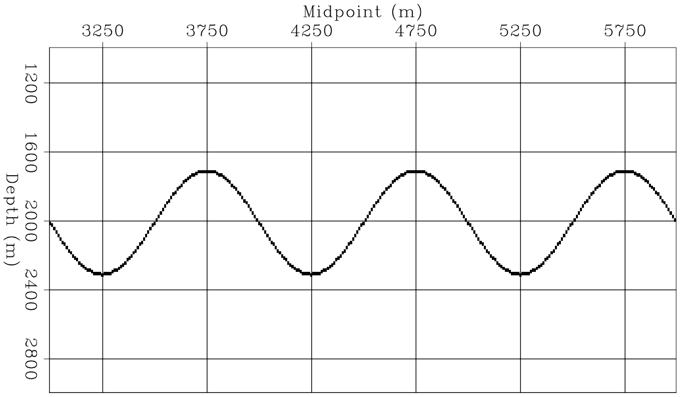


Figure 3: Sinusoidal reflector used to generate the synthetic prestack data set. [**ER**] biondo1/. Refl-sinus-overn

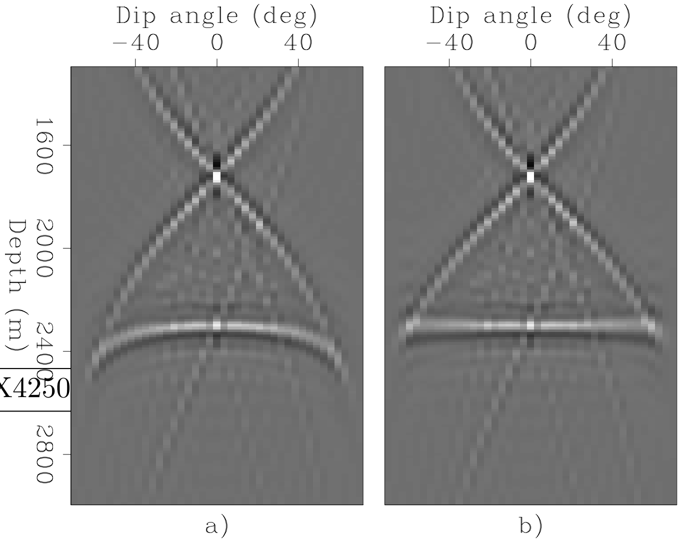


Figure 4: The dip-decomposed images at the midpoint of one of the bottoms of the sinusoidal reflector ($x$=4.250 km): without curvature correction (panel a) and after curvature correction (panel b). [**CR**] biondo1/. ResMig-short-dip-curv-all-X4250

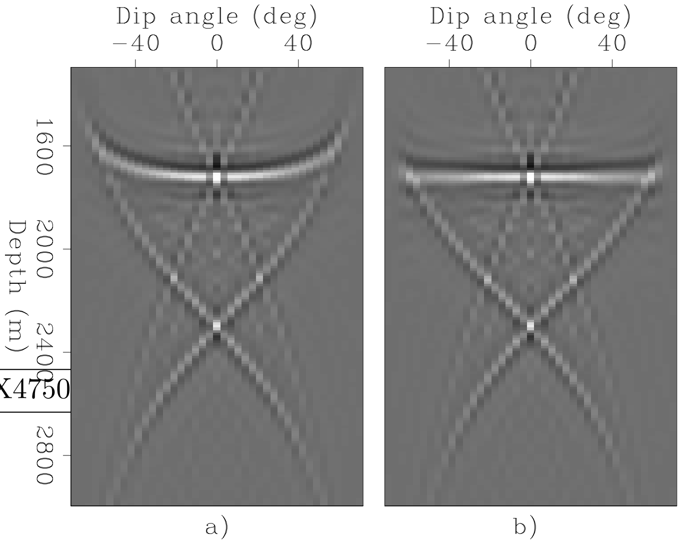


Figure 5: The dip-decomposed images at the midpoint of one of the tops of the sinusoidal reflector ($x$=4.750 km): without curvature correction (panel a) and after curvature correction (panel b). [**CR**] biondo1/. ResMig-short-dip-curv-all-X4750

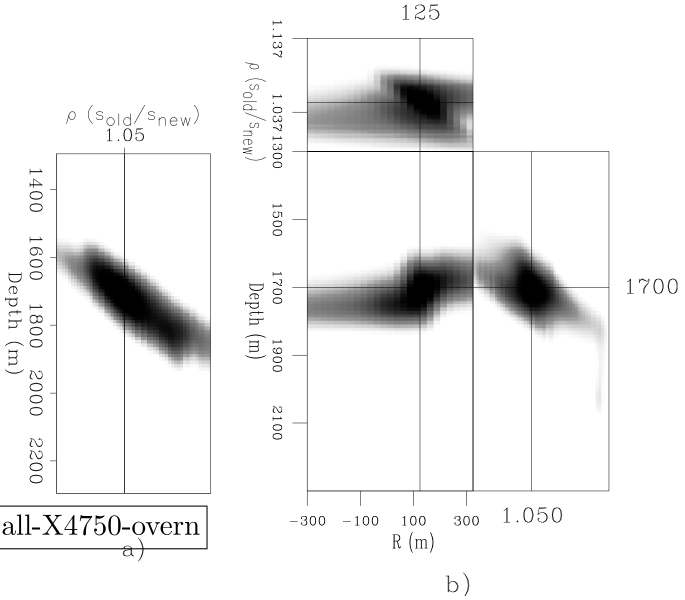


Figure 6: Comparison of the semblance fields computed by the conventional semblance functional that measures coherency only over aperture angles (panel a), with the semblance cube computed by the proposed image-focusing semblance functional that measures coherency over both aperture angles and structural dips (panel b). The figure shows the semblance fields at $x$=4.750 km. [**CR**] biondo1/. Wind-Sembl-short-ang-dip-all-X4750-overn

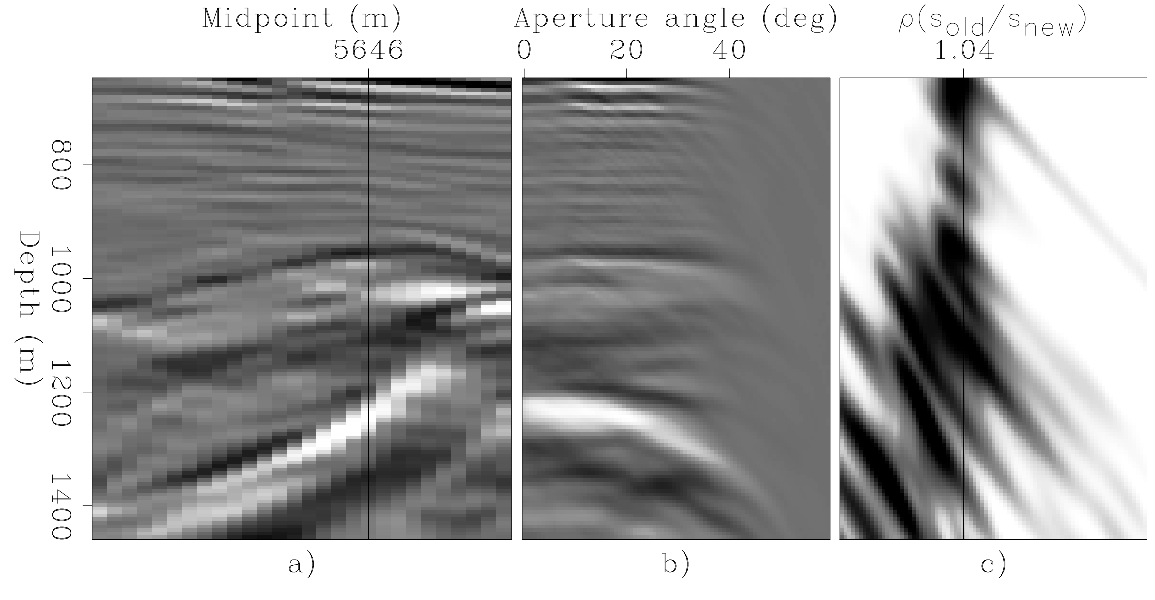


Figure 7: The migrated stack of the analysis window for $\rho$=1.04 (panel a), the aperture-angle gather at $x$=5.5646 km for $\rho$=1.04 (panel b), and the aperture-angle semblance section at $x$=5.646 km (panel c). [**CR**] biondo1/. Wind-ResMig-all-X5646-overn

Another potential advantage of explicitly taking into account, and correcting for, reflectors' curvature in the semblance analysis, is that it automatically enables the simultaneous measurements of coherency for several structural dips, in addition to the stationary dip, at each analysis point. The semblance measurements are thus automatically averaged along the reflector, following both its local dip and its local curvature. To test this hypothesis, I computed a modified version of the conventional semblance functional along the aperture-angle axis according to the following expression:

$$S_\gamma\left(\mathbf{x},\rho,R\right)=\frac{\sum_\alpha\left[\sum_\gamma\mathbf{R}_{\rm Curv}\left(\mathbf{x},\gamma,\alpha,\rho,R\right)\right]^2}{\sum_\alpha N_\gamma\sum_\gamma\mathbf{R}_{\rm Curv}\left(\mathbf{x},\gamma,\alpha,\rho,R\right)^2}, \tag{10}$$

that averages both numerator and denominator along the structural-dip axis. Figure 13 compares the result of conventional semblance with the result of computing the semblance functional defined in 10. Figure 13c displays conventional semblance, and it is the same panel shown in Figure 10c. Figure 13a displays the constant $\rho$ section of the semblance cube computed using 10 and Figure 13b displays the constant curvature ($R$=-75 meters) section of this semblance cube. Although both panel b) and panel c) are computed by measuring coherency only along the aperture-angle axis, the semblance peak corresponding to the concave reflector is clearly better focused and more easily pickable in panel b) than in panel c). This example suggests that there is an advantage on averaging semblance over structural dips. On the other hand, there is the additional cost of computing the dip-decomposed images and the additional complexity of picking a higher dimensionality semblance cube.

## ZERO-OFFSET SYNTHETIC-DATA EXAMPLE

The synthetic-data and field-data examples discussed in the two previous sections applied the image-focusing semblance to prestack data sets, where useful velocity information is provided by the data redundancy over offsets. In this section, I present experiments on two simple zero-offset synthetic data sets. The only velocity information contained in the migrated images obtained from zero-offset data is the focusing and unfocusing of reflections.

Figure 14 shows the reflectors' geometry assumed to model the two synthetic data sets. I modeled the first data set assuming a "cloud" of point diffractors (panel a), whereas I modeled the second data set assuming a "cloud" of convex reflectors (panel b). In both cases the velocity was assumed to be constant and equal to 2 km/s and the data were migrated assuming a high slowness of .5125 s/km; that is, 102.5% of the correct slowness.

Figure 15 summarizes the main result of this section. All three panels show the image-focusing semblance spatially averaged in an inner rectangle of the image space defined by the following inequalities along the depth axis: 1.850 km $\leq$ z $\leq$ 2.150 km, and by the following inequalities along the midpoint axis: 4.875 km $\leq$x$\leq$ 5.125 km. The panel shows the average semblance as a function of the velocity parameter $\rho$ and the radius of curvature $R$. Figure 15a shows the result corresponding to the point diffractors and Figure 15b shows the result corresponding to the convex reflectors. In both cases, I applied the curvature correction defined in 5 by using a field of local dips ($\bar{\alpha}$) estimated numerically by applying the Seplib program *Sdip* to the ensemble of residual migrated images for each value of $\rho$.

The important observation supported by this figure is that, in both Figure 15a and Figure 15b, the semblance energy is concentrated in a relatively narrow interval that includes the correct value of $\rho$; that is $\rho = 1.025$. This result indicates that we can extract useful velocity information from zero-offset data by using the image-focusing semblance.

The third panel in Figure 15, shows the semblance average computed from the images of the convex reflectors when I applied the curvature correction defined in 5 by using a constant local dip equal to zero; that is, when I uniformly set $\bar{\alpha} = 0$. As predicted by expression 9, there is strong ambiguity between the reflector curvature and the velocity parameter and the semblance is high also for values of $\rho$ that are far away from the correct one. We can consequently conclude that the velocity information contained in panel a) and b) derives from the inconsistency between the focusing information extracted using the image-focusing semblance and the local dip estimation. This inconsistency occurs when the image is sufficiently unfocused that the local dip estimation becomes unreliable. The following figures illustrate this concept.

Figures 16–20 provide a graphical explanation of the results shown in Figure 15. Figure 16 shows the migrated images of the point-diffractors data corresponding to the values of $\rho$ at the edges of the semblance peak in Figure 15a. The inner rectangle delimited by the grid superimposed to the images shows where the semblance is spatially averaged to produce the results shown in Figure 15. The image in Figure 16a is undermigrated and corresponds to $\rho = 1.0125$, whereas the image in Figure 16b is overmigrated and corresponds to $\rho = 1.0375$. In both of these images the unfocusing starts to cause crossing of events in the inner rectangle delimited by the grid superimposed to the images. The local dips are then multivalued and the automatic estimation of the local dips becomes unreliable and inconsistent with the more global behavior of the dips. Therefore, outside the interval $1.0125 \leq \rho \leq 1.0375$ the semblance average drops substantially in value.

Similar behavior is displayed by the migrated images of the convex-reflectors data corresponding to the values of $\rho$ at the edges of the semblance peak in Figure 15a. These images are shown in Figure 17, and correspond to $\rho = 1.01$ (Figure 17a), and to $\rho = 1.07$ (Figure 17b). In this case, the $\rho$ range is wider than in the previous case because the convex-reflectors' density is lower than the point-diffractors' density, and thus a larger velocity error is needed before poorly focused events start crossing.

Figures 18–20 show sections cut through the image-focusing semblance cubes at constant value of $\rho$ and $R$ before spatial averaging. Figure 18a shows semblance for the point-diffractors data for $\rho = 1.025$ and $R = 0$ meters; that is, the values of $\rho$ and $R$ for which the data are best focused. Figure 18a shows semblance for $\rho = 1.0125$ and $R = 40$ meters. This value of $\rho$ is the one corresponding to the undermigrated image in Figure 16a. Because of undermigration, the image from the point diffractors appears to have a positive radius of curvature approximately equal to 40 m. However, because of inconsistency between the focusing information and the local dip estimation, semblance is in average lower in the panel on the right than in the panel on the left.

Similar behavior is displayed by the image-focusing semblance cubes computed from the images of the convex-reflectors data. We find the "best focused" semblance panel (Figure 19a) still at infinite curvature ($R = 0$ meters), but at a wrong value of $\rho$; that is, at $\rho = 1.04$. However, the important result is that the interval with relative high semblance still includes the correct value of $\rho$. The section shown in Figure 19b corresponds to undermigrated image shown in Figure 19b, and it is taken for $\rho = 1.01$ and $R = 120$ meters. The apparent curvature is lower than for the point diffractors because the actual curvature of the reflector is lower.

Finally, Figure 20 shows sections through the image-focusing semblance cubes for the convex-reflectors data when the local dip is uniformly set equal to zero. These panels correspond to the average semblance shown in Figure 15c, and are sections taken for the same values of $\rho$ and $R$ as the sections shown in Figure 17. Because of the ambiguity between velocity and curvature, both panels show well-focused and high value semblance peaks.

## DISCUSSION AND CONCLUSIONS

Using image focusing and unfocusing for velocity estimation has been for long time an elusive goal in reflection seismology. The main challenge is the ambiguity between image focusing and reflectors' curvature. Consequently, previously published methods had to rely on strong assumptions on reflectors' curvature, such as assuming that reflections were generated by point diffractors; that is, by infinite-curvature reflectors. I present a method that does not rely on this assumption because it explicitly takes into account of reflectors' curvature when measuring image focusing.

The synthetic-data example I present in the third section and the field-data example I present in the fourth section show that the method may provide higher resolution and more robust velocity information than conventional methods based on measuring image coherency only along the aperture-angle axes (or the offset axes when constant-offset migration is performed.) Furthermore, the proposed method extract image-focusing information from prestack data that is consistent with the velocity information that we routinely extract by measuring image coherency along the aperture-angle axes.

The two zero-offset synthetic-data examples I show in the last section suggest that useful velocity information can be extracted from zero-offset data. Images that were migrated with approximately the correct velocity have no crossing events and thus the local dip information measured from these images is consistent with the focusing information measured by the image-focusing semblance functional. When the migration velocity is far from the correct one, the migrated images have a lot of crossing events. The local-dips information measured from these images is unreliable and inconsistent with the focusing information measured by the image-focusing semblance functional. These results suggest that it might be useful to estimate the reflectors' curvature by local curvature estimators (Al-Dossary and Marfurt, 2006) and use this information to further constrain the velocity estimates obtained by applying the image-focusing semblance proposed in this paper.

## APPENDIX A

## CURVATURE CORRECTION

This appendix derives the expression for the curvature correction presented in the main text in 2. The derivation is extremely simple and based on the geometry sketched in Figure 21.

The reflector is approximated with a parabola with radius of curvature $R$ at its vertex. In the rotated coordinates system $(z', x')$ the equation of the parabola is

$$z' = \frac{x'^2}{2R}. \tag{A-1}$$

The shift $\Delta z'$ that moves a tangent to the parabola to the vertex is equal to

$$\Delta z' = \tan^2 \alpha' \frac{R}{2}, \tag{A-2}$$

and consequently the normal shift $\Delta n$ is equal to

$$\Delta n = \frac{\cos \alpha' \tan^2 \alpha'}{2} R = \frac{\sin \alpha' \tan \alpha'}{2} R. \tag{A-3}$$

The coordinate system $(z', x')$ is rotated by $\bar{\alpha}$ with respect to $(z, x)$. Removing that rotation is equivalent to set $\alpha' = \alpha - \bar{\alpha}$; performing this substitution in the previous equation, I obtain the correction in 2; that is,

$$\Delta n = \frac{\sin (\alpha - \bar{\alpha}) \tan (\alpha - \bar{\alpha})}{2} R. \tag{A-4}$$

## REFERENCES

Al-Dossary, S. and K. J. Marfurt, 2006, 3d volumetric multispectral estimates of reflector curvature and rotation: Geophysics, **71**, P41–P51.

Audebert, F., P. Froidevaux, H. Rakotoarisoa, and J. Svay-Lucas, 2002, Insights into migration in the angle domain: SEG Technical Program Expanded Abstracts, **21**, 1188–1191.

Biondi, B., 2008a, Automatic wave-equation migration velocity analysis: 2008, **134**, 65–77.

———, 2008b, An image-focusing semblance functional for velocity analysis: SEP-Report, **136**, 43–54.

De Vries, D. and A. J. Berkhout, 1984, Velocity analysis based on minimum entropy: Geophysics, **49**, 2132–2142.

Fomel, S., 2002, Applications of plane-wave destruction filters: Geophysics, **67**, 1946–1960.

Fomel, S., E. Landa, and M. T. Taner, 2007, Poststack velocity analysis by separation and imaging of seismic diffractions: Geophysics, **72**, U89–U94.

Hale, D., 2007, Local dip filtering with directional laplacians: CWP-Report, **567**.

Harlan, W. S., J. F. Claerbout, and F. Rocca, 1984, Signal/noise separation and velocity estimation: Geophysics, **49**, 1869–1880.

Landa, E., S. Fomel, and M. Reshef, 2008, Separation, imaging, and velocity analysis of seismic diffractions using migrated dip-angle gathers: SEG Technical Program Expanded Abstracts, **27**, 2176–2180.

Reshef, M., 2008, Interval velocity analysis in the dip-angle domain: Geophysics, **73**, VE353–VE360.

Reshef, M. and A. Rüger, 2008, Influence of structural dip angles on interval velocity analysis: Geophysics, **73**, U13–U18.

Sava, P., 2004, Migration and velocity analysis by wavefield extrapolation: PhD thesis, Stanford University.

Sava, P. C., 2003, Prestack residual migration in frequency domain: Geophysics, **68**, 634–640.

Sava, P. C., B. Biondi, and J. Etgen, 2005, Wave-equation migration velocity analysis by focusing diffractions and reflections: Geophysics, **70**, U19–U27.

Stinson, K., E. Crase, W.-K. Chan, and S. Levy, 2005, Optimized determination of migration velocities: Recorder, **30**, 5–6.

Wang, B., V. Dirks, P. Guillaume, F. Audebert, and D. Epili, 2006, A 3d subsalt tomography based on wave-equation migration-perturbation scans: Geophysics, **71**, E1–E6.

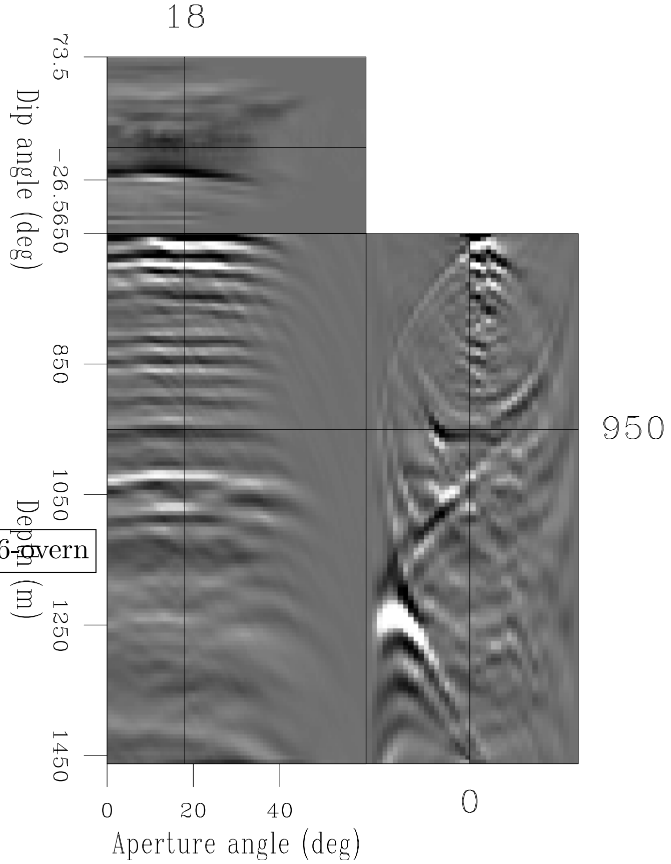


Figure 8: The dip-decomposed image at $x$=5.646 kilometers. The convex reflector of interest, at depth of 950 meters, shows an upward-smiling moveout in the structural-dips panel. [**CR**] biondo1/. Wind-ResMig-dip-ang-X5646-overn

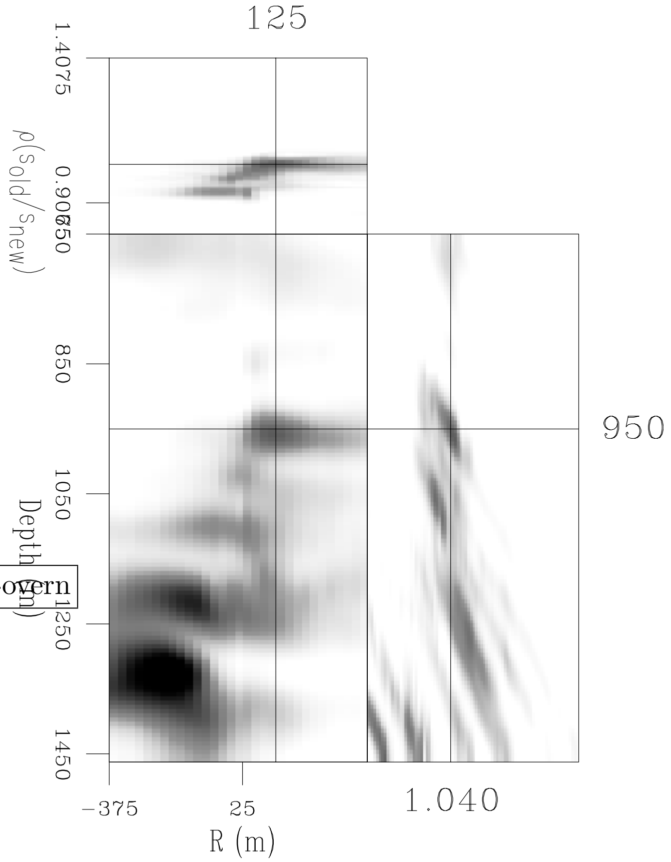


Figure 9: The image-focusing semblance cube at $x$=5.646 kilometers. The location of the semblance peak in the cube at depth of 950 meters is consistent with the location in the conventional $\rho$ scan shown in Figure 7c. The peak is slightly tighter than in the conventional scan. [**CR**] biondo1/. Wind-Sembl-dip-ang-X5646-overn

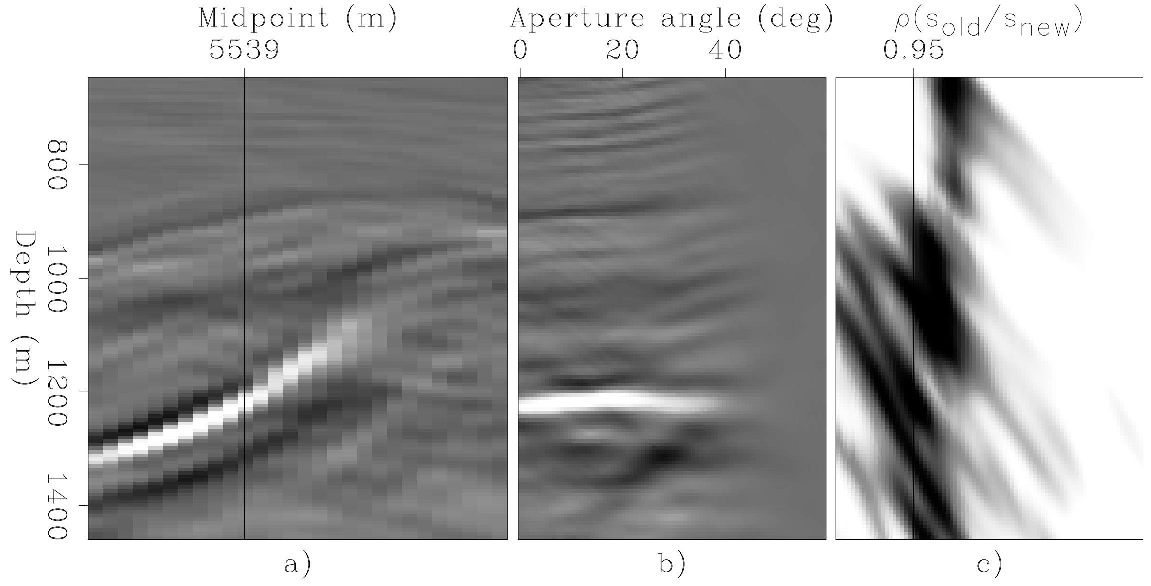


Figure 10: The migrated stack of the analysis window for $\rho$=0.95 (panel a), the aperture-angle gather at $x$=5.539 km for $\rho$=0.95 (panel b), and the aperture-angle semblance section at $x$=5.539 km (panel c). [**CR**] biondo1/. Wind-ResMig-all-X5539-overn

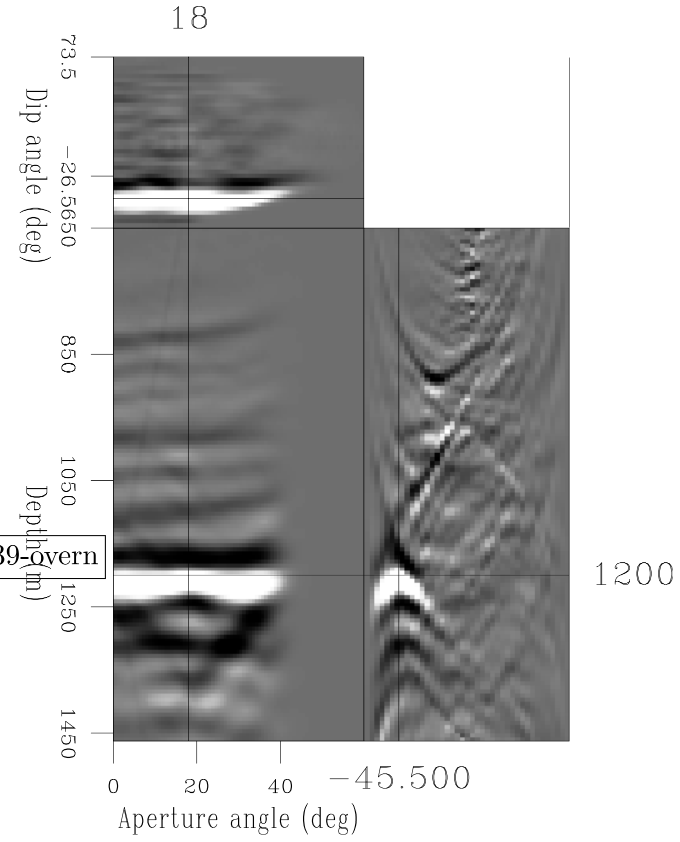


Figure 11: The dip-decomposed image at $x$=5.539 kilometers. The concave reflector of interest, at depth of 1200 meters and dip of 45 degrees, shows a downward-frowning moveout in the structural-dips panel. [**CR**] biondo1/. Wind-ResMig-dip-ang-X5539-overn

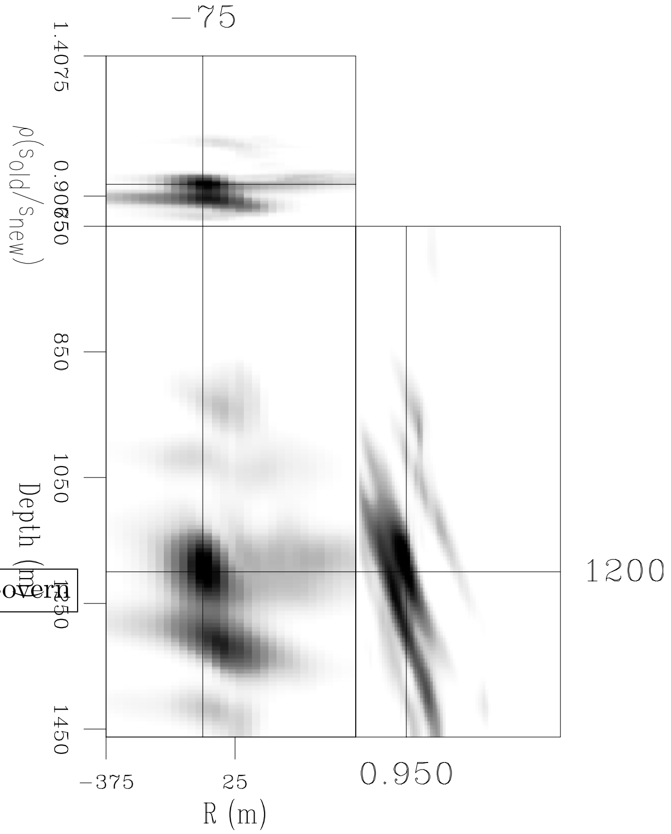


Figure 12: The image-focusing semblance cube at $x$=5.539 kilometers. The location of the semblance peak in the cube at depth of 1200 meters is consistent with the location in the conventional $\rho$ scan shown in Figure 10c, but is substantially better defined than in the conventional scan. [**CR**] biondo1/. Wind-Sembl-dip-ang-X5539-overn

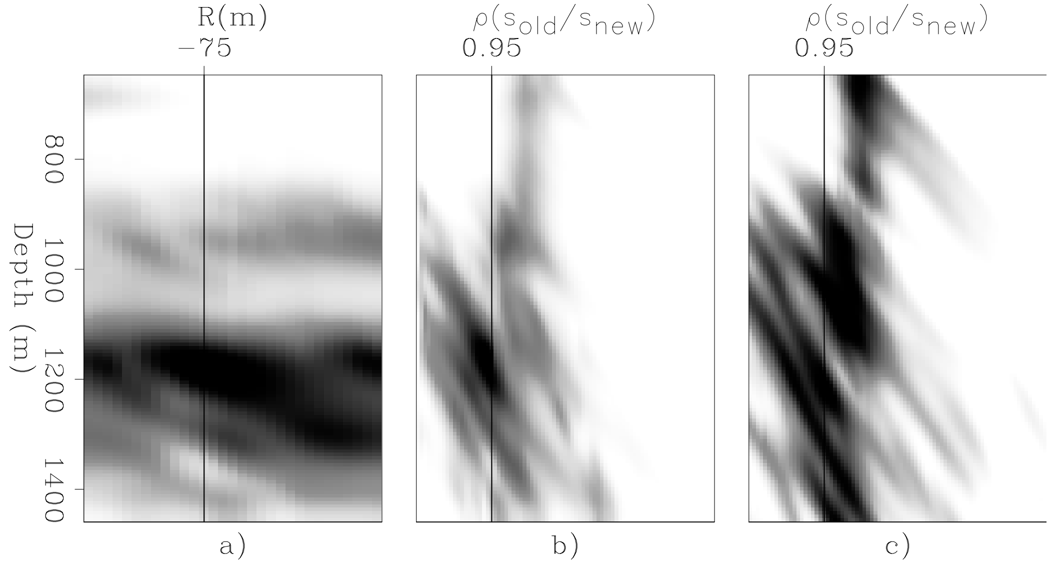


Figure 13: Comparison of the result of computing the semblance functional defined in 10 (panels a and b) with the result of conventional semblance (panel a), at $x$=5.539 kilometers. The semblance peak corresponding to the concave reflector is clearly better focused and more easily pickable in panel b) than in panel c). [**CR**] biondo1/. Wind-Sembl-curv-all-X5539-overn

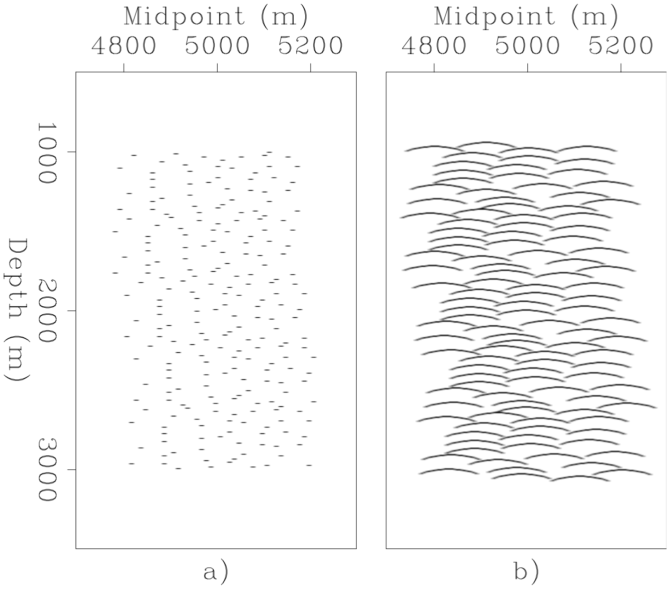


Figure 14: Reflectors' geometry assumed to model the two zero-offset synthetic data sets I used to test the proposed image-focusing velocity-estimation method: a) a "cloud" of point diffractors, and b) a "cloud" of convex reflectors. [**ER**] biondo1/. Refl-all-overn

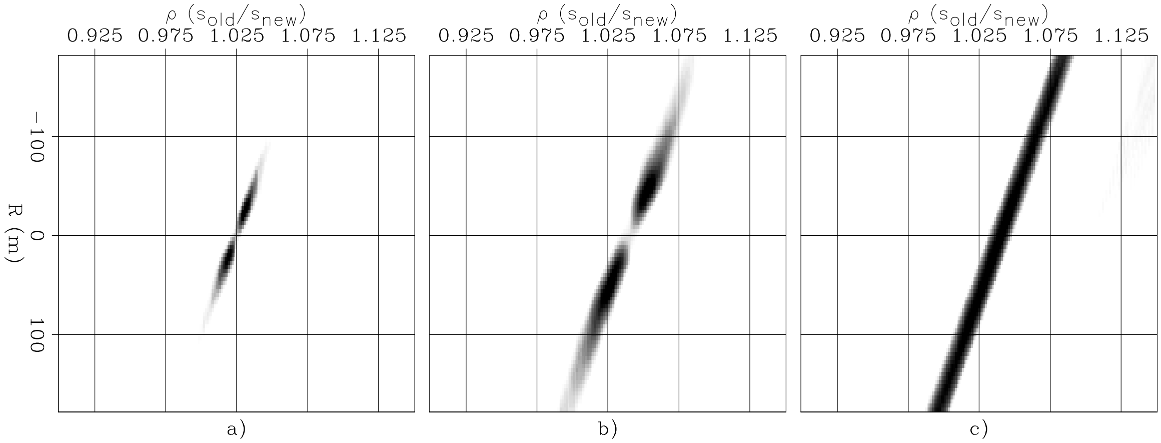


Figure 15: The image-focusing semblance spatially averaged in an inner rectangle of the image space as a function of velocity parameter $\rho$ and the radius of curvature $R$. Panel a) shows the result corresponding to the point diffractors, and panel b) shows the result corresponding to the convex reflectors when the curvature correction was applied by using a field of local dips estimated numerically from the migrated images. Panel c) shows the result corresponding to the convex reflectors when the curvature correction was applied by using a constant local dip equal to zero (i.e. $\bar{\alpha} = 0$). [**CR**] biondo1/. Wind-Stack-all-overn

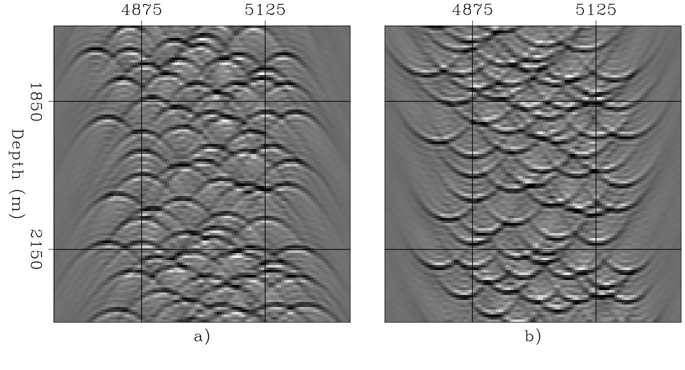


Figure 16: Migrated images of the point-diffractors data corresponding to the values of $\rho$ at the edges of the semblance peak in Figure 15a; that is, $\rho$ = 1.0125 for panel a), and $\rho$ = 1.0375 for panel b). [**CR**] biondo1/. ResMig-all-scatter-overn

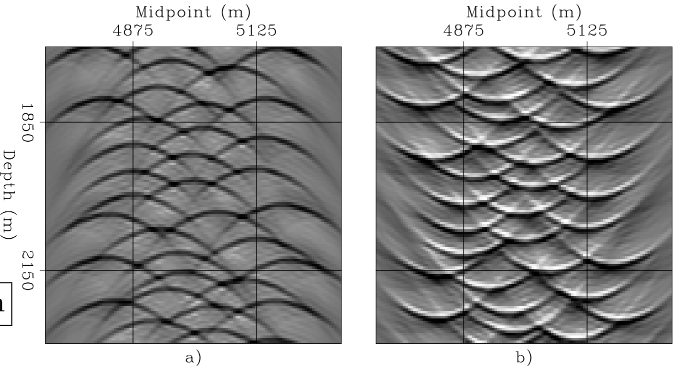


Figure 17: Migrated images of the convex-reflectors data corresponding to the values of $\rho$ at the edges of the semblance peak in Figure 15b; that is, $\rho = 1.01$ for panel a), and $\rho = 1.7$ for panel b). [**CR**] biondo1/. ResMig-all-repl-bump-overn

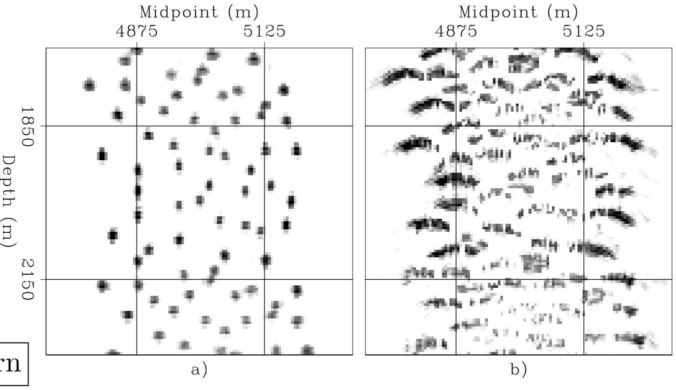


Figure 18: Sections cut through the image-focusing semblance cubes at constant value of $\rho$ and $R$ before spatial averaging. These panels were computed from the point-diffractors data. Panel a) shows semblance for $\rho = 1.025$ and $R = 0$ meters, and panel b) shows semblance for for $\rho = 1.0125$ and $R = 40$ meters. [**CR**] biondo1/. Wind-Sembl-scatter-all-overn

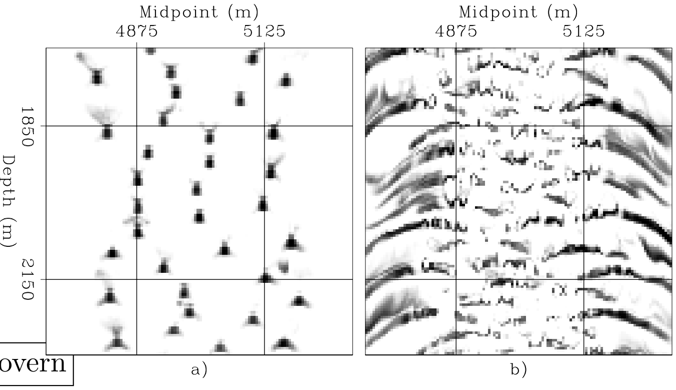


Figure 19: Sections cut through the image-focusing semblance cubes at constant value of $\rho$ and $R$ before spatial averaging. These panels were computed from the convex-reflectors data. Panel a) shows semblance for $\rho = 1.04$ and $R = 0$ meters, and panel b) shows semblance for for $\rho = 1.01$ and $R = 120$ meters. [**CR**] biondo1/. Wind-Sembl-repl-bump-all-overn

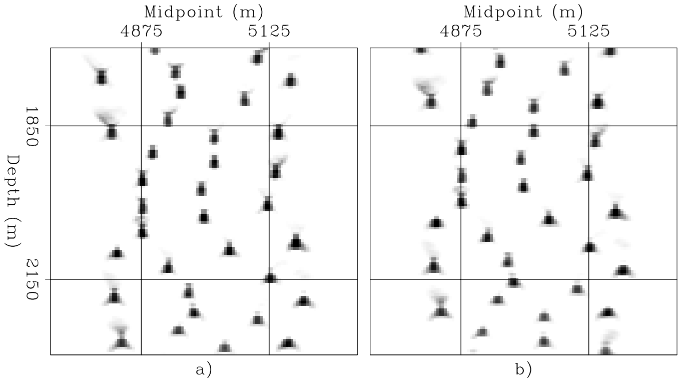


Figure 20: Sections cut through the image-focusing semblance cubes at constant value of $\rho$ and $R$ before spatial averaging. These panels were computed from the convex-reflectors data. The local dip was set to be constant and equal to zero when applying the curvature correction. In contrast, the local dips were numerically estimated when computing the semblance panels shown in Figure 19. Panel a) shows semblance for $\rho = 1.04$ and $R = 0$ meters, and panel b) shows semblance for for $\rho = 1.01$ and $R = 120$ meters. [**CR**] biondo1/. Wind-Sembl-repl-bump-dip-0-all-overn

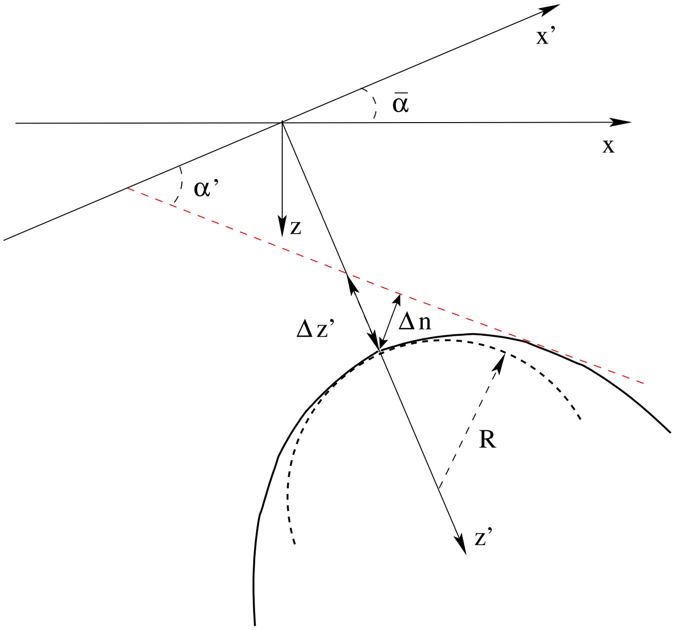


Figure 21: Sketch used to derive the curvature correction presented in 2. The tangent to the parabola (dashed line) needs to be shifted by $\Delta n$ to pass through the vertex of the parabola. [**NR**] biondo1/. Curv-corr

# Attribute combinations for image segmentation

*Adam Halpert and Robert G. Clapp*

## ABSTRACT

Seismic image segmentation relies upon attributes calculated from seismic data, but a single attribute (usually amplitude) is not always sufficient to produce an accurate result. Therefore, a combination of information from different attributes should lead to an improved segmentation outcome. This paper explores opportunities for combining attribute information at three different stages: before segmentation (by multiplying attribute volumes), after the eigenvector calculation (via a linear combination of individual eigenvectors), and after individual boundaries have been drawn (by using uncertainty calculations to extract the best elements of individual boundaries). Overall, a method that uses uncertainty calculations to determine weights for the eigenvector linear combination produces satisfactory results, while avoiding potential drawbacks of other methods. This method produces promising results when tested on field data in both two and three dimensions.

## INTRODUCTION

Image segmentation - an automated process of dividing an image into regions - offers a number of promising applications for seismic data. Among the most straightforward of these applications is to the task of picking salt bodies on seismic images, a process that can be ambiguous and time-consuming when undertaken manually, especially for large three-dimensional datasets with complex salt body geometries. The development of an algorithm for automatically tracking salt boundaries (Lomask, 2007; Lomask et al., 2007) in many cases allows for the quick, efficient and globally-optimized calculation of a salt interface location. Such information may then be used, for example, to quickly update a velocity model as part of an iterative migration system (Halpert et al., 2008).

The seismic image segmentation scheme is based on the Normalized Cut Image Segmentation (NCIS) algorithm (Shi and Malik, 2000), which calculates an eigenvector based on specific attributes gleaned from the image; the eigenvector is then used to trace a boundary across the image. Although the most straightforward attribute for delineating salt boundaries on seismic images is amplitude of the envelope, this attribute alone is not always sufficient to produce an accurate calculation of the boundary. In such cases, other attributes may be used for segmentation. For example, an estimate of dips in a seismic image is often used for interpretation purposes (Bednar, 1997), and strong variations in dominant dips within an image can be indicative of a salt interface. Halpert and Clapp (2008) provide details on using dip variability, as well as an instantaneous frequency attribute, for segmentation with a single attribute. Ideally, however, a segmentation algorithm will combine information from multiple attributes into a single result. In this paper, we discuss three strategies for combining attributes: a multiplication of attribute volumes, a combination of individually calculated boundaries, and a linear combination of individual eigenvectors. The latter method, when combined with an uncertainty measurement derived from the eigenvectors, produces results superior to those using only a single attribute. Since improvements in computing capabilities make increasingly complex segmentation problems tractable, it is important to extend this process to three dimensions. Initial results from a combined-attribute 3D segmentation scheme suggest that a more sophisticated, interpreter-guided segmentation process can be successful.

## ATTRIBUTE COMBINATIONS

In the segmentation algorithm, the determination of a salt interface takes place in three distinct stages. The first stage is the calculation of attributes that may be useful in indicating a boundary between sediments and a salt body. The second stage involves transforming the attribute volumes into eigenvectors of the image via the construction of a weight matrix based on the attribute values. Finally, the third stage "draws" the salt boundary using the eigenvector values. Each of these three stages represents an opportunity for combining information from different attributes. The following sections will explore these three options, and illustrate their advantages and disadvantages with example calculations on a 2D seismic section taken from a 3D Gulf of Mexico field dataset, seen in Figure 1.

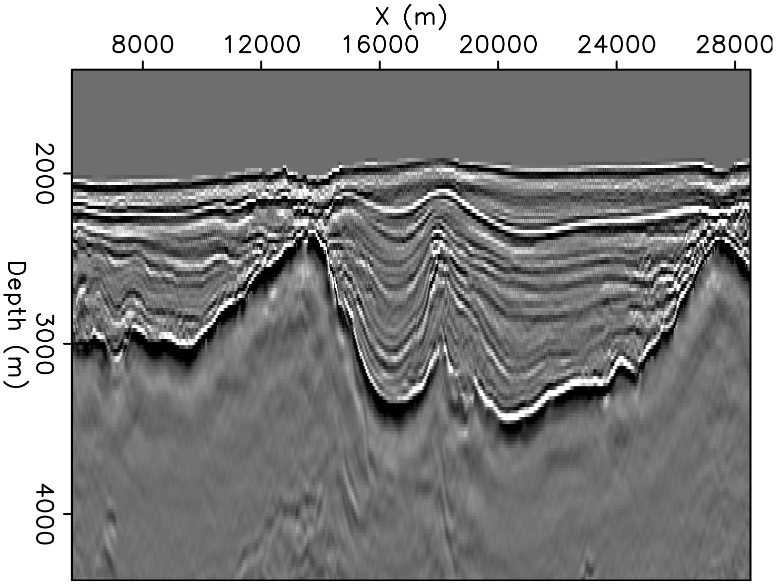


Figure 1: A migrated seismic section used for 2D segmentation examples. Note the discontinous nature of the strong reflector (salt boundary), which will present challenges for the segmentation algorithm. [**ER**] adam1/. dat

The following examples will seek to combine useful information from two attributes - amplitude and dip variability. Figure 2 shows eigenvectors derived from these two individual attributes. The eigenvector values range from -1 to +1; in the figures here, negative values are dark and positive values are light. The salt boundary is typically drawn along the zero-contour of the eigenvector, where values pass from negative to positive. Thus, a sharp transition from dark to light colors in the eigenvector indicates a boundary location with relative certainty, while a grey area indicates a slower transition from negative to positive values, and relative uncertainty of the boundary location. Clearly, the amplitude eigenvector provides better information throughout most of the image, although the transition near $x = 18000$ suggests significant uncertainty. This is logical, as the original section (Figure 1) shows a great deal on discontinuity at this location. Overall, the dip eigenvector shows much less certainty than the one derived from the amplitude attribute; however, the previously mentioned location appears more certain on the dip eigenvector. The boundary calculations corresponding to these two eigenvectors (Figure 3) confirm these observations. Therefore, an obvious goal for combining information from these two attributes is to produce a boundary that uses information from the amplitude attribute in most locations, but incorporates the dip information at this location.

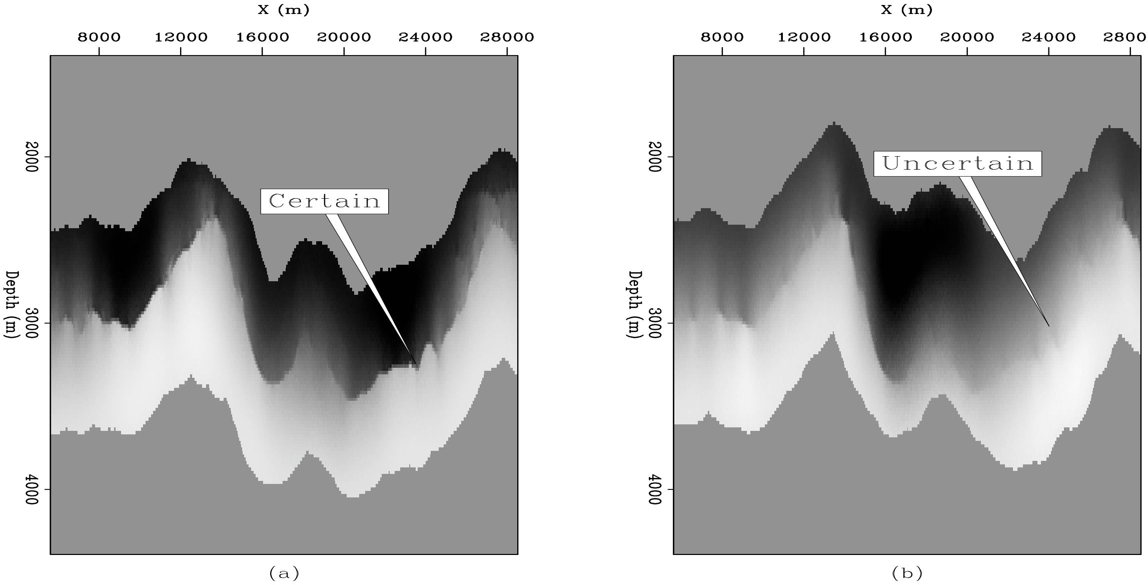


Figure 2: Eigenvectors derived from amplitude of the envelope (a) and dip variability (b) attributes. Areas of relative boundary certainty and uncertainty are indicated. [**CR**] adam1/. eigs-ann

## Attribute multiplication

One approach, suggested by Lomask (2007), is to combine multiple attribute volumes into a single volume via multiplication:

$$A = \prod_{i=\text{each attribute}}^{\text{all attributes}} a_i \ , \qquad (1)$$

where $a_i$ is an individual attribute volume, and then proceed with segmentation normally. Multiplication of the attribute data has the effect of reinforcing information in areas where the attributes “agree,” which can be beneficial. However, it also can have the effect of destroying potentially valuable information if the two attributes are not in agreement. Panel (a) in Figure 4 shows the boundary calculation resulting from this process.

Clearly, in this case the disadvantages of multiplying attribute volumes together outweigh the possible advantages - the process appears to have incorporated the worst infor-

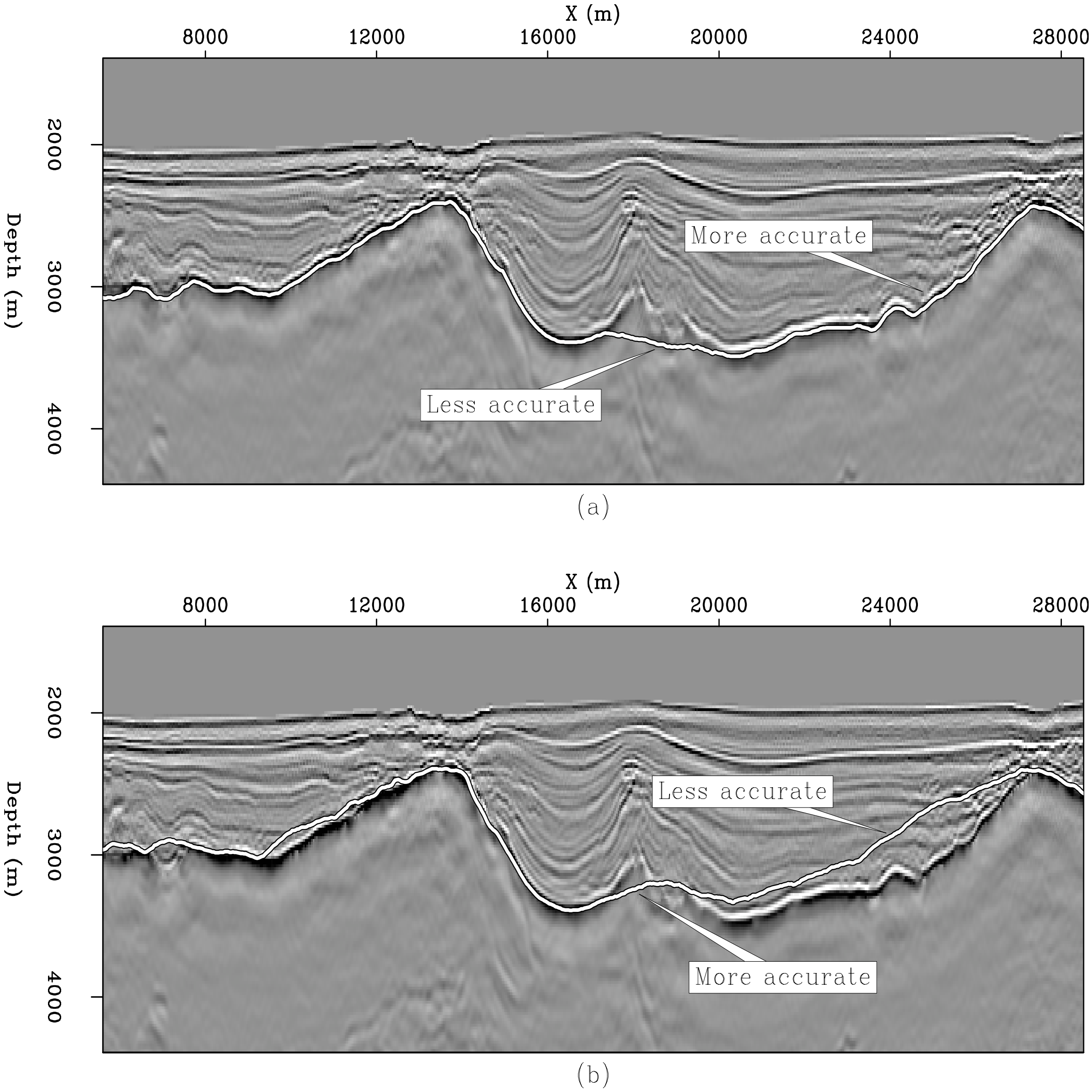


Figure 3: Zero-contour boundaries corresponding to the amplitude (a) and dip variability (b) eigenvectors seen in Figure 2. [**CR**] adam1/. ampdipbnd

mation from each of the attributes, resulting in a final boundary that does not improve on either of the individual results (Figure 3) in any location.

## Boundary combinations

A second "domain" in which information from different attributes may be combined is after individual boundary calculations have already taken place. This method requires a measure of uncertainty along each individual boundary, so that a new boundary can be created by incorporating the "most certain" boundary at each location in the image. As discussed previously, such an uncertainty measure may be gleaned from the zero-crossing of the eigenvector:

$$d = |p_1 - n_1| \ , \tag{2}$$

where $p_1$ and $n_1$ are the two values on either side of the boundary (one will be positive, one negative). A sharp transition from positive to negative values - quantitatively, a large value of $d$ at that location - signifies relative certainty, while a slow transition or small difference signals uncertainty. In this case, the measurement is taken perpendicular to the calculated boundary, so as to avoid the assumption that the boundary is in all locations locally horizontal. After such calculations are made at all locations for each boundary, a combined boundary is formed by taking the most certain boundary location (depth value) at each horizontal location. Panel (b) in Figure 4 shows the result of this process.

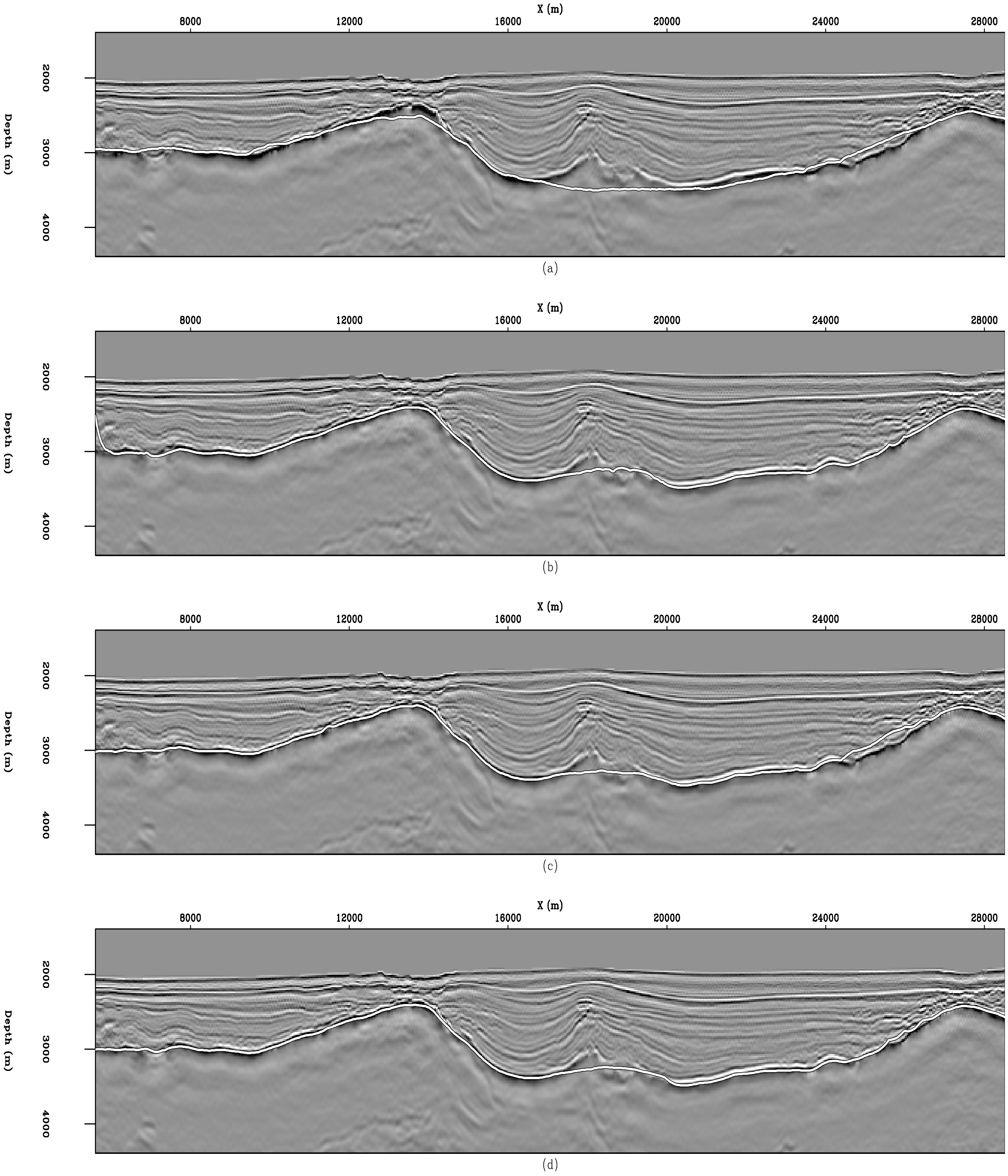
This approach performs very well in this example. Information from the amplitude attribute is honored nearly everywhere, and the dip information is incorporated only where it is superior to the amplitude information. However, the manner in which this approach is implemented could lead to problems in some circumstances. Taking the best elements of different boundaries could easily lead to erratic, "either/or" behavior in the combined boundary; indeed, some indications of this behavior may be seen in the jaggedness of the boundary where the dip information plays a significant role. It is likely that this behavior would be even more troublesome in three dimensions.

## Eigenvector combinations

Finally, a third approach is to use the individual attribute volumes to calculate multiple eigenvectors, and then combine the eigenvectors before determining a boundary. Following the recommendation of Shi and Malik (2000), a simple way to combine the eigenvectors is via linear combination:

$$E = \sum_{i=\text{each attribute}}^{\text{all attributes}} \lambda_i e_i \ , \tag{3}$$

where $e_i$ is an individual eigenvector and $\lambda_i$ is a specific weight value assigned to the attribute in question. Of course, taking this approach introduces the problem of determining weight values for each attribute. Panel (c) in Figure 4 shows the result of this approach if equal weights are given to the amplitude and dip attributes. While the boundary is satisfactory in many locations, the dip attribute clearly has too much influence in some areas where the amplitude attribute provides much better information. This method shows promise, but a



(a)

(b)

(c)

(d)

Figure 4: Calculated boundaries corresponding to: (a) Attribute multiplication segmentation; (b) Combination of individual boundaries; (c) Equally-weighted eigenvector combination; and (d) Uncertainty-weighted eigenvector combination. [**CR**] adam1/. uno-bnds

mechanism for assigning better weights is necessary. One such mechanism has already been discussed; we can use the eigenvector uncertainty measurement, utilized previously for the boundary combination approach, to assign attribute weight values for a linear combination of eigenvectors. In this way, we are able to follow the recommendation of Shi and Malik for combining information from different sources, while at the same time taking advantage of a "built-in" method for estimating uncertainties.

Since the eigenvectors range in value from -1 to +1, the eigenvector difference across one of the boundaries can never be greater than two. Thus, the value

$$\omega = \frac{1}{2}(1 + d_{amp} - d_{dip}) \ , \tag{4}$$

where $d$ is the difference across a calculated boundary at a particular $x$ location, will range from 0 to 1. If we want to heavily penalize uncertainty in one of the eigenvectors, we set the weight values as follows:

$$W_{amp} = \begin{cases} \omega^2 & \text{if } \omega < 0.5 \\ \sqrt{\omega} & \text{if } \omega > 0.5 \end{cases} \tag{5}$$

$$W_{dip} = 1 - w_{amp} \ . \tag{6}$$

The results of assigning weight values in this manner to create an eigenvector are shown in panel (d) of Figure 4. The boundary successfully follows the salt interface everywhere the amplitude-only boundary does, and incorporates the dip information only where the amplitude boundary fails. Furthermore, we do not see the erratic behavior in areas where the dip information is most significant, as we did for the boundary combination method.

## SEGMENTATION IN THREE DIMENSIONS

We have shown that image segmentation with one or multiple attributes can be very effective for 2D seismic data. However, it is in three dimensions that the advantages of automated image segmentation should become even more apparent. While a skilled human interpreter can easily examine a 2D section and pick out a salt interface, visualization and time constraints make this a very difficult process for a 3D survey. In contrast, a computer is not bound by these limitations and can excel at "seeing" in three dimensions. Furthermore, the drastic increase in the number of pixel-to-pixel comparisons available in three dimensions compared to two should also increase the robustness and accuracy of the segmentation process.

### Computational issues

Of course, moving to 3D also greatly increases the computational complexity and expense of the image segmentation process. Lomask (2007) describes several modifications to the algorithm that help to lessen the impact, such as comparing each pixel to a random selection of other pixels instead of all pixels in a specified neighborhood. However, constant technological advances in the computer hardware industry also contribute to the increasing tractability of large-scale computational problems such as this one. The segmentation algorithm used here involves heavy computations with very large, sparse matrices. As such, a promising avenue of interest is to work with many-core, large-memory machines such as those recently developed by SiCortex (Reilly et al., 2006). Because such machines feature very fast interprocessor communication capabilities, they lend themselves well to the sparse-matrix eigenvector calculation portion of the segmentation scheme that, in most cases, represents the majority of overall computational expense. Early implementations of the eigenvector calculation algorithm on a SiCortex development machine with 72 low-power, relatively low-performance nodes bear out this hypothesis. The matrix-vector multiplications needed for calculation of the eigenvector on a 250 x 400 x 50 cube of data required approximately four minutes on this machine, representing a speedup of over 750% when compared to the same calculations on a single processor with much higher relative speed and power consumption. Optimization of codes to take greater advantage of the machine's capabilities should further improve these results.

### Attribute combinations in 3D

An ideal goal for an image segmentation algorithm is to be able to extend information gathered from a 2D seismic section by using it to guide the segmentation for a 3D volume. For instance, if an interpreter picks an interface on the 2D section, an automated inversion scheme could determine which combination of attribute information would have led the segmentation algorithm to produce the same boundary. This information would then be used to segment the entire 3D cube. Here, we have an opportunity to test a primitive version of this process.

Figure 5 displays a depth slice, inline section and crossline section of the 3D cube containing a portion of the seismic section (Figure 1) used to demonstrate the 2D boundary combinations above; the inline section shown is not the same as the one used for Figure 1. Single-attribute 3D segmentations with amplitude and dip variability attributes produce the eigenvectors seen in Figure 6. As we saw in the 2D case, the amplitude segmentation shows greater certainty in most locations; however, the dip variability eigenvector is noticeably superior in the two indicated areas. We seek to combine the two eigenvector volumes such that the most accurate information from each attribute is contained in a single eigenvector volume.

Previously, we used an uncertainty-weighted eigenvector combination scheme to produce the boundary in panel (d) of Figure 4. From this process, we can retain the individual weight values used for each attribute's eigenvector at each $x$-direction sample. By making the assumption that these weights will remain constant in the crossline direction, we can combine the 3D eigenvector volumes by using these same weight values for every crossline section. Figure 7 shows the results of this process for the same slices displayed in Figure 5. The new eigenvector improves on the ambiguities indicated on the amplitude eigenvector in Figure 6, yet retains the amplitude eigenvector's superior results in other locations. The corresponding zero-contour boundaries for these slices are seen in Figure 8; the boundary accurately tracks the salt interface on all three sections.

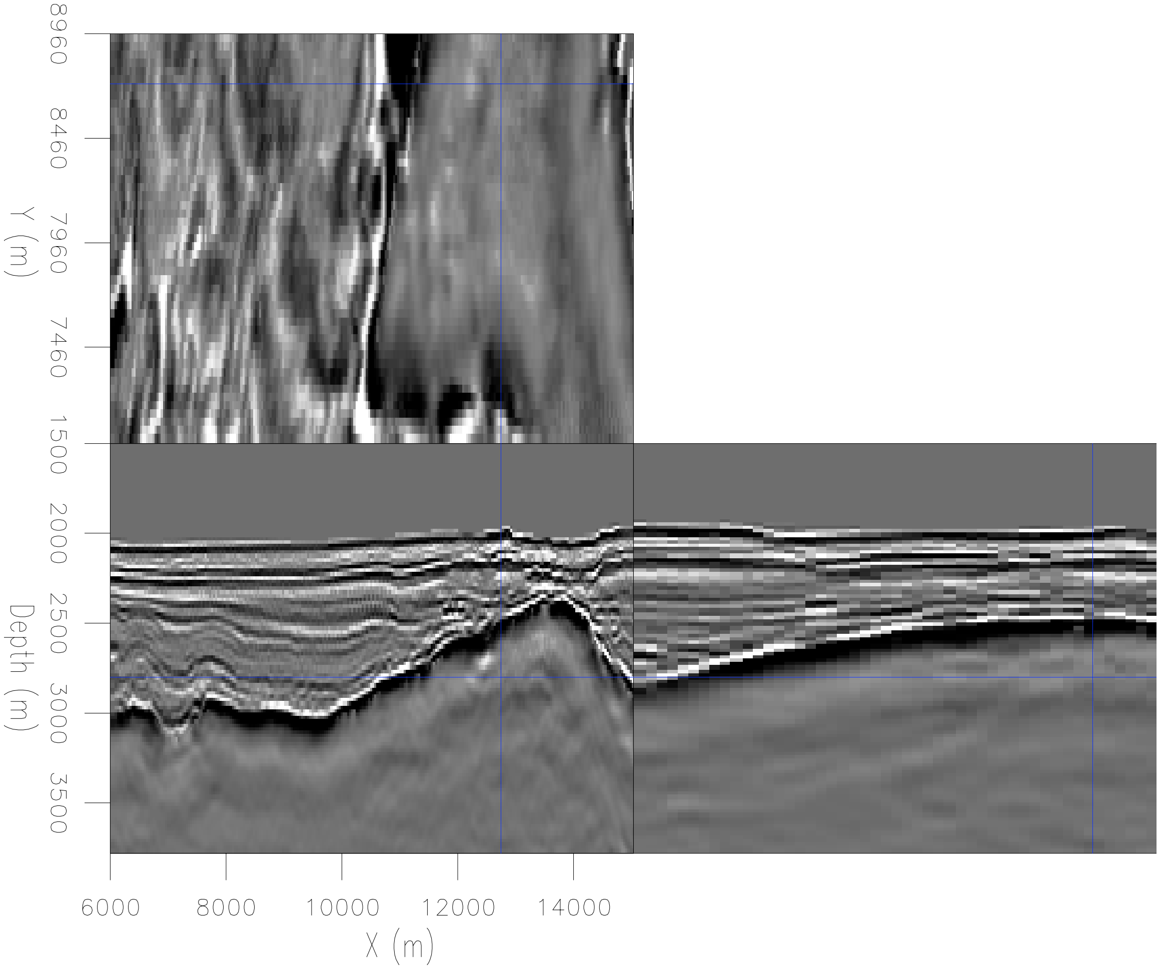


Figure 5: Depth slice and inline and crossline sections of a seismic data cube used for 3D image segmentation. [**ER**] adam1/. unocal4

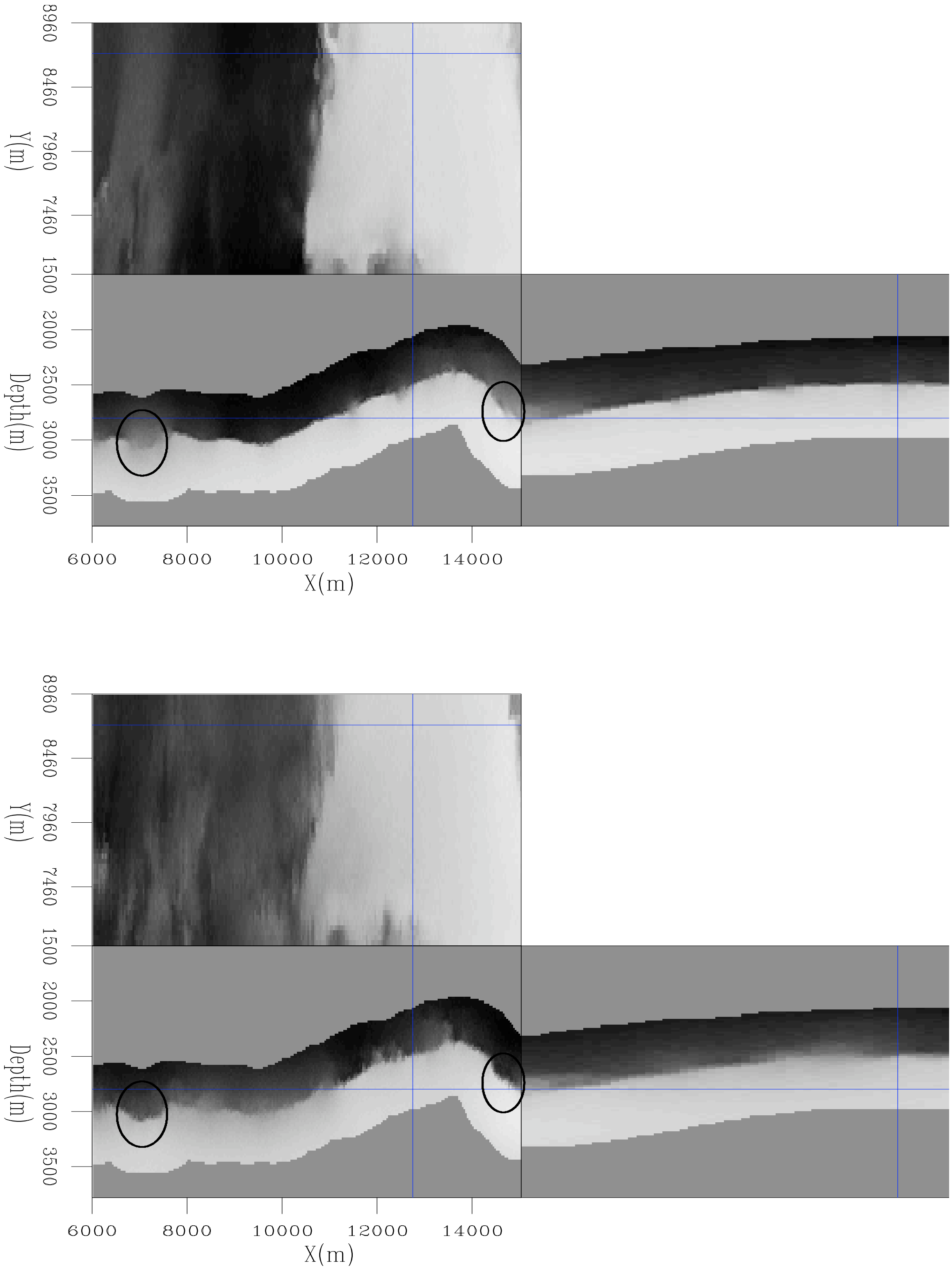


Figure 6: Slices of the 3D eigenvectors calculated from amplitude (top) and dip variability (bottom) attributes corresponding to the image in Figure 5. The circles indicate areas where the dip eigenvector is noticeably superior to the amplitude eigenvector. [**CR**] adam1/. 3deigs

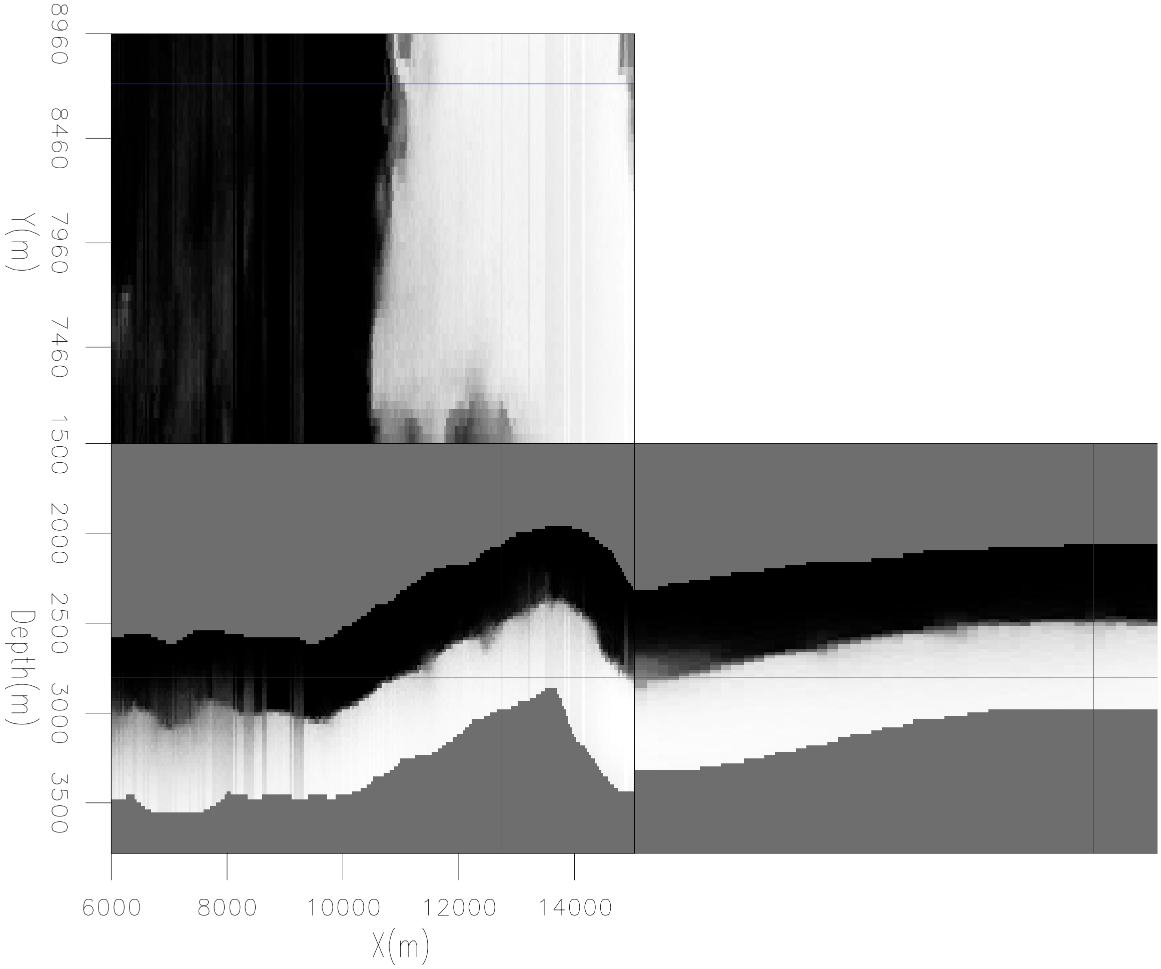


Figure 7: Combined eigenvector, using a linear combination of the eigenvectors in Figure 6 with weights determined during the 2D example. [**CR**] adam1/. comboeig

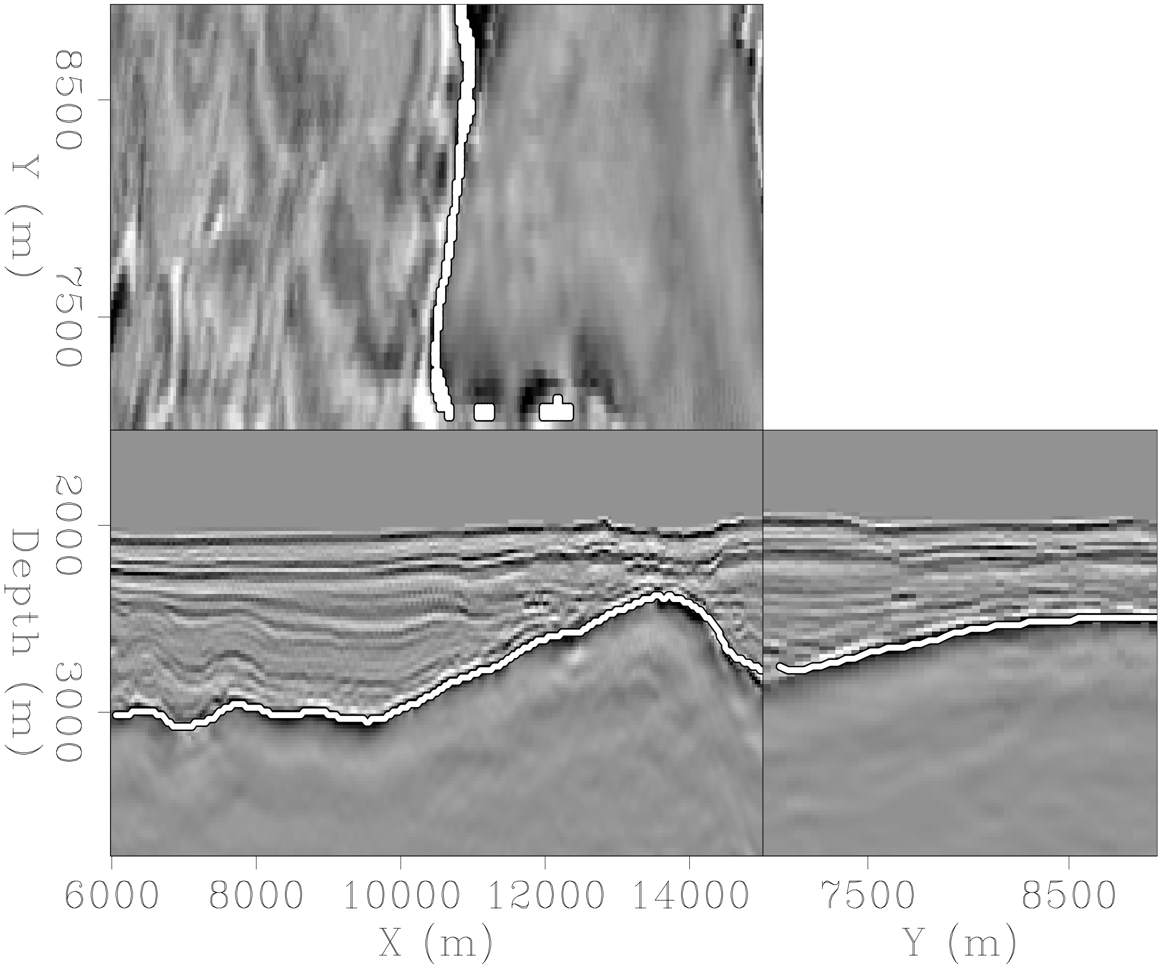


Figure 8: Zero-contour boundary corresponding to the combined eigenvector in Figure 7. The salt interface is accurately tracked on all three sections. [**CR**] adam1/. 3dbnd

## CONCLUSIONS

Seismic image segmentation using a single attribute is not always sufficient to produce an accurate salt boundary calculation, so the use of other attributes such as dip variability is often necessary. By combining information from different attributes, we hope to incorporate the most reliable information from each attribute into a single, improved segmentation result. While opportunities for such combinations exist at several stages of the segmentation process, the most promising method in 2D involves a linear combination of eigenvectors from individual attributes, weighted according to uncertainties derived from each eigenvector. In the examples here, this approach successfully incorporates useful information from two different attributes, while avoiding potential pitfalls of other methods. This method may be extended to three dimensions with the assumption that weight values are constant in the crossline direction; in the 3D example shown here, such an approach yields an improved eigenvector and accurate salt interface pick on a 3D seismic cube. While the constant-weights assumption is part of an early and somewhat primitive approach, the results here hold promise for the success of more sophisticated 3D image segmentations schemes.

## ACKNOWLEDGMENTS

We thank Phil Schultz (formerly at Unocal, now Chevron) for providing the field data used in these examples.

## REFERENCES

Bednar, J. B., 1997, Least squares dip and coherency attributes: SEP-Report, **95**, 219–225.

Halpert, A. and R. G. Clapp, 2008, Salt body segmentation with dip and frequency attributes: SEP-Report, **136**.

Halpert, A., R. G. Clapp, J. Lomask, and B. L. Biondi, 2008, Image segmentation for velocity model construction and updating: SEG Technical Program Expanded Abstracts, **27**.

Lomask, J., 2007, Seismic volumetric flattening and segmentation: PhD thesis, Stanford University.

Lomask, J., R. G. Clapp, and B. Biondi, 2007, Application of image segmentation to tracking 3d salt boundaries: Geophysics, **72**, P47–P56.

Reilly, M., L. C. Stewart, J. Leonard, and D. Gingold, 2006, Sicortex technical summary: SiCortex.

Shi, J. and J. Malik, 2000, Normalized cuts and image segmentation: Institute of Electrical and Electronics Engineers Transactions on Pattern Analysis and Machine Intelligence, **22**, 838–905.

# Wave-equation tomography using image-space phase-encoded data

*Claudio Guerra, Yaxun Tang, and Biondo Biondi*

## ABSTRACT

Wave-equation tomography in the image-space is a powerful technique that promises to yield more reliable velocity models than ray-based migration velocity analysis in areas of complex overburden. Its practical use, however, has been limited because of the high computational cost. Applying a target-oriented approach and using data reduction can make wave-equation tomography in the image space of practical use. Here, we present results of applying image-space wave-equation tomography in the generalized source domain, where a small number of synthesized shot gathers are generated. Specifically, we generate synthesized shot gathers by image-space phase encoding. This technique can also be used in a target-oriented way. The comparison of the gradients of the tomography objective functional obtained using image-space encoded gathers with those obtained using the original shot gathers shows that those encoded shot gathers can be used in wave-equation tomography problems. Velocity inversion using image-space phase-encoded gathers converges to reasonable results when compared to the correct velocity model. We illustrate our method by applying it to the Marmousi model.

## INTRODUCTION

Wave-equation tomography has the potential to overcome the problems faced by ray-based traveltime tomography when estimating the velocity model in complex geological scenarios. This is because wave-equation tomography uses band-limited wavefields instead of infinite-frequency rays as carriers of information; thus it is robust even in the presence of strong velocity contrasts and immune to multi-pathing issues. However, despite its theoretical advantages, wave-equation tomography is still computationally challenging.

Wave-equation tomography can be performed in the data-space domain (Tarantola, 1987; Woodward, 1992) or in the image-space domain (Biondi and Sava, 1999; Shen, 2004). The image-space approach minimizes the residual in the image domain obtained after migration. Regardless of the domain of application, using phase-encoded data can substantially decrease the computational cost of wave-equation tomography(Vigh and Starr, 2008; Shen and Symes, 2008). Tang et al. (2008) extended the theory of image-space wave-equation tomography from the conventional shot-profile domain (Shen, 2004) to the generalized source domain. The generalized source domain can be obtained in two different spaces. In the data-space, shot gathers are combined and the corresponding source function is synthesized, using a convenient phase-encoding scheme, which characterizes the data-space phase encoding (Whitmore, 1995; Romero et al., 2000). In the image-space, source- and receiver-areal data are synthesized by upward propagating wavefields. The initial condition for the modeling is a prestack image computed with wave-equation migration, according to the prestack exploding-reflector modeling (Biondi, 2006, 2007). The modeling experiments can be combined such that a small quantity of areal data is generated. In this case, to mitigate crosstalk during imaging, the modeling experiments and reflectors are phase-encoded, characterizing the image-space phase encoding (Guerra and Biondi, 2008). To encode the reflectors, a picking step of some key reflectors is necessary.

In this paper, we show that image-space phase-encoded wavefields can be used to estimate the velocity model in image-space wave-equation tomography. We show that the gradient of the tomographic objective functional is similar to that obtained in the original shot-profile domain, but with less computational cost. Velocity inversion using image-space phase-encoded gathers converges to reasonable results when compared to the correct velocity model, provided that crosstalk has been sufficiently attenuated. We briefly discuss the theory of wave-equation tomography in the image-space domain; then we explain the prestack exploding-reflector modeling and show that the image-space phase encoding can be used to accelerate wave-equation tomography in the image domain. We use the Marmousi model to illustrate the method.

## IMAGE-SPACE WAVE-EQUATION TOMOGRAPHY

Image-space wave-equation tomography is a non-linear inverse problem that tries to find an optimal background slowness that minimizes the residual field, $\Delta\mathbf{I}$, defined in the image space. The residual field is derived from the background image, $\mathbf{I}$, which is computed with a background slowness. The general form of the residual field is (Biondi, 2008)

$$\Delta\mathbf{I} = \mathbf{I} - \mathbf{F}(\mathbf{I}), \tag{1}$$

where $\mathbf{F}$ is a focusing operator, which measures the focusing of the migrated image. In particular, in the Differential Semblance Optimization (DSO) method (Shen, 2004), the focusing operator takes the form:

$$\mathbf{F}(\mathbf{I}) = (\mathbf{1} - \mathbf{O})\,\mathbf{I}, \tag{2}$$

where $\mathbf{1}$ is the identity operator and $\mathbf{O}$ is the DSO operator either in the subsurface offset domain or in the angle domain (Shen, 2004).

Under $\ell_2$ norm, the tomography objective function can be written as follows:

$$J = \frac{1}{2}||\Delta\mathbf{I}||_2 = \frac{1}{2}||\mathbf{I} - \mathbf{F}(\mathbf{I})||^2. \tag{3}$$

The gradient of $J$ with respect to the slowness $\mathbf{s}$ is

$$\nabla J = \left(\frac{\partial\mathbf{I}}{\partial\mathbf{s}} - \frac{\partial\mathbf{F}(\mathbf{I})}{\partial\mathbf{s}}\right)^{*}(\mathbf{I} - \mathbf{F}(\mathbf{I}))\,, \tag{4}$$

where $^*$ denotes the adjoint.

The linear operator $\frac{\partial\mathbf{I}}{\partial\mathbf{s}}\big|_{\mathbf{s}=\widehat{\mathbf{s}}}$, which defines a linear mapping from the slowness perturbation $\Delta\mathbf{s}$ to the image perturbation $\Delta\mathbf{I}$, can be computed by expanding the image $\mathbf{I}$ around the background slowness $\widehat{\mathbf{s}}$. Keeping only the zeroth and first order terms, we get the linear operator $\frac{\partial\mathbf{I}}{\partial\mathbf{s}}\big|_{\mathbf{s}=\widehat{\mathbf{s}}}$ as follows:

$$\Delta\mathbf{I} = \left.\frac{\partial\mathbf{I}}{\partial\mathbf{s}}\right|_{\mathbf{s}=\widehat{\mathbf{s}}}\Delta\mathbf{s} = \mathbf{T}\Delta\mathbf{s}, \tag{5}$$

where $\Delta\mathbf{I} = \mathbf{I} - \widehat{\mathbf{I}}$, $\widehat{\mathbf{I}}$ is the background image computed with the background slowness $\widehat{\mathbf{s}}$ and $\Delta\mathbf{s} = \mathbf{s} - \widehat{\mathbf{s}}$. $\mathbf{T} = \frac{\partial \mathbf{I}}{\partial \mathbf{s}}\big|_{\mathbf{s}=\widehat{\mathbf{s}}}$ is the wave-equation tomographic operator. The tomographic operator can be evaluated either in the source and receiver domain (Sava, 2004) or in the shot-profile domain (Shen, 2004).

In the shot-profile domain, both source and receiver wavefields are downward continued with the one-way wave equations (Claerbout, 1971)

$$\left\{ \begin{array}{l} \left(\frac{\partial}{\partial z} + i\Lambda\right) D(\mathbf{x}, \mathbf{x}_s, \omega) = 0 \\ D(x, y, z = 0, \mathbf{x}_s, \omega) = f_s(\omega)\delta(\mathbf{x} - \mathbf{x}_s) \end{array} \right. , \tag{6}$$

and

$$\left\{ \begin{array}{l} \left(\frac{\partial}{\partial z} + i\Lambda\right) U(\mathbf{x}, \mathbf{x}_s, \omega) = 0 \\ U(x, y, z = 0, \mathbf{x}_s, \omega) = Q(x, y, z = 0, \mathbf{x}_s, \omega) \end{array} \right. , \tag{7}$$

where $D(\mathbf{x}, \mathbf{x}_s, \omega)$ is the source wavefield for a single frequency $\omega$ at image point $\mathbf{x} = (x, y, z)$ with the source located at $\mathbf{x}_s = (x_s, y_s, 0)$; $U(\mathbf{x}, \mathbf{x}_s, \omega)$ is the receiver wavefield for a single frequency $\omega$ at image point $\mathbf{x}$ for the source located at $\mathbf{x}_s$; $f_s(\omega)$ is the frequency dependent source signature, and $f_s(\omega)\delta(\mathbf{x} - \mathbf{x}_s)$ defines the point source function at $\mathbf{x}_s$, which serves as the boundary condition of Equation 6. $Q(x, y, z = 0, \mathbf{x}_s, \omega)$ is the recorded shot gather for the shot located at $\mathbf{x}_s$, which serves as the boundary condition of Equation 7. $\Lambda$ is the square-root operator

$$\Lambda = \sqrt{\omega^2 s^2(\mathbf{x}) - |\mathbf{k}|^2}, \tag{8}$$

where $s(\mathbf{x})$ is the slowness at $\mathbf{x}$; $\mathbf{k} = (k_x, k_y)$ is the spatial wavenumber vector. The image is computed by applying the cross-correlation imaging condition:

$$I(\mathbf{x}, \mathbf{h}) = \sum_{\mathbf{x}_s} \sum_{\omega} \overline{D(\mathbf{x} - \mathbf{h}, \mathbf{x}_s, \omega)} U(\mathbf{x} + \mathbf{h}, \mathbf{x}_s, \omega), \tag{9}$$

where the overline stands for complex conjugate; $D(\mathbf{x}, \mathbf{x}_s, \omega)$ is the source wavefield for a single frequency $\omega$ at image point $\mathbf{x} = (x, y, z)$ with the source located at $\mathbf{x}_s = (x_s, y_s, 0)$; $U(\mathbf{x}, \mathbf{x}_s, \omega)$ is the receiver wavefield and $\mathbf{h} = (h_x, h_y, h_z)$ is the subsurface half-offset.

The perturbed image can be derived by the application of the chain rule to Equation 9:

$$\begin{aligned} \Delta I(\mathbf{x}, \mathbf{h}) \quad = \quad & \sum_{\mathbf{x}_s} \sum_{\omega} \Big( \overline{\Delta D(\mathbf{x} - \mathbf{h}, \mathbf{x}_s, \omega)} \widehat{U}(\mathbf{x} + \mathbf{h}, \mathbf{x}_s, \omega) + \\ & \overline{\widehat{D}(\mathbf{x} - \mathbf{h}, \mathbf{x}_s, \omega)} \Delta U(\mathbf{x} + \mathbf{h}, \mathbf{x}_s, \omega) \Big), \end{aligned} \tag{10}$$

where $\widehat{D}(\mathbf{x} - \mathbf{h}, \mathbf{x}_s, \omega)$ and $\widehat{U}(\mathbf{x} + \mathbf{h}, \mathbf{x}_s, \omega)$ are the background source and receiver wavefields computed with the background slowness $\widehat{s}(\mathbf{x})$; $\Delta D(\mathbf{x} - \mathbf{h}, \mathbf{x}_s, \omega)$ and $\Delta U(\mathbf{x} + \mathbf{h}, \mathbf{x}_s, \omega)$ are the perturbed source wavefield and perturbed receiver wavefield, which are the results of the slowness perturbation $\Delta s(\mathbf{x})$.

To evaluate the adjoint of the tomographic operator, $\mathbf{T}^*$, we first apply the adjoint of the imaging condition to get the perturbed source and receiver wavefields, $\Delta D(\mathbf{x} + \mathbf{h}, \mathbf{x}_s, \omega)$ and $\Delta U(\mathbf{x} + \mathbf{h}, \mathbf{x}_s, \omega)$, as follows

$$\begin{aligned}\Delta D(\mathbf{x},\mathbf{x}_s,\omega) &= \sum_{\mathbf{h}} \Delta I(\mathbf{x},\mathbf{h})\widehat{U}(\mathbf{x}+\mathbf{h},\mathbf{x}_s,\omega)\\ \Delta U(\mathbf{x},\mathbf{x}_s,\omega) &= \sum_{\mathbf{h}} \Delta I(\mathbf{x},\mathbf{h})\widehat{D}(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega).\end{aligned} \tag{11}$$

The perturbed source and receiver wavefields satisfy the following one-way wave equations, linearized with respect to slowness:

$$\begin{cases}\left(\frac{\partial}{\partial z}+i\Lambda\right)\Delta D(\mathbf{x},\mathbf{x}_s,\omega) = \left(\frac{-i\omega}{\sqrt{1-\frac{|\mathbf{k}|^2}{\omega^2\widehat{s}^2(\mathbf{x})}}}\widehat{D}(\mathbf{x},\mathbf{x}_s,\omega)\right)\Delta s(\mathbf{x}) \\ \Delta D(x,y,z=0,\mathbf{x}_s,\omega)=0\end{cases}, \tag{12}$$

and

$$\begin{cases}\left(\frac{\partial}{\partial z}+i\Lambda\right)\Delta U(\mathbf{x},\mathbf{x}_s,\omega) = \left(\frac{-i\omega}{\sqrt{1-\frac{|\mathbf{k}|^2}{\omega^2\widehat{s}^2(\mathbf{x})}}}\widehat{U}(\mathbf{x},\mathbf{x}_s,\omega)\right)\Delta s(\mathbf{x}) \\ \Delta U(x,y,z=0,\mathbf{x}_s,\omega)=0\end{cases}. \tag{13}$$

When solving the optimization problem, the gradient of the objective function is obtained by computing the perturbed wavefields using the adjoint of the imaging operator (equation 11), where the image perturbation results from the application of a focusing operator (equation 1) on the background image; then the scattered wavefields are obtained by applying the adjoint of the one-way wave equations 12 and 13; and, finally, the adjoint scattering operator cross-correlates the upward propagated the scattered wavefields with the modified background wavefields (term in the parenthesis on the right-hand side of equations 12 and 13). Figure 1 displays the image-space wave-equation tomography flowchart. The gray box on the left represents the process of obtaining the image perturbation, while the gray box on the right corresponds to the application of the adjoint of the wave-equation tomography operator. WE stands for wavefield extrapolation. The light gray boxes contain the wavefields, images and slowness perturbation. The processes and operators are represented as white boxes. More detailed information on how to evaluate the forward and adjoint operators can be found in Tang et al. (2008).

## PRESTACK EXPLODING-REFLECTOR MODELING

The general idea of prestack exploding-reflector modeling (Biondi, 2006) is to model the data and corresponding source function that are related to only one event in the subsurface. In this case, a single unfocused subsurface-offset-domain common-image gather (SODCIG) containing a single reflector is used as the initial condition for recursive upward continuation with the following one-way wave equations:

$$\begin{cases}\left(\frac{\partial}{\partial z}-i\Lambda\right)Q_D(\mathbf{x},\omega;x_m,y_m)=I_D(\mathbf{x},\mathbf{h};x_m,y_m)\\ Q_D(x,y,z=z_{\max},\omega;x_m,y_m)=0\end{cases}, \tag{14}$$

and

$$\begin{cases}\left(\frac{\partial}{\partial z}-i\Lambda\right)Q_U(\mathbf{x},\omega;x_m,y_m)=I_U(\mathbf{x},\mathbf{h};x_m,y_m)\\ Q_U(x,y,z=z_{\max},\omega;x_m,y_m)=0\end{cases}, \tag{15}$$

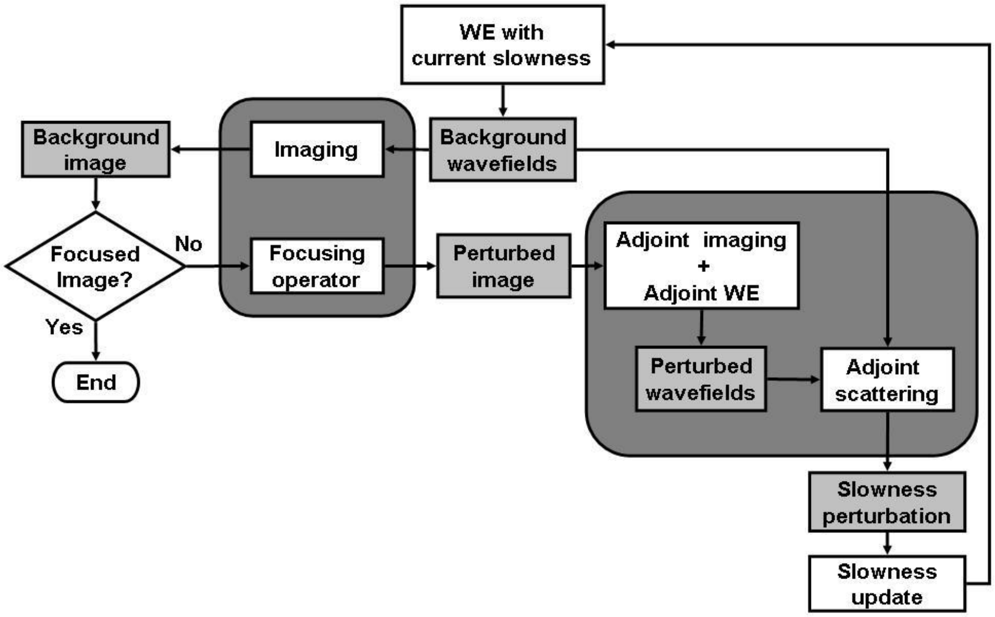


Figure 1: Image-space wave-equation tomography flowchart. The gray box on the left represents the process of obtaining the image perturbation, while the gray box on the right corresponds to the application of the adjoint of the wave-equation tomography operator.[**NR**] claudio1/. ISWET

where $I_D(\mathbf{x}, \mathbf{h}; x_m, y_m)$ and $I_U(\mathbf{x}, \mathbf{h}; x_m, y_m)$ are the isolated SODCIGs at the horizontal location $(x_m, y_m)$ for a single reflector, and are suitable for the initial conditions for the source and receiver wavefields, respectively. As Biondi (2006) discusses, a rotation of the image gathers according to the apparent geological dip must be performed prior to modeling. By collecting the wavefields at the surface, we obtain the areal source data $Q_D(x, y, z = 0, \omega; x_m, y_m)$ and the areal receiver data $Q_U(x, y, z = 0, \omega; x_m, y_m)$ for a single reflector and a single SODCIG located at $(x_m, y_m)$. $\Lambda$ is the square-root operator defined by

$$\Lambda = \sqrt{\omega^2 \hat{s}^2(\mathbf{x}) - |\mathbf{k}|^2},$$

where $s(\mathbf{x})$ is the slowness at $\mathbf{x}$ and $\mathbf{k} = (k_x, k_y)$ is the spatial wavenumber vector.

Since the size of the migrated image volume can be very large in practice, and there are usually many reflectors in the subsurface, modeling each reflector and each SODCIG may generate a data set even larger than the original data set. One strategy to reduce the cost is to model several reflectors and several SODCIGs simultaneously (Biondi, 2006); however, this process generates unwanted crosstalk. As discussed by Guerra and Biondi (2008), random phase encoding can be used to attenuate the crosstalk.

One important characteristic of the prestack exploding reflector modeling is that, for velocity model building, the wavefields can be upward propagated to a certain depth level or depth horizon, provided that the velocity model above is sufficiently accurate. Therefore, a target-oriented strategy can be applied to derive the velocity model below the that depth.

## IMAGE-SPACE PHASE-ENCODED WAVEFIELDS

The randomly encoded areal source and areal receiver wavefields can be computed as follows:

$$\left\{ \begin{array}{l} \left(\frac{\partial}{\partial z} - i\Lambda\right) \widetilde{Q}_D(\mathbf{x}, \mathbf{p}_m, \omega) = \widetilde{I}_D(\mathbf{x}, \mathbf{h}, \mathbf{p}_m, \omega) \\ \widetilde{Q}_D(x, y, z = z_{\max}, \mathbf{p}_m, \omega) = 0 \end{array} \right. , \tag{16}$$

and

$$\left\{ \begin{array}{l} \left(\frac{\partial}{\partial z} - i\Lambda\right) \widetilde{Q}_U(\mathbf{x}, \mathbf{p}_m, \omega) = \widetilde{I}_U(\mathbf{x}, \mathbf{h}, \mathbf{p}_m, \omega) \\ \widetilde{Q}_U(x, y, z = z_{\max}, \mathbf{p}_m, \omega) = 0 \end{array} \right. , \tag{17}$$

where $\widetilde{I}_D(\mathbf{x}, \mathbf{h}, \mathbf{p}_m, \omega)$ and $\widetilde{I}_U(\mathbf{x}, \mathbf{h}, \mathbf{p}_m, \omega)$ are the encoded SODCIGs. They are defined as:

$$\begin{array}{l} \widetilde{I}_D(\mathbf{x}, \mathbf{h}, \mathbf{p}_m, \omega) = \sum_{x_m, y_m} I_D(\mathbf{x}, \mathbf{h}, x_m, y_m)\beta, \\ \widetilde{I}_U(\mathbf{x}, \mathbf{h}, \mathbf{p}_m, \omega) = \sum_{x_m, y_m} I_U(\mathbf{x}, \mathbf{h}, x_m, y_m)\beta, \end{array} \tag{18}$$

where $\beta = e^{i\gamma(\mathbf{x}, x_m, y_m, \mathbf{p}_m, \omega)}$ is chosen to be the random phase-encoding function, with $\gamma(\mathbf{x}, x_m, y_m, \mathbf{p}_m, \omega)$ being a uniformly distributed random sequence in $\mathbf{x}$, $x_m$, $y_m$ and $\omega$; the variable $\mathbf{p}_m$ is the index of different realizations of the random sequence. Recursively solving Equations 16 and 17 gives us the encoded areal source data $\widetilde{Q}_D(\mathbf{x}, \mathbf{p}_m, \omega)$ and encoded areal receiver data $\widetilde{Q}_U(\mathbf{x}, \mathbf{p}_m, \omega)$, which can be collected at any depth.

In image-space wave-equation tomography, the image-space phase-encoded areal data sets are downward continued using the one-way wave equation. The background image is produced by cross-correlating the two wavefields and summing images for all realizations $\mathbf{p}_m$, as follows:

$$I_{\rm me}(\mathbf{x},\mathbf{h}) = \textstyle\sum_{\mathbf{p}_m,\omega} \overline{\widetilde{D}(\mathbf{x},\mathbf{p}_m,\omega)}\widetilde{U}(\mathbf{x},\mathbf{p}_m,\omega). \qquad (19)$$

The initial condition for modeling simultaneous events is set by regularly selecting SODCIGs in the prestack image. The amount of crosstalk in the image $I_{\rm me}(\mathbf{x},\mathbf{h})$ can be controlled by choosing a convenient sampling interval for SODCIGs used simultaneously for the modeling. For instance, if only one reflector is present and the correct velocity is used, no crosstalk is generated if the SODCIG interval is greater than twice the maximum subsurface offset of the prestack image. In the extreme case, when an incorrect velocity is used and the reflector's energy spreads through the whole range of subsurface offsets, crosstalk is not generated if the the SODCIG interval is greater than four times the maximum subsurface offset. In the presence of more than one reflector, crosstalk between reflectors occurs, regardless of the distance between SODCIGs input to modeling. By phase-encoding the reflectors, we can mitigate the crosstalk.

To phase-encode the reflectors it is necessary to pick some significant reflectors in the prestack migrated data. This implies a horizon-based approach for the prestack exploding-reflector modeling. In velocity-model updating, the idea of using some key reflectors to extract the residual-moveout information is an established strategy (Stork, 1992; Kosloff et al., 1996; Jiao et al., 2008).

The perturbed image is obtained by applying the chain rule to Equation 19. The slowness perturbation is computed by applying the adjoint of the tomographic operator, $\mathbf{T}^*$, to the image perturbation.

## NUMERICAL EXAMPLES

We test the image-space wave-equation tomography using image-space encoded data on a smoothed version of the Marmousi model, computed by applying a 200 m 2D-median filter to the slowness model. One-way data were synthesized considering a reflectivity computed from the Marmousi stratigraphic velocity model. We modeled 376 shots, ranging from 0 to 9000 m, with 24 m spacing. We used split-spread acquisition geometry, with maximum offset of 6600 m and receiver spacing of 24 m.

Figure 2(a) shows the true slowness model. The background velocity model is equal to the correct velocity model above 2400 m depth and above the anticline with apex at (x = 6000 m, z = 1850m). Therefore, the slowness perturbation is zero in this portion of the model. Below these horizons, the background model is characterized by a smoother version of the original Marmousi model, computed with a 400 m 2D-median filter and scaled down by a factor of 5%. Figure 2(b) shows the background slowness. By using this background slowness model, we assume that a layer striping approach has been used and that the model is accurately defined up to a certain horizon, as usually occurs in projects of velocity model building. The slowness perturbation, computed by taking the difference between the correct and background slownesses, is shown in Figure 3(a). In the part where the slowness perturbation is different from zero, the ratio between the true and the background slowness ranges approximately from 0.8 to 0.92 (Figure 3(b)). Notice that the minimum depth is 1500 m. Henceforth, all the figures will be displayed with a minimum depth of 1500 m.

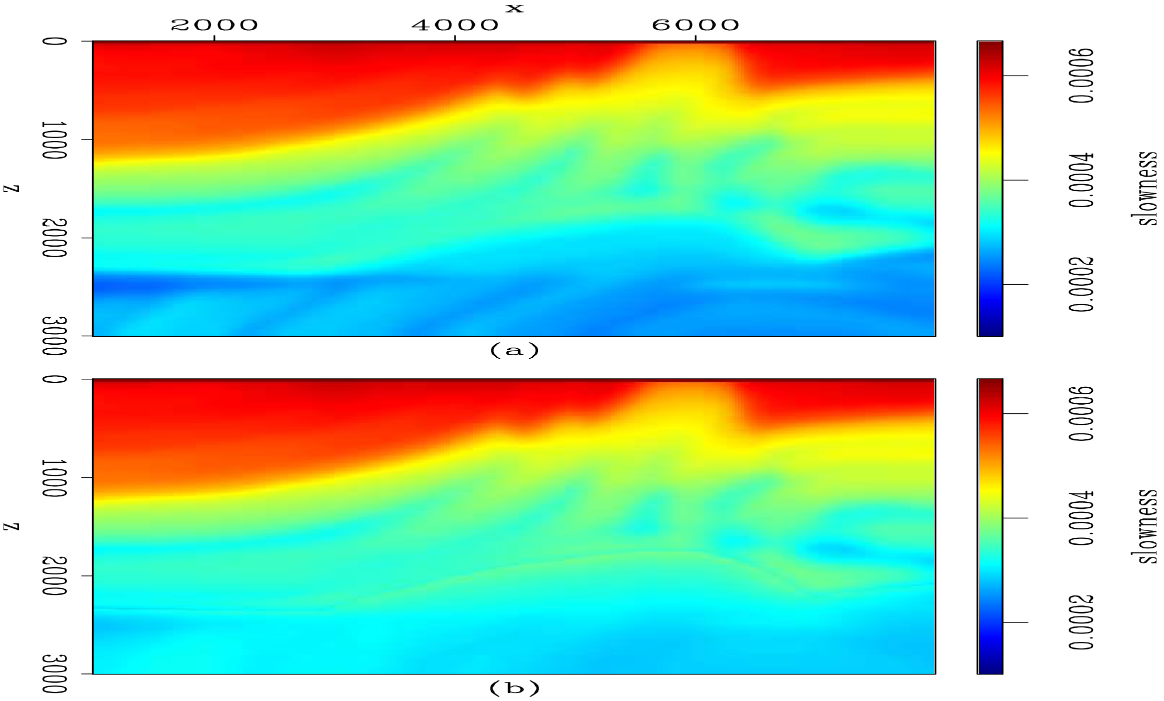


Figure 2: a) Correct slowness; b) Background slowness.[**ER**] claudio1/. islow

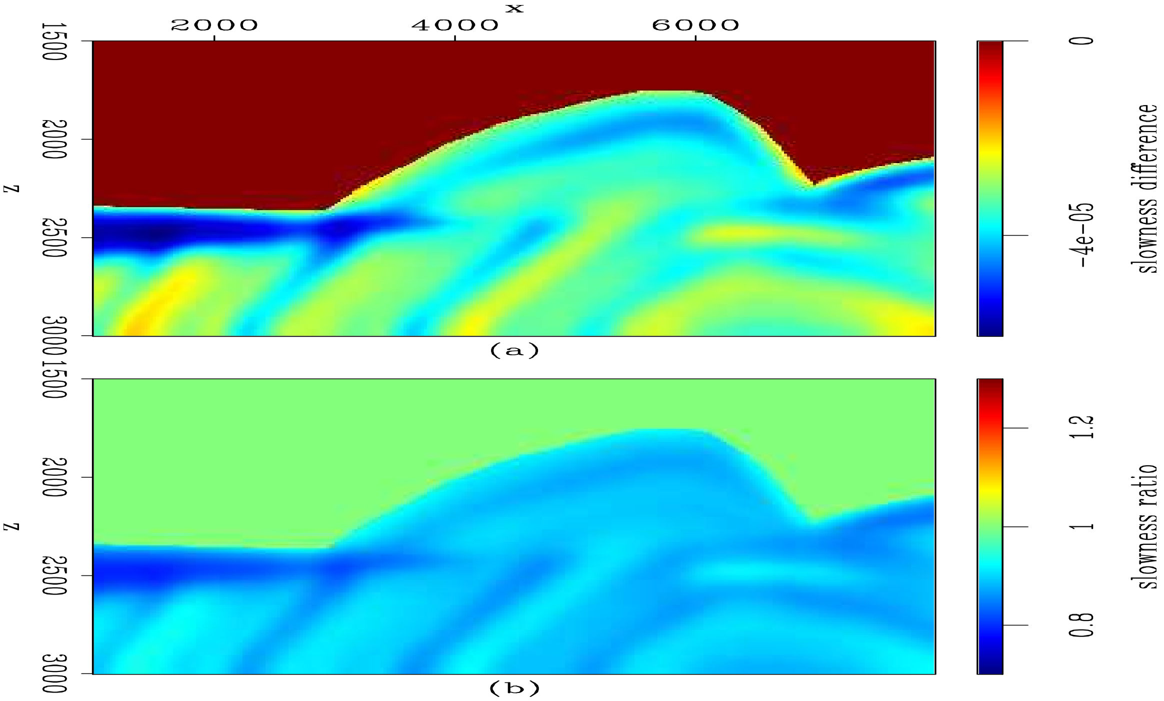


Figure 3: a) Slowness perturbation; b) Ratio between the true and the background slownesses.[**ER**] claudio1/. dslow

To compute the image-space phase-encoded data, we picked 10 reflectors in the non-zero slowness perturbation part, in the prestack image computed with the 376 original shots using the background slowness model. Figure 4 shows the background image (Figure 4a) computed with shot-profile migration. The panel on the left corresponds to the zero-subsurface-offset section, and the panel on the right is the SODCIG at CMP position 5500 m. Notice the effects of using an inaccurate background slowness. The reflector at (7000 m, 2500 m) is pulled up, as are the subjacent reflectors. In the SODCIG, the energy is not focused at the zero subsurface offset.

Figure 4(b) shows the picked reflectors used to model the image-space phase-encoded data. This image is used as input for the rotation of the reflectors in the SODCIGs with respect to the apparent geological dip, and the results are used as the initial conditions to model the image-space phase-encoded data, as discussed by Biondi (2006, 2007). Figure 5 shows the initial conditions for the prestack modeling. Figure 5(a) shows the initial condition for modeling the receiver wavefield, and Figure 5(b) shows the initial condition for modeling the source wavefield.

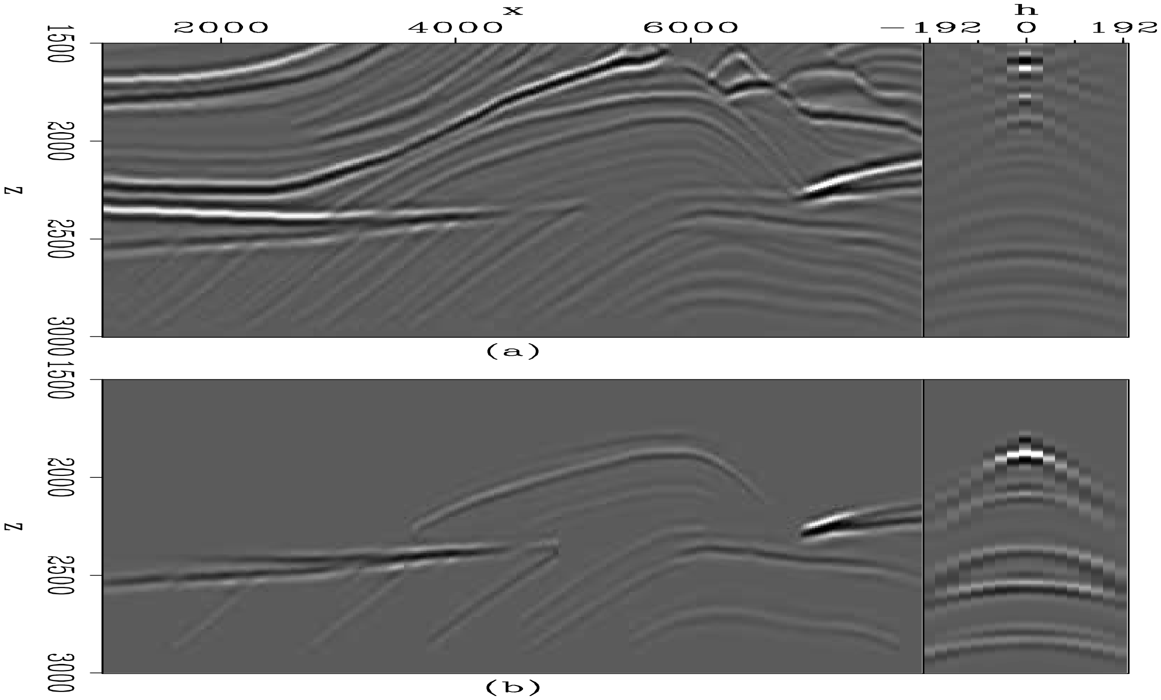


Figure 4: Zero-subsurface offset section (panel on the left) and SODCIG at 5500 m (panel on the right) showing: a) background image, and b) windowed image used to compute image-space-encoded data.[**CR**] claudio1/. bimg1

Two image-space phase-encoded data sets were synthesized using different parameters. One contains one random realization of phase-encoded areal shots initiated simultaneously with SODCIG sampling interval of 264 m and encoded according to the CMP position and reflector number, generating 11 areal shot gathers. The other data set corresponds to two random realizations modeled with SODCIG sampling interval of 840 m, composed of 70 areal shot gathers. Because in the velocity inversion we consider the maximum subsurface offset to be 192 m, this data set is expected to generate less crosstalk. In some comparisons, we use just one random realization (35 areal shots) of the second data set. We use the two random realizations when comparing the results of the non-linear optimization of the slowness model. Henceforth, the image-space phase-encoded data sets are called areal shots.

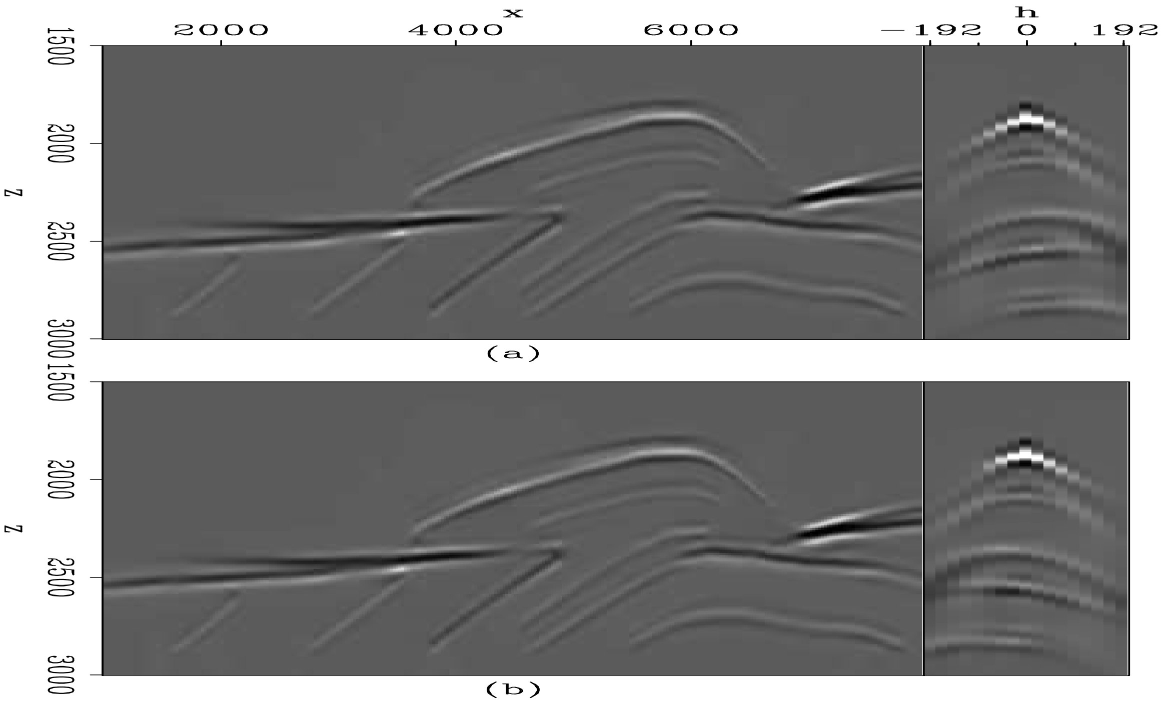


Figure 5: Zero-subsurface offset section (panel on the left) and SODCIG at 5500 m (panel on the right) showing: a) initial condition for modeling the receiver wavefield, and b) initial condition for modeling the source wavefield.[**CR**] claudio1/. rimg1

In the slowness optimization problem, we compute the image perturbation by applying the DSO operator, **O**, to the background image in the subsurface-offset domain. The corresponding objective functional is

$$J = \frac{1}{2}||\mathbf{O}\widehat{\mathbf{I}}||^2 = \frac{1}{2}||h\widehat{\mathbf{I}}||^2. \tag{20}$$

Since the DSO operator is independent of the slowness, the gradient of $J$ with respect to the slowness $\mathbf{s}$ is

$$\nabla J = \left( \frac{\partial \mathbf{I}}{\partial \mathbf{s}}\bigg|_{\mathbf{s}=\widehat{\mathbf{s}}} \right)^* \mathbf{O}^*\mathbf{O}\widehat{\mathbf{I}} = \mathbf{T}^*\mathbf{O}^*\mathbf{O}\widehat{\mathbf{I}}. \tag{21}$$

Given that the computation of the DSO objective functional is fully automated, it can be minimized by using quasi-Newton methods. Here, we specifically use the constrained L-BFGS algorithm (Nocedal and Wright, 2000).

To guarantee smoothness of the wave-equation tomography gradient, we use a B-spline representation with nodes located every 960 m in the $x$-direction and 16 m in the $z$-direction.

Figure 6 shows the image perturbation computed by applying the forward tomographic operator, **T**, and using the background slowness of Figure 2(b) and the known slowness perturbation of Figure 3. Figure 6a shows the image perturbation computed in the shot-profile domain for the 376 shots; Figure 6b shows the image perturbation computed in the image-space phase-encoded domain using 11 areal shots; and Figure 6c shows the image perturbation computed in the image-space phase-encoded domain using 35 areal shots. Notice that the dispersed crosstalk is stronger in Figure 6b than in Figure 6c.

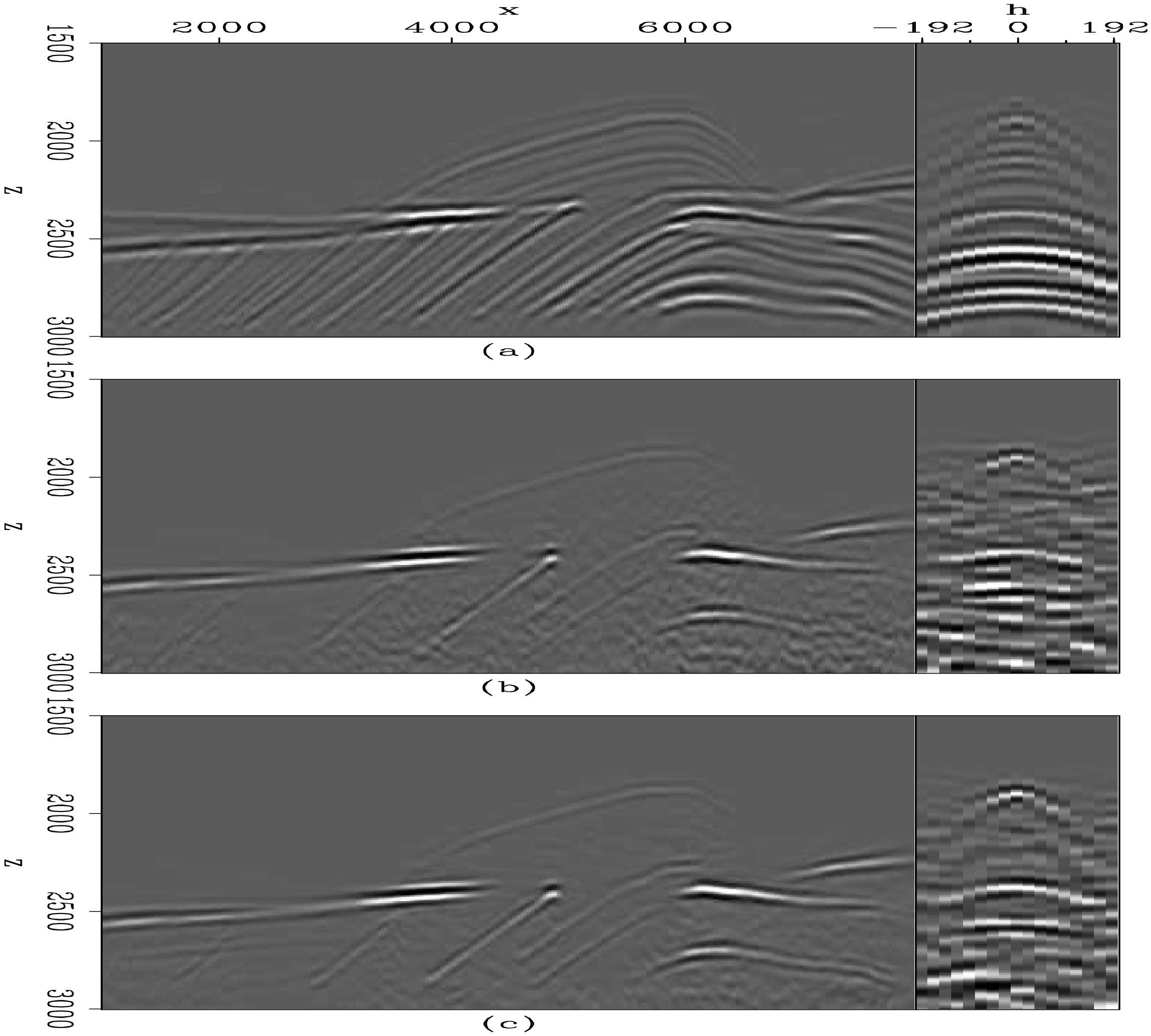


Figure 6: Zero-subsurface offset section (panel on the left) and SODCIG at 5500 m (panel on the right) showing: a) image perturbation in the shot-profile domain; b) image perturbation computed with 11 areal shots; and c) image perturbation computed with 35 areal shots.[**CR**] claudio1/. dimg1

Figure 7 illustrates the normalized slowness perturbations obtained by applying the adjoint tomographic operator $\mathbf{T}^*$ to the image perturbations of Figure 6. Compare with the correct slowness perturbation of Figure 3. Figure 7a is the predicted slowness perturbation found by back-projecting Figure 6a using all 376 shot gathers; Figure 7b shows the back-projection of Figure 6b using 11 areal shots; and Figure 7c shows the back-projection of Figure 6c using 35 areal shots. Notice that we do not use a B-spline representation for the slowness perturbations. In general, the predicted slowness perturbation with image-space phase-encoded gathers shows a structure similar to that obtained with the original shot gathers. The differences can be credited, at first order, to the occurrence of residual crosstalk in the image-space phase-encoded perturbed image and to a sub-optimal number of selected reflectors for the prestack exploding-reflector modeling.

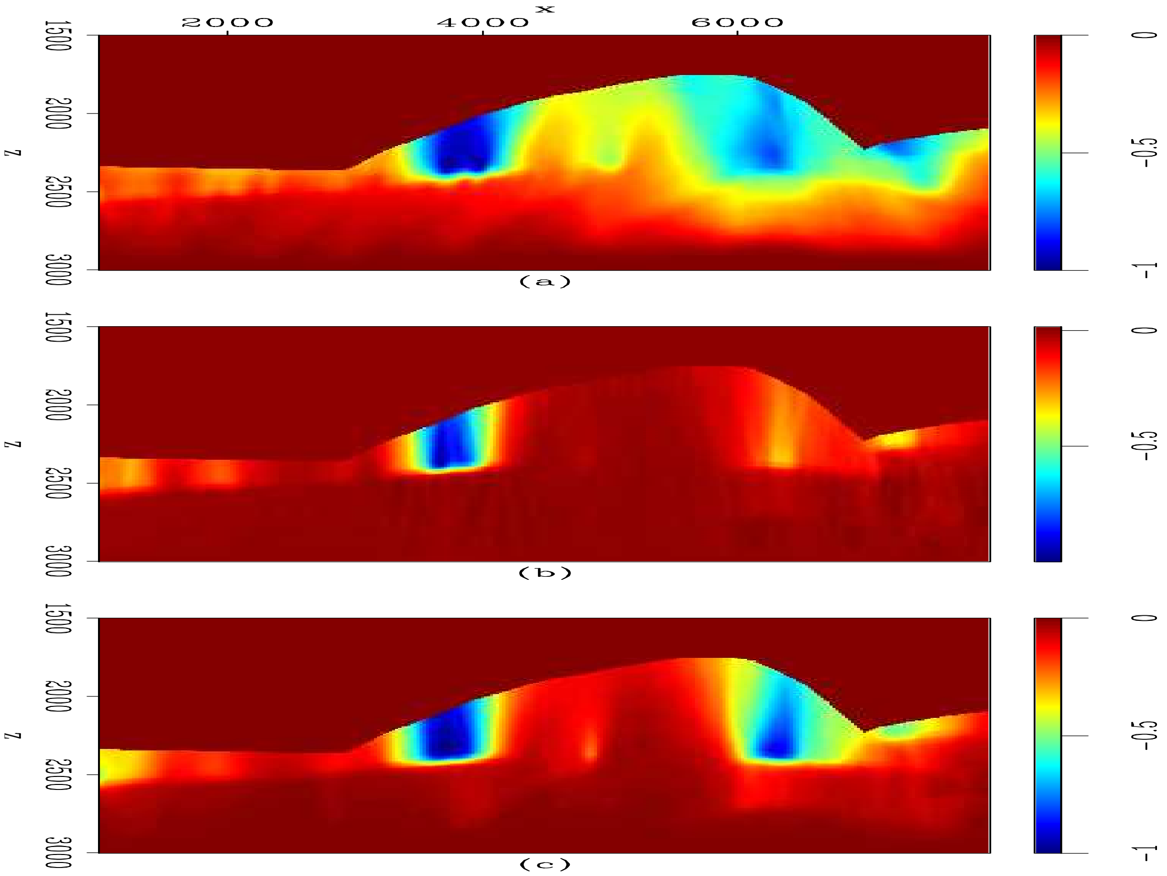


Figure 7: Normalized slowness perturbation obtained by applying the adjoint tomographic operator $\mathbf{T}^*$ on the image perturbations in Figure 6. a) Slowness perturbation computed from Figure 6a. b) Slowness perturbation computed from Figure 6b. c) Slowness perturbation computed from Figure 6c.[**CR**] claudio1/. dsadj

Finally, we compare the optimized slowness models with the correct slowness model of Figure 2(a). After 5 non-linear iterations for both the 11 areal shots and 35 areal shots (one random realization) of the 70-gather image-space phase-encoded data set the optimization stopped because the difference between the objective functional of successive iterations was smaller than the machine precision. The number of function evaluations was 28 for the 11 areal shots, and 27 for the 35 areal shots. We also computed 2 non-linear iterations with a total of 6 function evaluations using the two random realizations of the 70 areal shots. To verify the accuracy of the resulting optimized slowness models, we also migrated the original shot gathers with the three optimized slownesses and also with the correct slowness.

Figure 8 displays the evolution of the objective functional with the non-linear iterations for the 11 areal shots (Figure 8(a)) and 35 areal shots (Figure 8(b)). For comparison, the value of the objective functional for the true velocity is also shown as dashed lines. The values are normalized according to the value of the objective functional of the first iteration for each case. The objective functional was reduced in 22% and 36% for the 11 areal shots and for the 35 areal shots, respectively. Notice that those values are 23% and 47%, respectively, when using the true slowness model. The smaller difference between the final optimized value of the objective functional and the objective functional computed with the true slowness model for the optimization with the 11 areal shots, can be partially credited to the more severe crosstalk generated by this data set than the 35 areal shots. Even if the correct slowness model is used, residual crosstalk is amplified when applying the DSO operator.

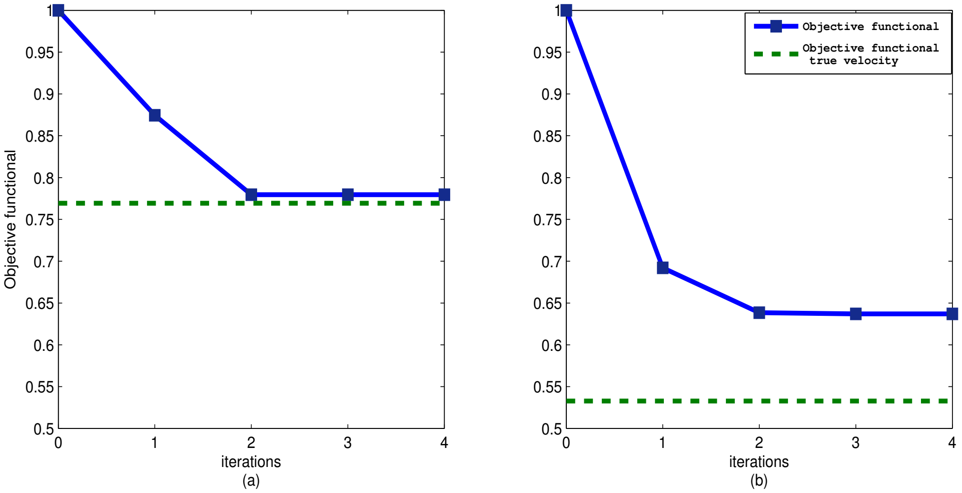


Figure 8: Evolution of the objective functional with the non-linear iterations. The dashed lines represent the value of the objective functional for the true slowness model. a) Normalized objective functional for the 11 areal shots. b) Normalized objective functional for the 35 areal shots.[**NR**] claudio1/. plot

Figure 9 shows the optimized slownesses and, for comparison purposes, the true slowness. Figure 9(a) displays the slowness model; Figure 9(b) is the slowness model obtained with the 11-gather image-space phase encoded data; Figure 9(c) is the slowness model obtained with the 35-gather image-space phase encoded data; and Figure 9(d) is the slowness model obtained with the 70-gather image-space phase encoded data. In general, the predicted slownesses are reasonable. The predicted slowness using the 11 areal shots shows slightly lower values than the other two predicted slownesses. Notice that the detailed slowness variation present in the true slowness is not recovered in the optimized slownesses, due to the B-spline representation of the gradient of the wave-equation tomography objective functional. In addition, as we are solving for the deeper portion of the model with dipping reflectors, it is likely that deficient illumination prevents us to obtain a more accurate slowness model. However, the slowness model obtained with the 70 areal shots recovers the low slowness values on the left of model better than the other two predicted slownesses.

Figure 10 shows the histograms of the ratio between the true and background interval slowness (continuous line) and between the true and predicted interval slownesses obtained with the 11 (fine dash), 35 (fine dot), and 70 areal shots (large dash) below the depth of 2400 m.

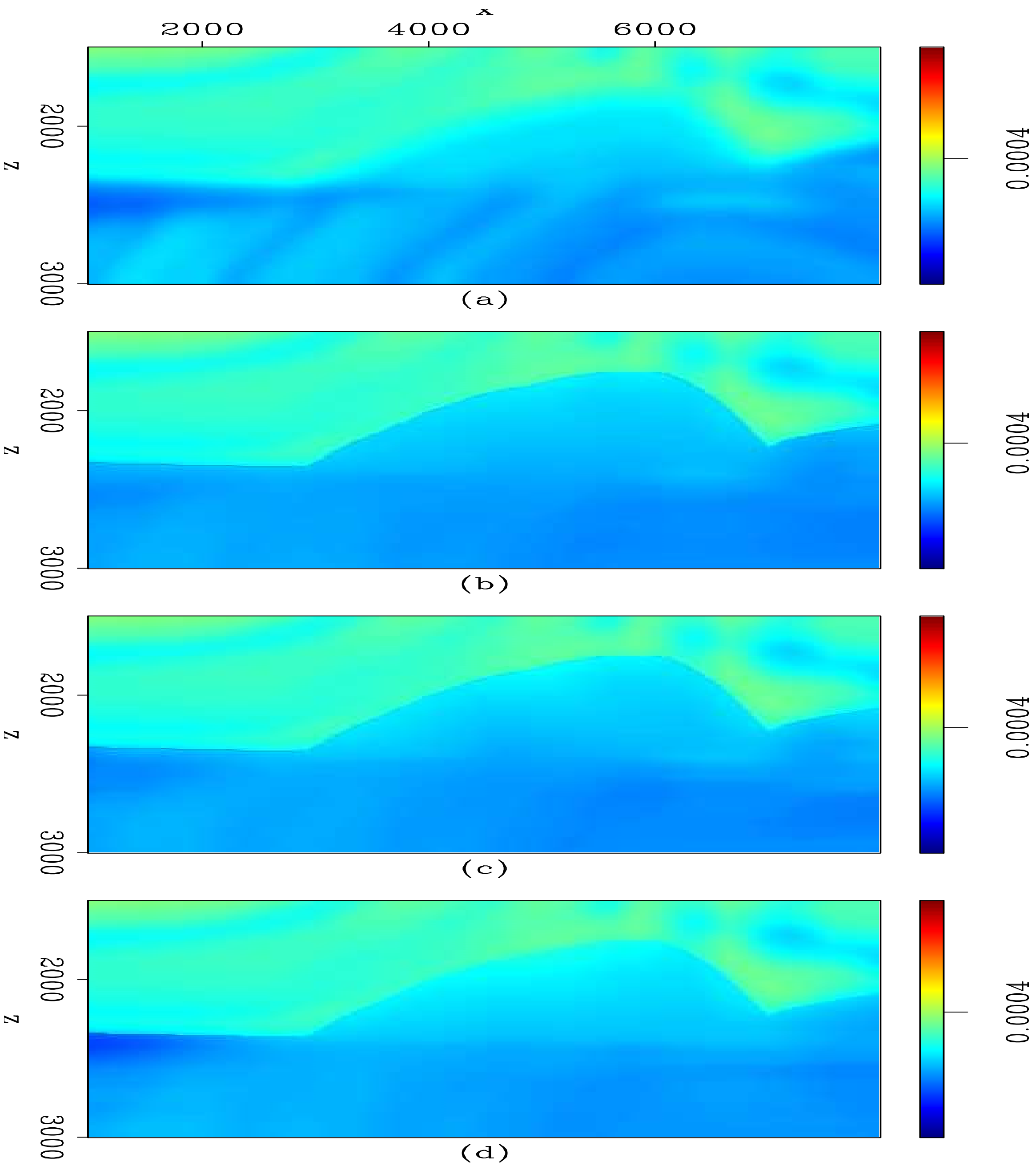


Figure 9: True and optimized slownesses. a) True slowness model; b) Slowness model obtained with the 11-gather image-space phase-encoded data. c) Slowness model obtained with the 35gather -image-space phase-encoded data. d) Slowness model obtained with the 70-gather image-space phase-encoded data.[**CR**] claudio1/. sfperm

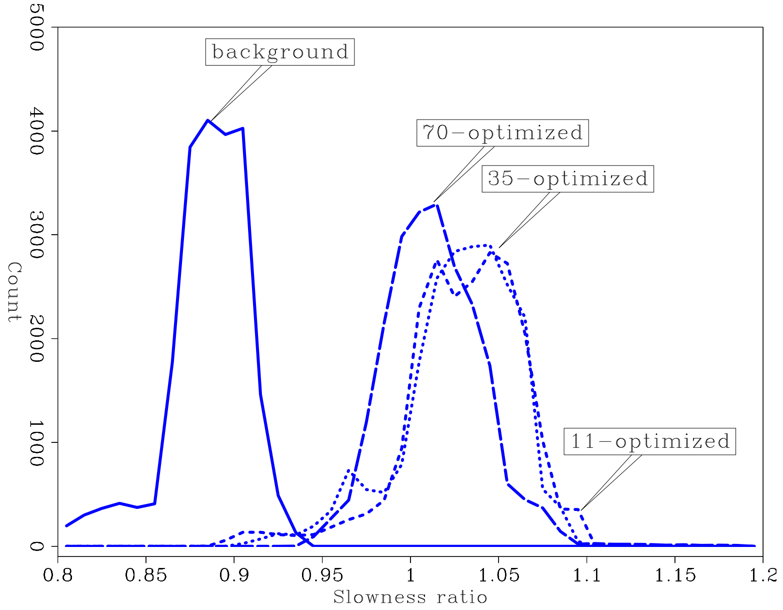


Figure 10: Histograms of the slowness ratios between the true and background interval slowness (continuous line) and between the true and predicted interval slownesses obtained with the 11 (fine dash), 35 (fine dot), and 70 areal shots (large dash).[**CR**] claudio1/. hsfperm

The mean and standard deviation of the corresponding distributions are summarized in Table 1. In general, the predicted slownesses vary between 94% to 100% of the true slowness. The background slowness varies between 110% to 116% of the true slowness.

| Slowness ratio | mean | $\sigma$ |
|---|---|---|
| background | 0.884 | 0.025 |
| 11 gathers | 1.030 | 0.033 |
| 35 gathers | 1.027 | 0.032 |
| 70 gathers | 1.013 | 0.027 |

Table 1: Mean and standard deviation of the interval slowness ratio.

Figure 11 displays the zero-subsurface-offset section after migration of the 376 original shot gathers using the true slowness model (Figure 11(b)), the predicted slowness using 11 areal shots (Figure 11(c)), the predicted slowness using 35 areal shots (Figure 11(d)), and the predicted slowness using 70 areal shots (Figure 11(e)). For comparison, we also display in Figure 11(a) the zero-subsurface-offset section after migration with the background slowness of Figure 2(b). Notice that reflectors in the central portion of Figure 11(a) are pulled up when comparing to Figure 11(b). The image obtained with the optimized slowness model computed with the 11 areal shots (Figure 11(c))presents pushed down reflectors around (4000 m, 2500 m) as a consequence of the lower values of the optimized slowness. In addition, in this image the undulating character of the reflector at (7000 m, 2600 m) reflects some velocity inaccuracy, when compared to Figures 11(b) and (e).

From top to bottom, Figure 12 displays SODCIGs at 1500 m, 3500 m, 5500 m and 7500 m after migration of the 376 original shot gathers, using the background slowness of Figure 2(b) in the first row, using the true slowness model in the second row, using the predicted slowness with 11 areal shots in the third row, using the predicted slowness with 35 areal shots in the fourth row, and using the predicted slowness with 70 areal shots in the fifth row. The subsurface-offset ranges from -192 m to 192 m. The analysis of the SODCIGs in Figures 12(c) to (e) shows that better focusing is achieved when more image-space phase-encoded gathers are used in the wave-equation tomography.

Figure 13 displays the angle-domain common-image gathers (ADCIGs) taken at the same CMP position as SODCIGs of Figure 12. From top to bottom, Figure 13 displays ADCIGs after migration using the background slowness in the first row, using the true slowness model in the second row, using the predicted slowness with 11 areal shots in the third row, using the predicted slowness with 35 areal shots in the fourth row, and using the predicted slowness with 70 areal shots in the fifth row. Notice that migration with the predicted slowness using 70 areal shots shows virtually no residual moveout. For the case of predicted slowness using 11 and 35 areal shots some residual moveout occurs for CMP position 5500 m. As can be seen if Figure 13, the angular coverage for the dipping deep reflectors we consider in the slowness inversion ranges from $-15°$ to $15°$. For the reflectors at the central portion of the model it varies between $-25°$ to $25°$. This limited angular coverage decreases the resolution of the slowness estimate.

The accuracy of the optimized slowness improves when using more phase-encoded gathers in the wave-equation tomography, or, in other words, when the crosstalk in the perturbed image is less severe, as Figures 6 and 7 suggest. Figure 14 shows the perturbed image computed by applying the DSO operator on the image migrated with the background slowness of Figure 2(b). The panel on the left corresponds to the subsurface-offset -144 m and the panel on the right is the SODCIG taken at 5500 m. Figure 14(a) shows the perturbed image using 11 areal shots; Figure 14(b) shows the perturbed image using 35 areal shots; and Figure 14(c) shows the perturbed image using 70 areal shots. Notice how the signal-to-noise ratio improves as more phase-encoded gathers are used. The SODCIG of the perturbed image of Figure 14(a) presents coherent events, related to unattenuated crosstalk, curving upward at z = 2700 m; these events are not present in Figures 14(b) and (c). If these events are sufficiently incoherent along the subsurface-offset sections, a two-dimensional filter could be applied to attenuate them. In that case, a new objective function should be defined. This deserves future investigation.

## CONCLUSIONS

We present a cost-effective method to perform image-space wave-equation tomography using image-space phase-encoded shot gathers. One important advantage is that we are able to synthesize a much smaller data set while still keeping necessary velocity information for migration velocity analysis; hence the computational cost of performing image-space wave-equation tomography can be significantly reduced. Our results show that using the image-space phase-encoded wavefields in the image-space wave-equation tomography problem can provide reliable optimized slowness model, given that crosstalk is sufficiently attenuated.

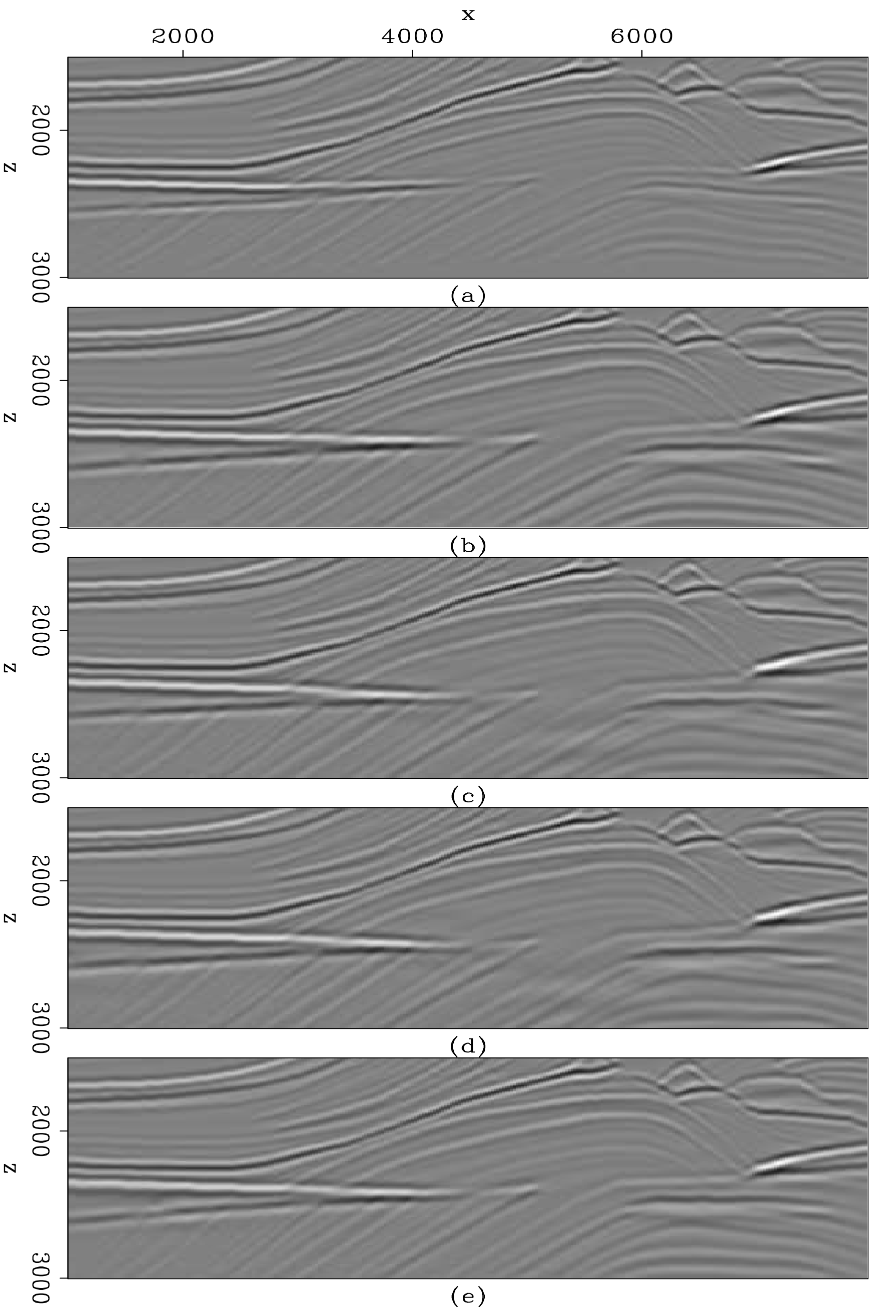


Figure 11: Zero-subsurface-offset section after migration of the 376 original shot gathers using: a) the true slowness model; b) the predicted slowness model of Figure 9(a); c) the predicted slowness model of Figure 9(b); and d) the predicted slowness model of Figure 9(c).[**CR**] claudio1/. fimg1

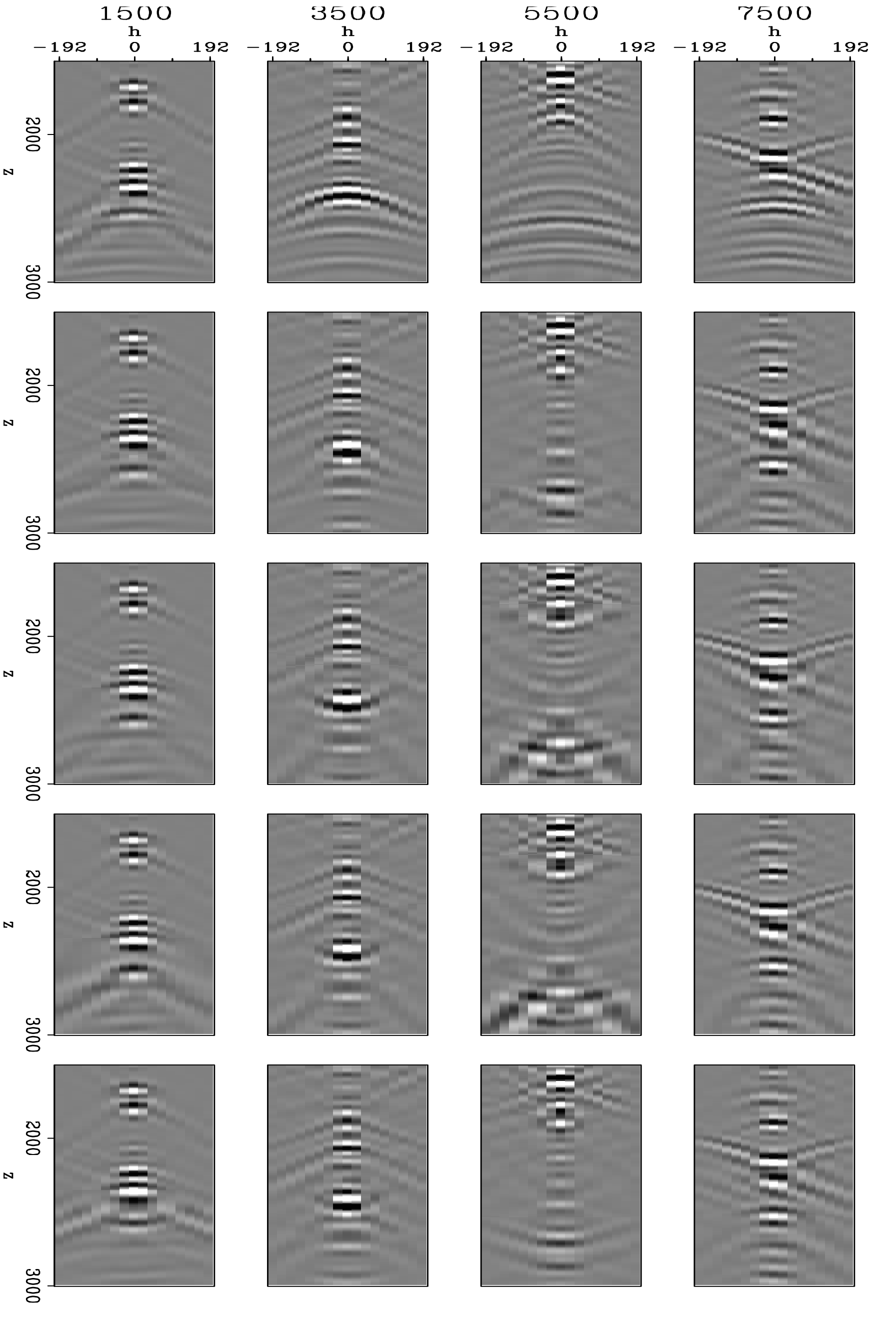


Figure 12: Subsurface-offset gathers after migration of the 376 original shot gathers. From top to bottom: in the first row, using the background slowness model; the second row, using the true slowness model; in the third row, using the predicted slowness model of Figure 9(b); in the fourth row, using the predicted slowness model of Figure 9(c); and in the fifth row, using the predicted slowness model of Figure 9(d).[**CR**] claudio1/. fimg11

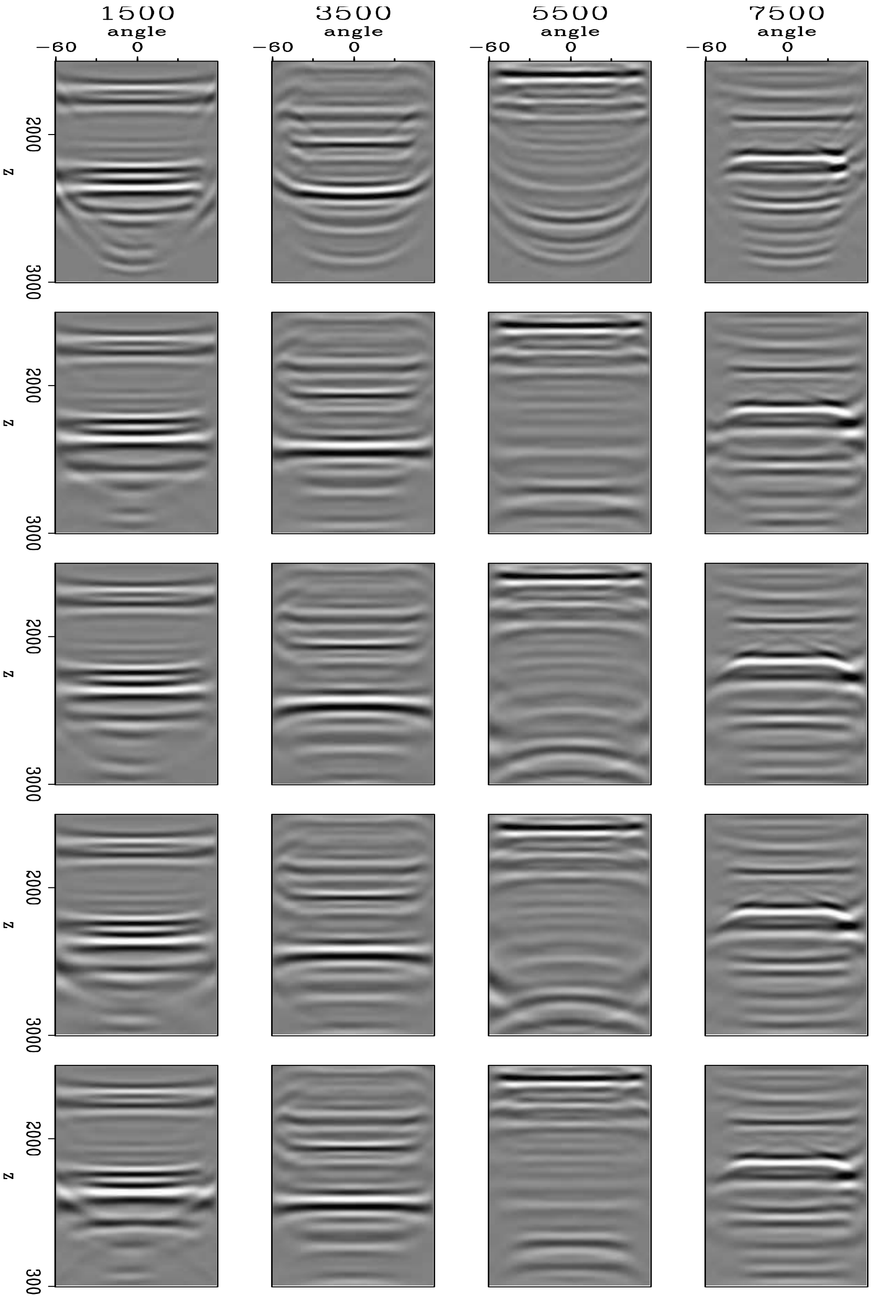


Figure 13: ADCIGs after angle transformation of the SODCIGs of Figure 12. From top to bottom: in the first row, using the background slowness model; the second row, using the true slowness model; in the third row, using the predicted slowness model of Figure 9(b); in the fourth row, using the predicted slowness model of Figure 9(c); and in the fifth row, using the predicted slowness model of Figure 9(d).[**CR**] claudio1/. fang1

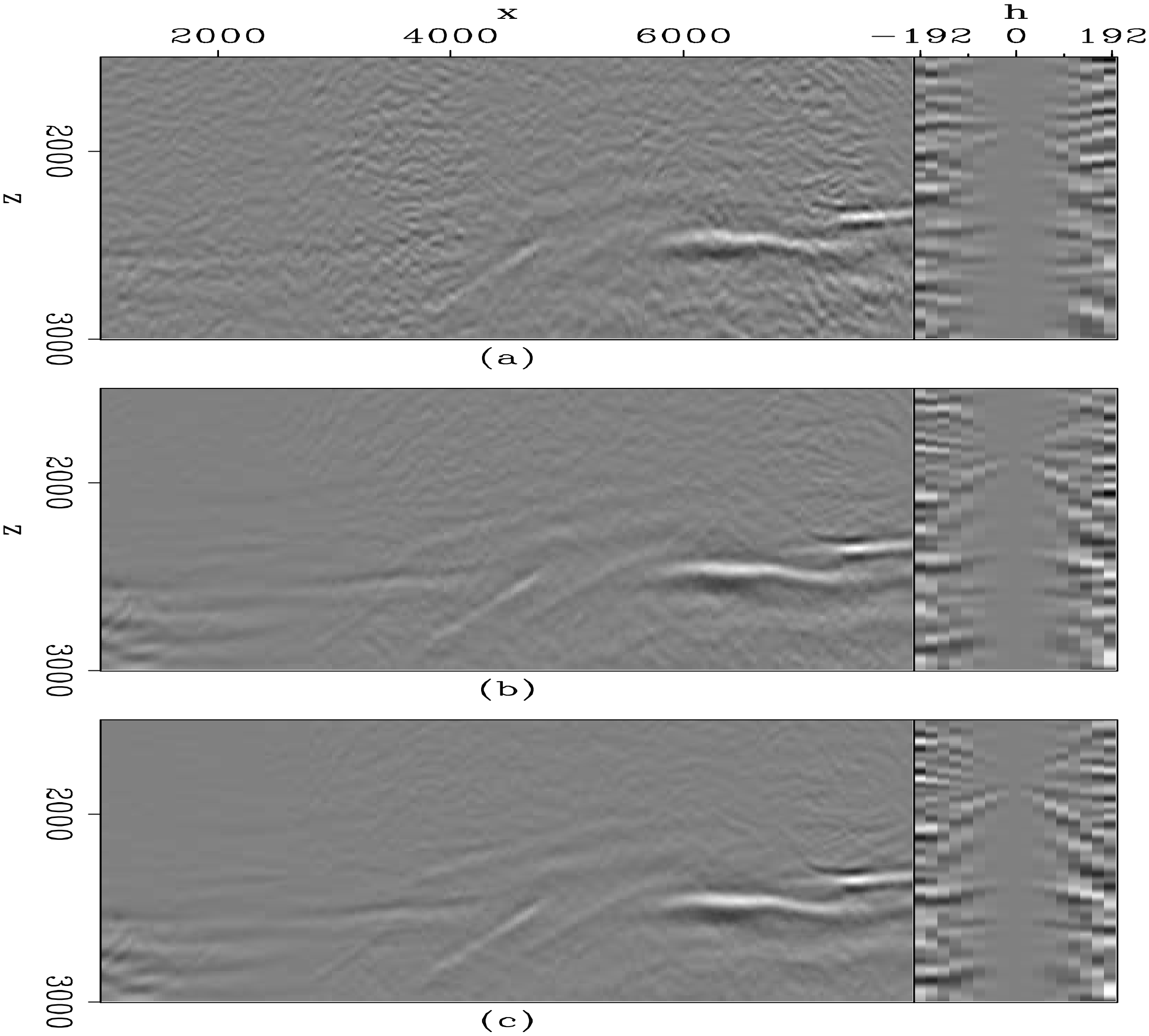


Figure 14: Perturbed images computed with the DSO operator. a) Perturbed image using 11 areal shots. b) Perturbed image using 35 areal shots. c) Perturbed image using 70 areal shots.[**CR**] claudio1/. dso

## REFERENCES

Biondi, B., 2006, Prestack exploding-reflectors modeling for migration velocity analysis: 76th Ann. Internat. Mtg., Expanded Abstracts, 3056–3060, Soc. of Expl. Geophys.

———, 2007, Prestack modeling of image events for migration velocity analysis: **SEP-131**, 101–118.

———, 2008, Automatic wave-equation migration velocity analysis: **SEP-134**, 65–78.

Biondi, B. and P. Sava, 1999, Wave-equation migration velocity analysis: 69th Ann. Internat. Mtg., Expanded Abstracts, 1723–1726, Soc. of Expl. Geophys.

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

Guerra, C. and B. Biondi, 2008, Prestack exploding reflector modeling: The crosstalk problem: **SEP-134**, 79–92.

Jiao, J., D. R. Lowrey, J. F. Willis, and R. D. Martínez, 2008, Practical approaches for subsalt velocity model building: Geophysics, **73**, VE183–VE194.

Kosloff, D., J. Sherwood, Z. Koren, E. Machet, and Y. Falkovitz, 1996, Velocity and interface depth determination by tomography of depth migrated gathers: Geophysics, **61**, 1511–1523.

Nocedal, J. and S. Wright, 2000, Numerical optimization: Springer Verlag,New York.

Romero, L. A., D. C. Ghiglia, C. C. Ober, and S. A. Morton, 2000, Phase encoding of shot records in prestack migration: Geophysics, **65**, 426–436.

Sava, P., 2004, Migration and Velocity Analysis by Wavefield Extrapolation: PhD thesis, Stanford University.

Shen, P., 2004, Wave-equation Migration Velocity Analysis by Differential Semblance Optimization: PhD thesis, Rice University.

Shen, P. and W. W. Symes, 2008, Automatic velocity analysis via shot profile migration: Geophysics, **73**, VE49–VE59.

Stork, C., 1992, Reflection tomography in the postmigrated domain: Geophysics, **57**, 680–692.

Tang, Y., C. Guerra, and B. Biondi, 2008, Image-space wave-equation tomography in the generalized source domain: **SEP-136**, 1–22.

Tarantola, A., 1987, Inverse problem theory: Methods for data fitting and model parameter estimation: Elsevier.

Vigh, D. and E. W. Starr, 2008, 3d prestack plane-wave, full-waveform inversion: Geophysics, **73**, VE135–VE144.

Whitmore, N. D., 1995, An Imaging Hierarchy for Common Angle Plane Wave Seismogram: PhD thesis, University of Tulsa.

Woodward, M. J., 1992, Wave-equation tomography: Geophysics, **57**, 15–26.

# Seismic tomography with co-located soft data

*Mohammad Maysami and Robert G. Clapp*

## ABSTRACT

There is a wide range of uncertainties present in seismic data. Limited subsurface illumination is also common, specially in areas with salt structures. These shortcomings are only a few of many different reasons that makes seismic tomography an under-determined problem with a large null space. We can use additional information to reduce the uncertainty and constrain this large null space. The additional information, also known as co-located soft (secondary) data, can be the result of integrating a non-seismic data from the same subsurface area. A measure of structural similarity between the two given data fields can create a link between the different types of data. We use cross-gradient functions to incorporate this structural information, given by secondary data, into the inverse problem as a constraint.

## INTRODUCTION

Seismic data contain a wide range of uncertainties which directly affects the quality of seismic images. Previous studies have tried to extract more information from raw seismic data to reduce the uncertainty in the seismic-imaging problem (Yilmaz, 2001; Aki and Richards, 2002). Since velocity analysis plays a fundamental role in seismic imaging, uncertainties in velocities lead to significant inaccuracies in seismic images. Without an accurate velocity estimate, seismic reflectors are misplaced, the image is unfocused, and seismic images can easily mislead earth scientists (Claerbout, 1999; Clapp, 2001). Defining a reliable velocity model for seismic imaging is a difficult task, especially when sharp lateral and vertical velocity variations are present. Moreover, velocity estimation becomes even more challenging when seismic data are noisy (Clapp, 2001).

In areas with significant lateral velocity variations, reflection tomography methods, where traveltimes are mapped to slowness, are often more effective than conventional velocity-estimation methods based on measurements of stacking velocities (Biondi, 1990; Clapp, 2001). However, reflection tomography may also fail to converge to a geologically reasonable velocity estimation when the wavefield propagation is complex.

Unfortunately, the reflection tomography problem is ill-posed and under-determined. Furthermore, it may not converge to a realistic velocity model without *a priori* information, e.g., regularization constraints and other types of geophysical properties, in addition to seismic data (Clapp, 2001). Better velocity estimation can be achieved by integrating co-located soft data, such as non-seismic geological data, in the reflection tomography problem.

Lack of an analytical relationship between different measured geological properties limits our ability to use co-located soft data. Besides the conventional probabilistic relations, similarity-measurement tools can be used to enforce the structural information contained in soft data into seismic velocity estimates. Based on these tools, differences in two images are classified as structural differences and non-structural differences. Since gradient fields are a good choice for geometrical (structural) comparisons, the cross-gradient function is one useful similarity-measurement tool. This is true because the variations of geophysical properties can be described by a magnitude and a direction (Gallardo and Meju, 2004, 2007).

Here we use the cross-gradient function to integrate a given set of soft data—the resistivity field measured by magnetotelluric (MT) sounding in our case—into the reflection tomography problem. This integration requires consideration of differences in frequency in seismic and resistivity data. In the following sections we study the behavior of cross-gradient functions in different cases and then give an overview of how an understanding of these differences can be used to improve velocity estimates given by seismic tomography.

## THE CROSS-GRADIENT FUNCTION: A STRUCTURAL SIMILARITY MEASURE

Integration of soft data into the seismic tomography problem can reduce model uncertainty and result in a better velocity estimation, especially in areas with complex structure. Different geophysical methods probs the same structures in the Earth's subsurface. Among the techniques for integrating different types of geological data, structural similarity-measurement tools may be a good choice for our tomography problem. The cross-gradient function is one tool that measures the structural similarity between any two fields. Following Gallardo and Meju (2004), we can define the cross-gradient function for the tomography problem as

$$\mathbf{g} = \nabla \mathbf{r} \times \nabla \mathbf{s}, \tag{1}$$

where $\mathbf{r}$ and $\mathbf{s}$ can represent any two model parameters. In our case, they represent resistivity and slowness, respectively. Zero values of the cross-gradient function correspond to points where spatial changes in both geophysical properties, i.e., $\nabla \mathbf{r}$ and $\nabla \mathbf{s}$, align. However, the function is also zero where the magnitude of spatial variations of either field is negligible, e.g., where either property is smooth. Note that the cross-gradient function is a non-linear function of $\mathbf{r}$ and $\mathbf{s}$ if both are unknowns. In a 2-D problem, $\mathbf{g}$ simplifies to a scalar function at each point, given by

$$\mathbf{g} = \frac{\partial \mathbf{s}}{\partial x}\ \frac{\partial \mathbf{r}}{\partial z} - \frac{\partial \mathbf{s}}{\partial z}\ \frac{\partial \mathbf{r}}{\partial x}\ , \tag{2}$$

where the model parameters are given in the $x - z$ plane. To compute the cross-gradient function, we can further simplify it by using first-order forward-differences approximations of the first derivative operators.

Figures 1(a) and 1(b) show the smooth Marmousi synthetic 2-D velocity model (Versteeg and Grau, 1991) and its cross-gradient with itself, respectively. The cross-gradient of a field with itself is called the auto-gradient hereafter. Note that the auto-gradient of a field should be zero everywhere; however, since the figures are prepared with a first-order linear approximation of the cross-gradient function, it is not zero, especially in areas with sharp edges.

Although we expect different types of geophysical methods to result in similar structural maps, in practice each method maps the subsurface through different filters and frequency contents. Typical frequencies in magnetotelluric data are much lower than those of seismic data (Kaufman and Keller, 1981). This difference in the frequency content of two fields may affect how the cross-gradient represents the structural similarity of two fields. To investigate the effect of different spatial frequency content, we prepared Figure 1, in which the cross-gradient of the Marmousi velocity model and a smooth version of it is computed. In Figure 1(d), we have increased the smoothing factor. This increase is equivalent to a lower cut-off spatial frequency for a lowpass filter. Note that because of the relatively sharp edges in the original velocity model, the cross-gradients in Figures 1(c) and 1(d) seem to include some structure as well as higher amplitudes as compared with Figure 1(b). However, this synthetic example is an extreme case of complexity and sharp edges. As shown by the results for the Pillow velocity model in Figure 2, in simpler cases of subsurface structure, the cross-gradient with a smooth version of the velocity model leads to an acceptable similarity indicator. The amplitude may be improved by using a higher-order linear approximation of the cross-gradient computation. These figures in general may imply that the cross-gradient function can be used as a constraint for joint data inversion problems or to integrate *a priori* information from other fields into the seismic tomography problem.

## REFLECTION TOMOGRAPHY

By definition, tomography is an inverse problem, in which a field is reconstructed from its known linear path integrals, i.e., projections (Clayton, 1984; Iyer and Hirahara, 1993). Tomography can be represented by a matrix operator **T**, which integrates slowness along the raypath. The tomography problem can then be stated as

$$\mathbf{t} = \mathbf{T}\ \mathbf{s}, \tag{3}$$

where **t** and **s** are traveltime and slowness vector, respectively (Clapp, 2001). The tomography operator is a function of the model parameters, since the raypaths depend on the velocity field. Consequently, the tomography problem is non-linear. A common technique to overcome this non-linearity is to iteratively linearize the operator around an *a priori* estimation of the slowness field $\mathbf{s}_0$ (Biondi, 1990; Etgen, 1990; Clapp, 2001). The linearization of the tomography problem by using a Taylor expansion is given by

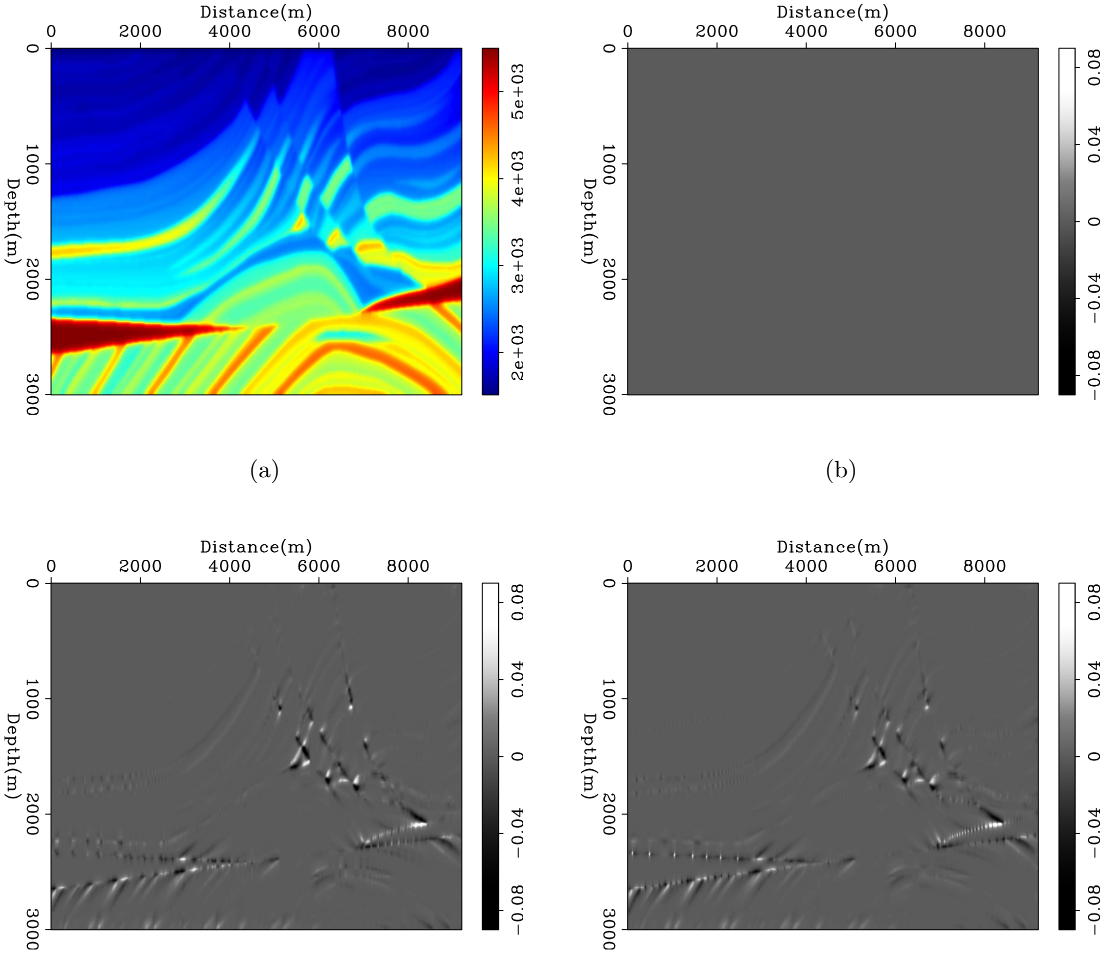
$$\mathbf{t} \approx \mathbf{T}\mathbf{s}_0 + \frac{\partial \mathbf{T}}{\partial \mathbf{s}}\Big|_{\mathbf{s}=\mathbf{s}_0} \Delta \mathbf{s}. \tag{4}$$

Here, $\Delta \mathbf{s} = \mathbf{s} - \mathbf{s}_0$ represents the update in the slowness field with respect to the *a priori* slowness estimation, $\mathbf{s}_0$. Equation 4 can be simplified as

$$\Delta \mathbf{t} = \mathbf{t} - \mathbf{T}\mathbf{s}_0 \approx \mathbf{T}_L \Delta \mathbf{s}, \tag{5}$$

where $\mathbf{T}_L = \frac{\partial \mathbf{T}}{\partial \mathbf{s}}\big|_{\mathbf{s}=\mathbf{s}_0}$ is a linear approximation of **T**. A second, but not lesser, difficulty arises because the locations of reflection points are unknown and are a function of the velocity field (van Trier, 1990; Stork, 1992).

Clapp (2001) attempts to resolve some of the difficulties caused by the non-linearity of the seismic tomography problem by introducing a new tomography operator in the tau domain and by using steering filters. In addition to geological models, other types of geophysical data can also be extremely important for yielding improved velocity estimates. In the following section, we show how the cross-gradient function can be used to add constraints to the seismic tomography problem in order to decrease the uncertainties in the estimated velocity model.



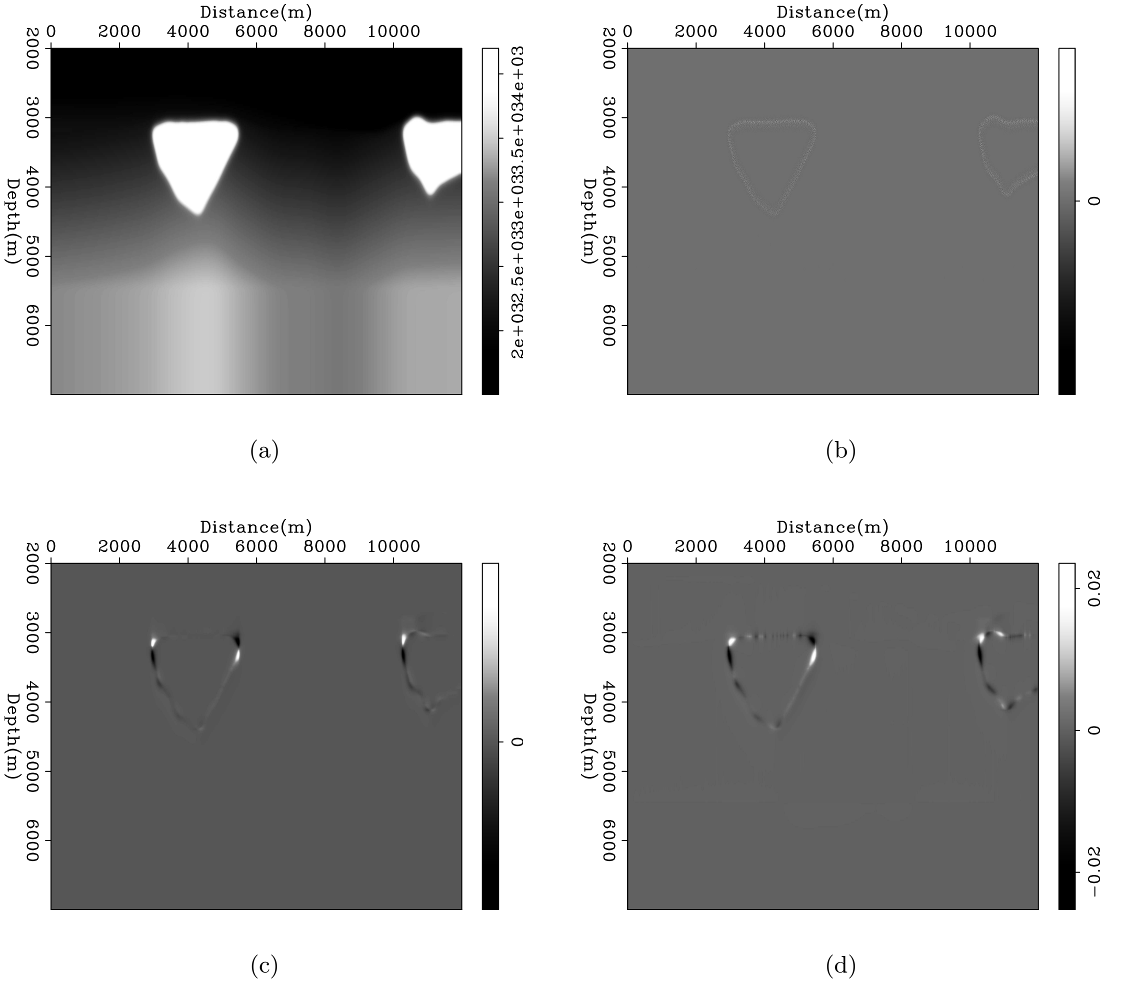
(a)

(b)

(c)

(d)

Figure 1: Frequency sensitivity of the cross-gradient function: **(a)** The Marmousi velocity model; **(b)** its auto-gradient. Cross-gradient values of the Marmousi velocity model and its **(c)** smooth and **(d)** very smooth copies. **[ER]**
mohammad1/. marm-S0vel,marm-S0xg,marm-S1xg,marm-S2xg



(a)

(b)

(c)

(d)

Figure 2: Frequency sensitivity of the cross-gradient function: **(a)** The Pillow velocity model; **(b)** its auto-gradient. Cross-gradient values of the Pillow velocity model and its **(c)** smooth and **(d)** very smooth copies. [**ER**] mohammad1/. pilw-S0vel,pilw-S0xg,pilw-S1xg,pilw-S2xg

## APPLICATION OF THE CROSS-GRADIENT FUNCTION IN SEISMIC TOMOGRAPHY

Figure 3 shows a velocity map and corresponding resistivity map of a synthetic 2-D model. That includes a water velocity of about 1.5 $\frac{km}{s}$ at the top and a semi-circular fault in the middle of the ocean bottom. There are also laterally smooth velocity anomalies in the model. The resistivity profile and velocity profile are connected using the *Archie/time-average* cross-property relation (Carcione et al., 2007) with arbitrary parameter values.

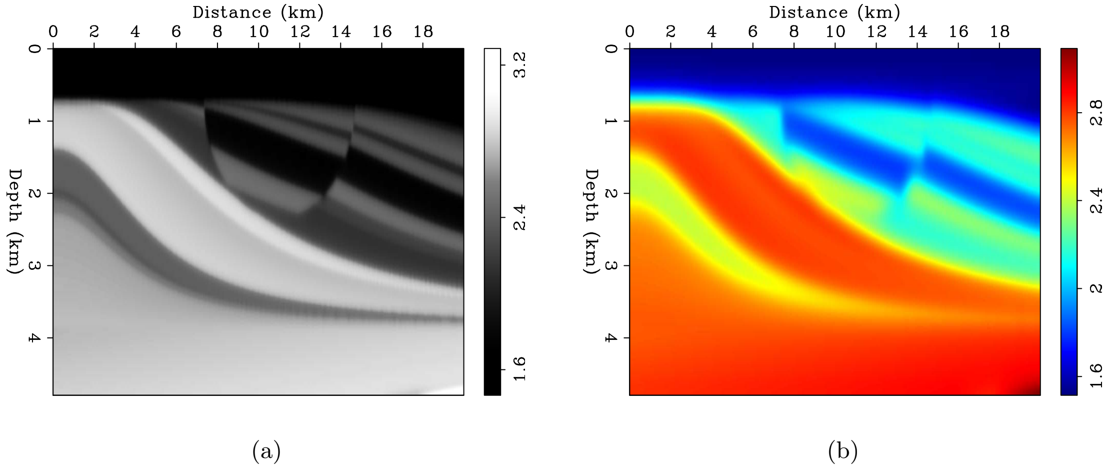


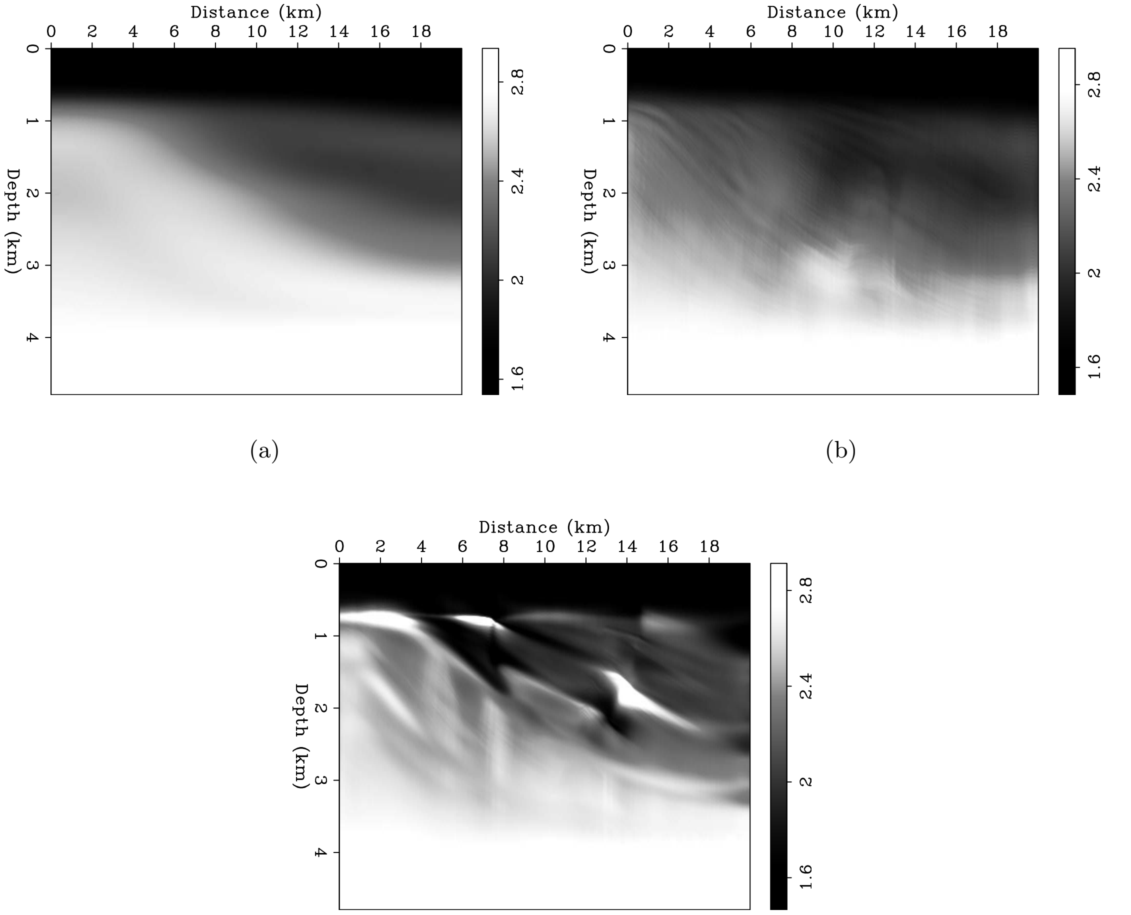
Figure 3: Synthetic sinusoidal model with **(a)** two velocity anomalies and corresponding **(b)** resistivity model. **[ER]** mohammad1/. vel-t,softdata1-0

We use the resistivity map as soft data to constrain the tomography problem with the cross-gradient function. In this case, we can write the cross-gradient function given in equation 2 as a linear operator $\mathbf{G}$ on the slowness field, $\mathbf{s}_0 + \Delta\mathbf{s}$. We can then extend the linearized tomography problem by employing $\mathbf{G}$ as an additional constraint. The objective function, $\mathcal{P}(\Delta\mathbf{s})$, of this extended problem becomes

$$\mathcal{P}(\Delta\mathbf{s}) = ||\Delta\mathbf{t} - \mathbf{T}_L\Delta\mathbf{s}||^2 + \epsilon_1^2 \; ||\mathbf{G}(\mathbf{s}_0 + \Delta\mathbf{s})||^2, \tag{6}$$

where $\epsilon_1$ is a problem-specific weight factor to regularize the tomography problem (Clapp, 2001).

Figure 4 shows the initial velocity and the estimated velocities found by solving the tomography problem both with steering filters and the cross-gradient constraint. The results show that steering filters yield a good result for low frequency features such as smooth lateral velocity anomalies; however, it ignores high-frequency structures of the velocity model. On the other hand, the cross-gradient functions are able to provide better estimates for high-frequency features of the velocity model, such as sharp salt boundaries and faults. Steering filters assume *a priori* knowledge of the model parameters, while the cross-gradient function uses the co-located soft data field to build this information. The combination of these two method may be an optimal tool for addressing the velocity estimation problem in more general subsurface structures.



(a)

(b)

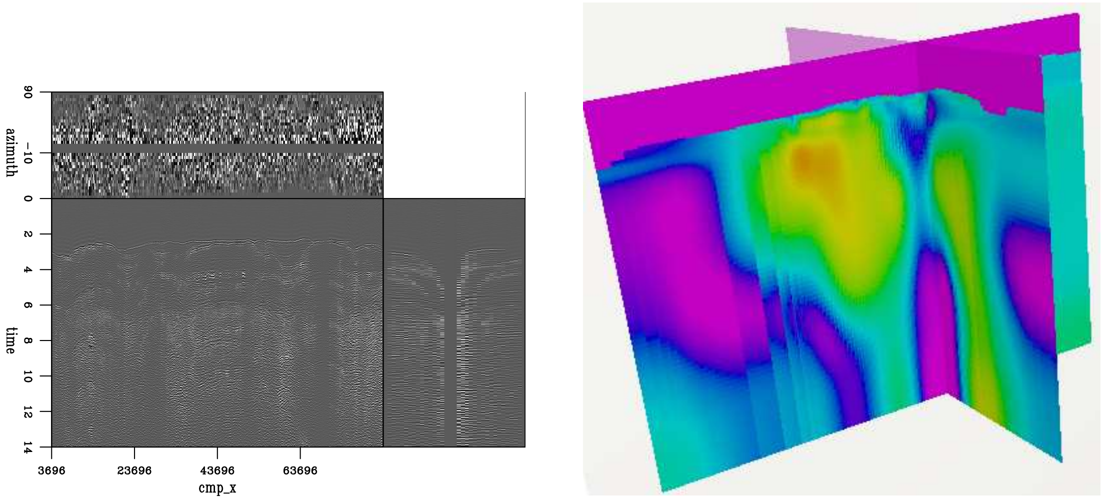
(c)

Figure 4: Velocity estimates by seismic tomography: Initial velocity estimate **(a)** and estimated velocity **(b)** with steering filers and **(c)** with cross-gradient constraint on soft-data. **[CR]** mohammad1/. vel-0,vel-ds0,velx-dsx0

## CONCLUSIONS AND FUTURE WORK

We have reviewed the issues involved in solving a typical seismic tomography problem and how we can address some of them by introduction of additional information. We also discussed our motivations for using the cross-gradient function to incorporate this additional information. The preliminary sensitivity analysis on two synthetic velocity models shows that the cross-gradient functions are a potential tool to integrate different types of geophysical data into the tomography problem. Finally, the comparison between estimated velocities by use of steering filters and cross-gradients functions suggest that we may use these two types of constraints to resolve more general cases of velocity models, including sharp boundaries and smooth anomalies.

This method may lead to improved subsurface interpretations in regions mapped using more than one geophysical method. Figures 5(a) and 5(b) show a CMP gather of seismic data from a marine field dataset and co-located inverted MT resistivity data, respectively. We hope to improve the velocity estimations given by the seismic data itself by including the co-located smooth resistivity map in the tomography problem. Note that the frequency contents of seismic and resistivity data are different, and the resistivity field provides only a low frequency estimation of the subsurface structure. However, we hope to enforce a reasonable geological structure on the output of the seismic tomography problem by using this smooth image as the constraint.



(a) (b)

Figure 5: Field data provided by WesternGeco company: **(a)** a seismic CMP gather from field data and **(b)** The inverted resistivity map from the MT survey. [**NR**] mohammad1/. wg-cmp2,wg-resist2

This method can be extended to seismic tomography constrained by training images, where we can also aim for different realizations of the velocity model by altering the co-located data or training image.

## ACKNOWLEDGMENTS

We thank WesternGeco company and Olav Lindtjorn for providing the field dataset.

## REFERENCES

Aki, K. and P. G. Richards, 2002, Quantitative seismology, second ed.: University Science Books.

Biondi, B., 1990, Velocity analysis using beam stacks: PhD thesis, Stanford University.

Carcione, J. M., B. Ursin, and J. I. Nordskag, 2007, Cross-property relations between electrical conductivity and the seismic velocity of rocks: Geophysics, **72**, E193–E204.

Claerbout, J., 1999, Everything depends on v(x,y,z): SEP-Report, **100**, 1–10.

Clapp, R. G., 2001, Geologically constrained migration velocity analysis: PhD thesis, Stanford University.

Clayton, R. W., 1984, Seismic tomography: Eos, Trans., Am. Geophys Union., **65**.

Etgen, J., 1990, Residual prestack migration and interval velocity estimation: PhD thesis, Stanford University.

Gallardo, L. A. and M. A. Meju, 2004, Joint two-dimensional DC resistivity and seismic travel time inversion with cross-gradients constraints: J. Geophys. Res., **109**.

———, 2007, Joint two-dimensional cross-gradient imaging of magnetotelluric and seismic traveltime data for structural and lithological classication: Geoph. J. Int., **169**, 1261–1272.

Iyer, H. and K. Hirahara, eds., 1993, Seismic tomography: Theory and practice, 1st. ed.: Chapman & Hall.

Kaufman, A. A. and G. V. Keller, eds., 1981, The magnetotelluric sounding method, 1st. ed.: Elsevier Scientific Pub. Co.

Stork, C., 1992, Reflection tomography in the postmigrated domain: Geophysics, **57**, 680–692.

van Trier, J., 1990, Tomographic determination of structural velocities from depth migrated seismic data: PhD thesis, Stanford University.

Versteeg, R. and G. Grau, 1991, The Marmousi experience: Proceedings of the 1990 EAEG Workshop on Practical Aspects of Seismic Data Inversion, 5–16.

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

# Automatic velocity picking by simulated annealing

*Yunyue (Elita) Li and Biondo Biondi*

## ABSTRACT

Manual velocity picking is an inevitable and tedious process in the petroleum industry. An ideal velocity model is both geologically significant and geophysically smooth. Velocity picking can be phrased as a nonlinear optimization problem with multiple contradictory objectives. In this paper, we develop an automatic velocity picking technique based on the Simulated Annealing (SA) Algorithm. Accuracy and smoothness of the velocity model are used as objective functions. To improve the convergence of the algorithm, we include prior knowledge of the velocity model in the initialization and the constraints. The algorithm is adapted for this problem and demostrated using a 2-D field example.

## INTRODUCTION

Measuring errors in velocity is one of the key steps in the processing of seismic data. An accurate velocity model produces accurate depth migration and optimal stack response and can be used directly as the lithology indicator. Residual migration has been shown to be a powerful tool for performing velocity error analysis because of its low computational cost (Rothman et al., 1985; Sava, 2003). After residual migration, we have a cube of residual migration images as a function of $\rho$, the ratio of true velocity to current velocity. Semblance panels corresponding to each value of $\rho$ are computed to evaluate the focusing of the residual migration. However, manual velocity picking is required to obtain the updated velocity model.

An ideal velocity model is both geologically significant and geophysically smooth. It is easy to pick the peaks (maximum values) of the semblance panels for each CMP at each depth. The peak $\rho$ values correspond to the optimal focusing update of the velocity model; however, those values often have large variations both horizontally and vertically. To solve the nonlinear velocity inversion problem, Singh et al. (2008) proposed a customized, multiobjective evolutionary algorithm. The simulated annealing algorithm is also a global optimization method and is capable of coping with the nonlinear relationship between the seismic data and the velocity model.

The simulated annealing algorithm is a Monte Carlo approach for minimizing multivariate functions. The term “simulated annealing” derives from the roughly analogous physical process of heating and then slowly cooling a substance to obtain a strong crystalline structure. In the simulation, a minimum of the cost function corresponds to this ground state of the substance. The simulated annealing process gradually lowers the temperature in stages until the system freezes and no further changes occur.

In this paper, we customize the simulated annealing algorithm to automatically pick the semblance panels and give an optimized velocity model which is both semblance focused and smooth. The algorithm is briefly explained and its objective functions are introduced. We perform experiments on different sets of initialization and constraint parameters, the results and the convergence of which are compared. To test the accuracy of the velocity models, we apply them to the residual migration cube.

## SIMULATED ANNEALING ALGORITHM

The simulated annealing (SA) algorithm is the computational analog of slowly cooling a metal so that it adopts a low-energy, crystalline state. It is a provably convergent optimizer. Geman and Geman (1984) provided a proof that simulated annealing, if the annealing is sufficiently slow, converges to the global optimum. In geophysics, SA has been employed to solve the problems of statics (Rothman, 1985), waveform inversion (Sen and Stoffa, 1991) and ray tracing (Bona et al., 2009). Here we propose the application of the simulated annealing algorithm to automatic velocity picking.

We initialize the system at a high temperature. At this stage the particles are then free to move around by a small pertubation; as the temperature is lowered, however, they are increasingly confined due to the high energy cost of movement. At each temperature T, SA perturbs the system randomly until it reaches equilibrium. The new state of the perturbed system is accepted according to the metropolis algorithm:

$$P \ = \ min(1, \ e^{-\frac{(E(\rho') \ - \ E(\rho))}{T}}), \tag{1}$$

where $\rho$ is the current velocity model, $\rho'$ is the perturbed velocity model, and E is the objective function presented later in the chapter. As shown in the equation, a higher T introduces a higher probability that an uphill perturbation will be accepted, which means that SA can draw samples from the whole population. As T decreases, only perturbations leading to smaller increases in E are accepted, so that only limited exploration is possible as the system settles to the global minimum. The pseudo code of the algorithm is given in Table 1.

The simulated annealing algorithm is popular and has been well-developed for single-objective optimization. Traditionally, multiobjective optimization can be converted into a single-objective optimization by different fix-up approaches such as the weighted-sum or $\epsilon$-constraint method. In our case, we use a composite-objective function:

$$E(\rho) = \omega_{semb} \ \frac{1}{Semb(\rho) + \epsilon} \ + \ \omega_{smooth} \ \nabla\rho, \tag{2}$$

where $Semb(\rho)^{-1}$ is the inverse of semblance and is a measurement of focusing; $\epsilon$ is a arbitrary small number to avoid diverging when $Semb(\rho) \ = \ 0$; and $\nabla\rho$ is the residual after passing the velocity model through the Laplacian operator and is a measurement of smoothness. $\omega_{semb}$ and $\omega_{smooth}$ are the weights for these two measurements, respectively, making $E(\rho)$ is the linear combination of these two objective functions.

There are different ways to compose this single objective function. By using the inverse of semblance and residual of Laplacian we cast the optimization as a minimization problem. This combined objective function is then used as the energy or cost to be minimized in an

Table 1: Algorithm - Simulated Annealing

```
Inputs:
  $\{T_k\}_{k=1}^{K}$   Sequence of temperature values
  $\{L_k\}_{k=1}^{K}$   Sequence of epoch durations
  $\rho$                Initial velocity model
SA loop:
  do itemp = 1, K
    do iepoch = 1, $L_K$
      $\rho'$ : = $perturb(\rho)$
      $\delta$E: = $E(\rho') - E(\rho)$
      rand: = rand(0,1)
      if rand < min(1, exp(-$\delta$E/$T_k$))
        $\rho = \rho'$
      end if
    end do
  end do
```

SA optimizer. For the case we discuss here, the fixed weights are chosen according to the relative importance of the objective functions.

## REPRESENTATION OF THE VELOCITY MODEL

Although we are picking $\rho$ instead of picking actual velocities, discussing the representation of the velocity model is still helpful to better understand the shape and size of model parameters. Kirkpatrick et al. (1983) report that the run time of simulated annealing is related almost linearly to the number of parameters being estimated. Thus, a proper presentation of velocity model which requires fewer parameters will greatly decrease the computational cost of SA. Generally, two widely used classes of velocity models are blocky and smooth velocity models. Blocky models represent the geologically stratified sedimentary rocks, while smooth models have many numerical advantages. Both have been utilized for velocity optimizations using global methods (Jervis et al., 1996; Docherty et al., 1997; Mansanné, 2000).

Here, we are optimizing the residual migration parameter $\rho$ rather than velocity itself. Regarding the initial velocity model, $\rho$ values at different location might be independent with each other, suggesting it is better for us to use a grid to represent the $\rho$ model. At this stage, we optimize the $\rho$ model point by point using grid samplings.

When perturbing the system, we select one sample in the velocity model randomly, and change the velocity at that point to a random velocity value within a reasonable range. The main obstacle for a practical application of such a global optimization method is the computational cost. The larger the parameter space that must be searched and the greater the number of parameters, the more expensive the method tends to be. Thus, the prior knowledge could be both useful for speed and convergence. There are two slots where the prior knowledge can be inserted into the algorithm: initialization and constrains. However, experiments show that incorporating the prior knowledge into initialization is more efficient than into constrains. Thus, we initialize the system by the semblance peaks and randomly perturbing the system by changing the value to any possible $\rho$ value defined by residual migration. Weights for the two objective functions are chosen arbitrarily. The results are presented in the next section.

## 2-D VELOCITY MODEL EXAMPLE

We apply the simulated annealing algorithm to the ELF7D data set. This dataset is complicated not only because of the salt dome, but also the complicated geological structure caused by the rising of the salt.

To optimize the velocity model, we use grid samplings of 63 × 266 m, and the number of grid points is 832. We start from a residual migration semblance cube. The residual migration parameter $\rho$ ranges from 0.925 to 1.0745. The aim of the algorithm is to produce a smooth and accurate velocity model in accordance with the semblance cube. We initialize the system with the $\rho$ values corresponding to semblance peaks, and randomly perturb the system through changing the $\rho$ value to any possible $\rho$ value defined by the residual migration. To create a smooth velocity model, we choose a much larger weight for smoothness than for semblance.

The result of the simulated annealing algorithm is shown in Figure 1. The model generated by 3200 iterations is much smoother than the initial model, but its main structure is still present in the final velocity model.

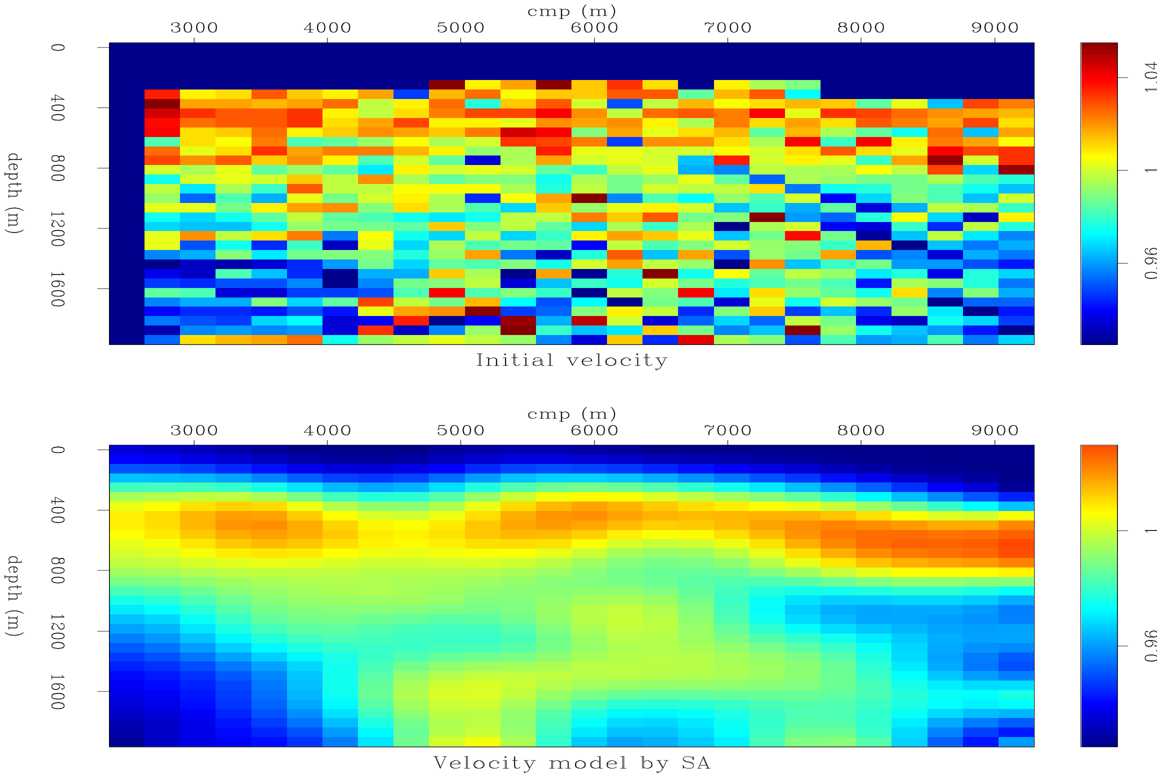
The top panel of Figure 2 shows the composite cost function while cooling. The cost function fluctuates up and down when the temperature is high, but the general trend is always decreasing. When the temperature is lower, the cost function goes down more steadily. As seen in the plot, the uphill perturbations are accepted less and less frequently by the Metropolis algorithm along the cooling path. Generally, the cost decreases much faster during the first iterations. As the annealing proceeds, the cost becomes more stable, until complete stability is reached at crystallization. The bottom two plots in Figure 2 show the cost curve of semblance and smoothness, respectively. Started from the minimum solution for semblance, simulated annealing maintains the increase of semblance within a factor of 2 while significantly enhancing the smoothness.

Figure 3 and Figure 5 show the migration images before and after SA velocity analysis, respectively. Figure 4 and Figure 6 show the corresponding angle domain common image gathers (ADCIGs). It can be seen that the corners of the geological structure are focused and the boundaries of the salt body are connected after the velocity update.

## FUTURE STEPS

### Parallel computing

As Geman and Geman (1984) pointed out, simulated annealing can be implemented in parallel processes. In theory, using N processors would reduce run time by a factor of N. Additionally, parallel computing can also break the large problem down into small problems.



Initial velocity

Velocity model by SA

Figure 1: Initial velocity model (top). Velocity model after 3200 iterations (bottom). The SA velocity is much smoother and still has the structure of the initial velocity model. **[ER]** **elita1/. velocity-semb**

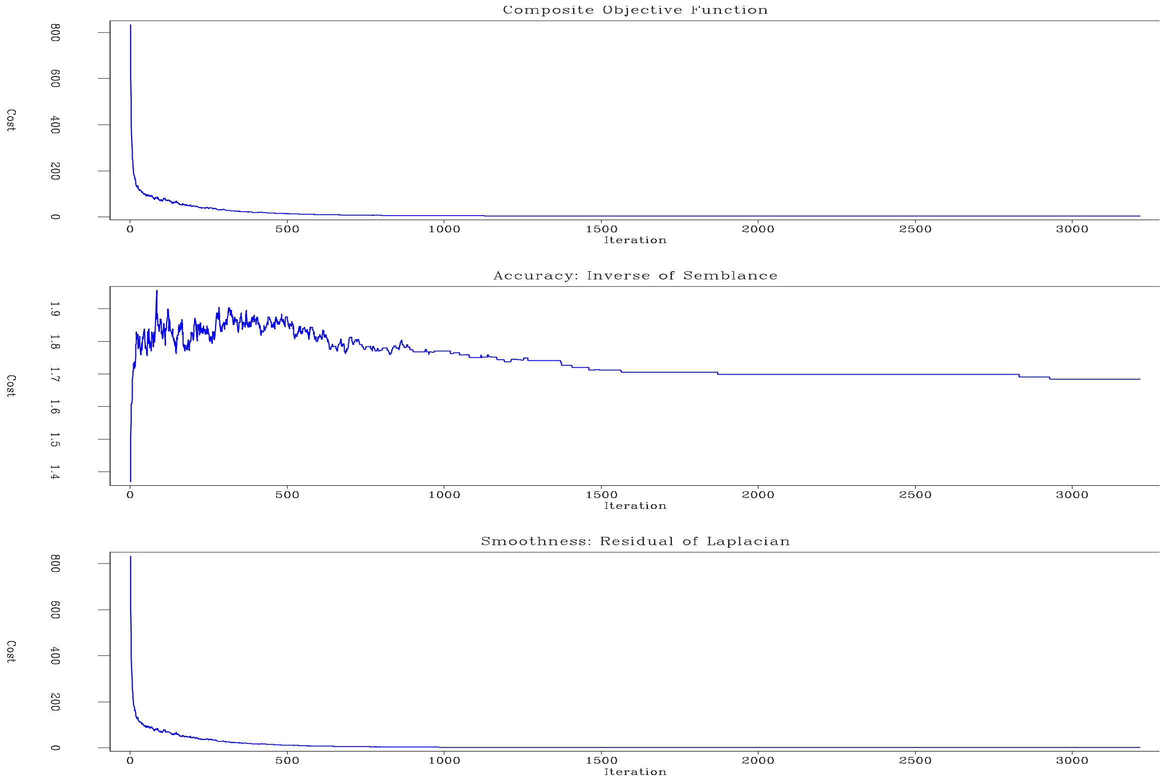


Figure 2: The plot on the top is the composite cost function; the plot in the middle is the cost of semblance; the plot at the bottom is the cost of smoothness. SA succeeds in controlling the increase of semblance when significantly enhancing smoothness. **[ER]** **elita1/. costpart**

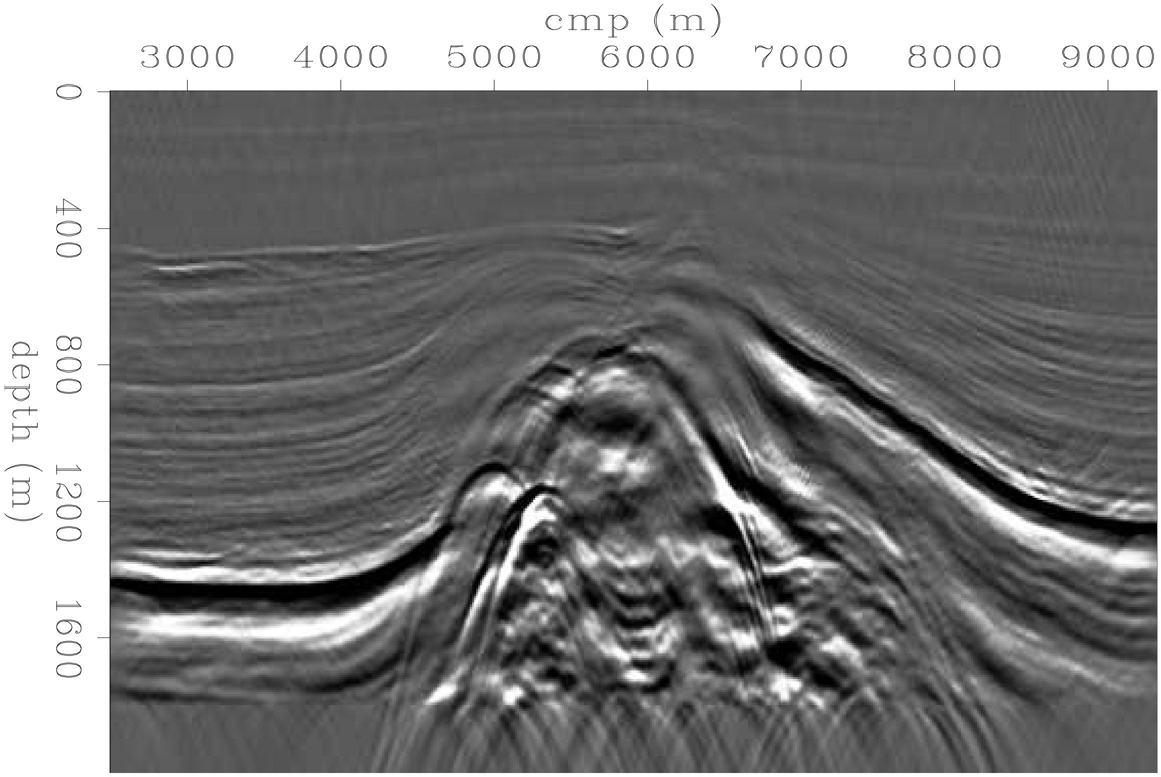


Figure 3: Migration image before performing SA velocity analysis. **[ER]** **elita1/. stack0**

Then the computation time is reduced to the same length of time that the small problems require.

### Pareto optimal front

Automatic velocity picking is a multiobjective optimization problem. In this paper, we use a composite-objective function to find the optimal result. In fact, for multiobjective optimization, there exists a set of solutions known as Pareto optimal: no other solutions are feasible, which would decrease some objective without causing a simultaneous increase in at least one other objective. Several attempts (Pereyra, 2009; Nam and Park, 1999) have been made to explore the Pareto Front of multiobjective optimization. The next step of our project is to progress rapidly to the Pareto-optimal front, and then to finalize the result along the front in accordance with some constraints.

## CONCLUSIONS

Simulated annealing is an effective global optimization method to cope with the nonlinear velocity picking problem. We cast two contradictory objective functions into a single objective function by a weighted-sum criterion. The weights for each function are chosen according to their importance. Experiments show that including prior information into

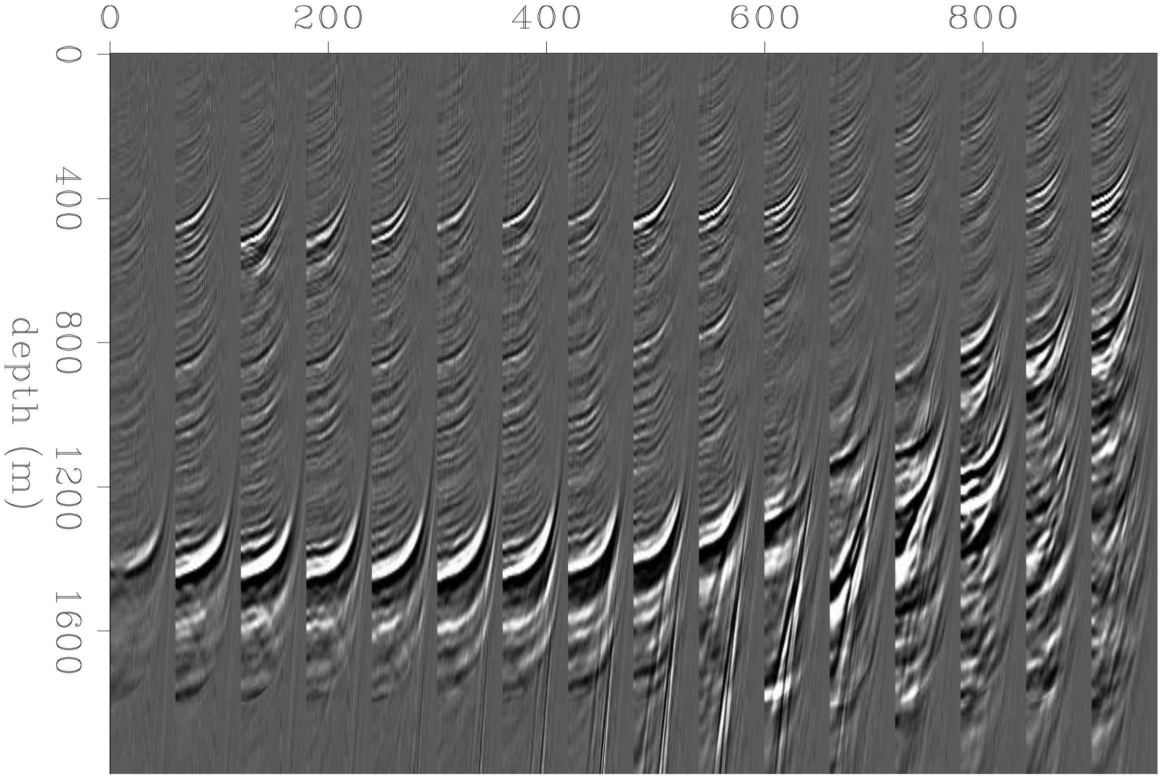


Figure 4: Corresponding angle domain common image gathers of figure 3. **[ER]** **elita1/. cigplane0**

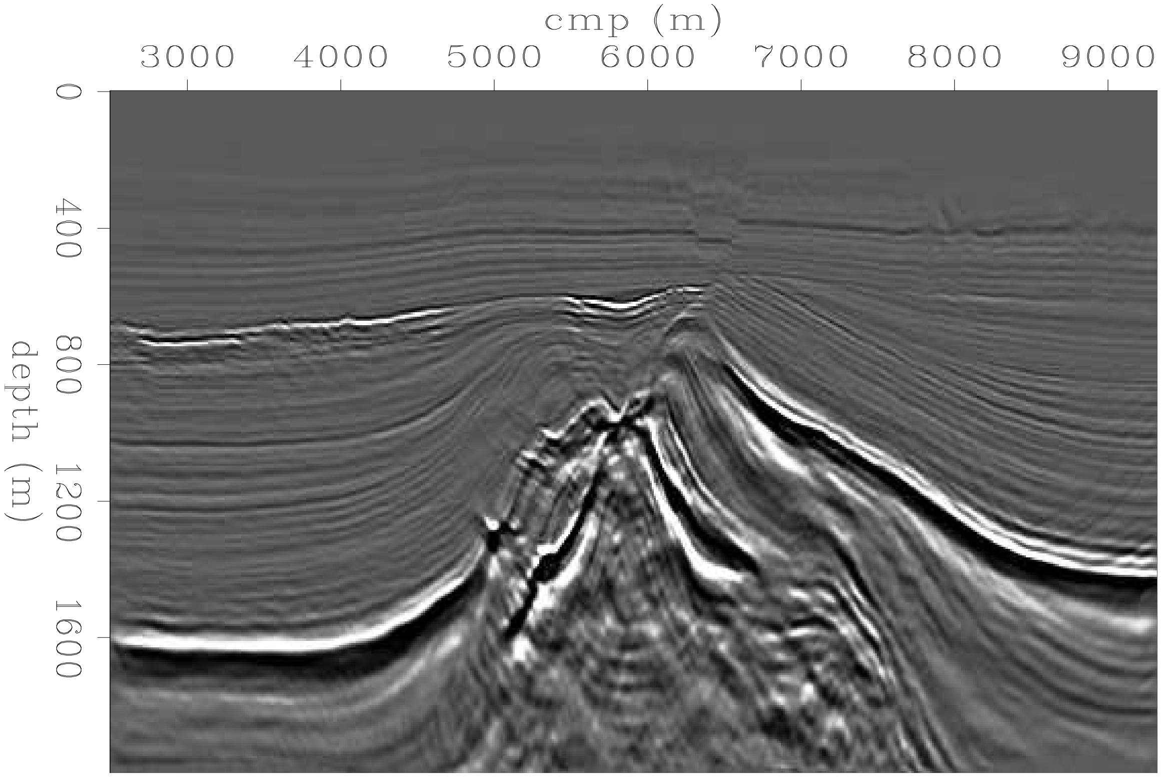


Figure 5: The residual migration image using the SA velocity model. Structures are well focused and the salt boundaries are connected. **[ER]** **elita1/. stack1**

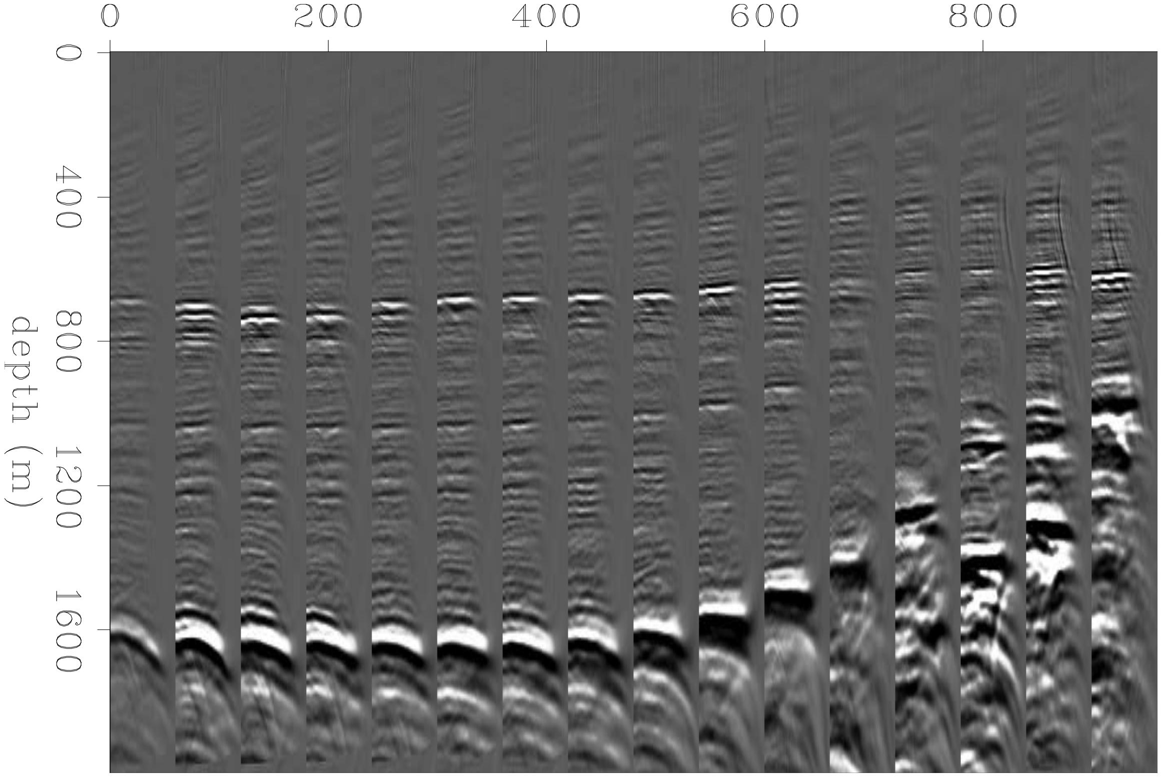


Figure 6: Angle domain common image gathers according to SA velocity. They are not flat in the complex area. The smiling events above water bottom can be corrected by further contrails. **[ER]** **elita1/. cigplane1**

the initialization is important both speed and convergence. The results demostrate the robustness of the algorithm.

## REFERENCES

Bona, A., M. A. Slawinski, and P. Smith, 2009, Ray tracing by simulated annealing: Bending method: Geophysics, **74**, T25–T32.

Docherty, P., R. Silva, S. Singh, Z. Song, and M. Wood, 1997, Migration velocity analysis using a genetic algorithm: Geophysical prospecting, **45**, 865–878.

Geman, S. and D. Geman, 1984, Stochastic relaxation, gibbs distributions, and the bayesian restoration of images: IEEE Transactions on Pattern Analysis and Machine Intelligence, **PAMI-6**.

Jervis, M., M. K. Sen, and P. L. Stoffa, 1996, Prestack migration velocity estimation using nonlinear methods: Geophysics, **61**, 138–150.

Kirkpatrick, S., C. Gelatt, and M. Vecchi, 1983, Optimization by simulated annealing: Science, **220**, 671–680.

Mansanné, F., 2000, Analyzse d'algorithmes d'évolution artificielle appliqués au domaine pétrolier: PhD thesis, Université de Pau.

Nam, D. and C. H. Park, 1999, Pareto-based cost simulated annealing for multiobjective optimization: 127–128.

Pereyra, V., 2009, Fast computation of equispaced pareto manifolds and pareto fronts for multiobjective optimization problems: Mathematics and Computers in Simulation, **75**, 1935–1947.

Rothman, D. H., 1985, Large near-surface anomalies, seismic reflection data, and simulated annealing: PhD thesis, Stanford University.

Rothman, D. H., S. A. Levin, and F. Rocca, 1985, Residual migration: Applications and limitations: Geophysics, **50**, 110–126.

Sava, P., 2003, Prestack residual migration in the frequency domain: Geophysics, **68**, 634–640.

Sen, M. K. and P. L. Stoffa, 1991, Nonlinear one-dimensional seismic waveform inversion using simulated annealing: Geophysics, **56**, 1624–1638.

Singh, V., B. Duquet, M. Leger, and M. Schoenauer, 2008, Automatic wave-equation migration velocity inversion using multiobjective evolutionary algorithms: Geophysics, **73**, VE61–VE73.

# Least-squares migration/inversion of blended data

*Yaxun Tang and Biondo Biondi*

## ABSTRACT

We present a method based on least-squares migration/inversion to directly image data collected from recently developed wide-azimuth acquisition geometries, such as simultaneous shooting and continuous shooting, where two or more shot records are often blended together. We show that by using least-squares migration/inversion, we not only enhance the resolution of the image, but more importantly, we also suppress the crosstalk or acquisition footprint, without any pre-separation of the blended data. We demonstrate the concept and methodology in 2-D and apply the data-space inversion scheme to the Marmousi model, where an optimally reconstructed image, free from crosstalk artifacts, is obtained.

## INTRODUCTION

High-quality seismic images are extremely important for subsalt exploration, but data collected from conventional narrow-azimuth towed streamers (NATS) often produce poor subsalt images due to insufficient azimuth coverage. Recently developed wide-azimuth towed streamers (WATS) (Michell et al., 2006) and multi-azimuth towed streamers (MATS) (Keggin et al., 2006; Howard and Moldoveanu, 2006) acquisition technologies have greatly improved subsalt illumination, and hence better subsalt images are obtained. However, acquiring marine WATS or MATS data is expensive. One main reason is the inefficiency of the conventional way of acquiring data, which requires waiting long enough between shots to prevent interference (Beasley et al., 1998; Beasley, 2008; Berkhout, 2008). As a consequence, the source domain is often poorly sampled to reduce the survey time.

To gain efficiency, simultaneous shooting (Beasley et al., 1998; Beasley, 2008; Hampson et al., 2008) and continuous shooting, or more generally, blended acquisition geometry (Berkhout, 2008), have been proposed to replace the conventional shooting strategy. In the blended acquisition geometry, we try to keep shooting and recording continuously, so that waiting between shots is minimized and a denser source sampling can be obtained. However, this shooting and recording strategy results in two or more shot records blending together and brings processing challenges. A common practice for processing these blended data is to first separate the blended shot gathers into individual ones in the data domain (Spitz et al., 2008; Akerberg et al., 2008), called "deblending" by Berkhout (2008). Then conventional processing flows are applied to these deblended shot gathers. The main issue with this strategy is that it can be extremely difficult to separate the blended gathers when the shot spacing is close and many shots are blended together.

In this paper, we present an alternative method of processing these blended data sets. Instead of deblending the data prior to the imaging step, we propose to directly image them without any pre-separation. The simplest way for direct imaging would be migration; however, migration of blended data generates images contaminated by crosstalk. The crosstalk is due to the introduction of the blending operator (Berkhout, 2008), which makes the corresponding combined Born modeling operator far from unitary; thus its adjoint, also known as migration, gives poor reconstruction of the reflectivity. A possible solution is to go beyond migration by formulating the imaging problem as a least-squares migration/inversion (LSI) problem, which uses the pseudo-inverse of the combined Born modeling operator to reconstruct the reflectivity of the subsurface.

We extend the LSI theory from the conventional acquisition geometry (Nemeth et al., 1999; Clapp, 2005; Valenciano, 2008; Tang, 2008b) to the blended acquisition geometry and develop inversion schemes in both data space and model space. The former minimizes a data-space defined objective function, while the latter minimizes a model-space defined objective function. By comparing the pros and cons of both inversion schemes, we show that the data-space approach is preferred over the model-space approach if the combined Born modeling operator is far from unitary; that is, its normal operator, the Hessian, has many non-negligible off-diagonal elements. Hence an approximate Hessian with a limited number of off-diagonal elements cannot capture the characteristics of the crosstalk, making it less effective in removing the crosstalk in the model space. Big Hessian filters, which sufficiently capture the information of the crosstalk, are too expensive for practical applications. Therefore, the data-space inversion approach, which does not require explicitly computing the Hessian, becomes more attractive. We demonstrate our ideas with simple synthetic examples, and we also test the data-space inversion scheme on the Marmousi model to illustrate how the crosstalk is suppressed through inverting the combined Born modeling operator. Application to 4-D (time-lapse) inversion using blended data sets is discussed in a companion paper by Ayeni et al. (2009).

This paper is organized as follows: we first describe the problem of directly imaging the blended data through migration; then we develop the theory of LSI in both data space and model space for blended data, and compare the pros and cons of the two domains for imaging blended data. Finally, we apply the data-space inversion approach to the Marmousi model to test its performance on a complex model.

## PROBLEMS WITH DIRECT MIGRATION

Within limits of the Born approximation of the acoustic wave equation, the seismic data can be modeled with a linear operator as follows:

$$\mathbf{d} = \mathbf{L}\mathbf{m}, \tag{1}$$

where $\mathbf{d}$ is the modeled data, $\mathbf{L}$ is the forward Born modeling operator, and $\mathbf{m}$ denotes the reflectivity, a perturbed quantity from the background velocity. Equation 1 models the data for the conventional acquisition geometry, i.e., without interference between different shots. For the blended acquisition geometry, however, two or more shot records are often blended together, creating one or more super-areal shot record(s). This blending process can be described by a linear transform as follows:

$$\widetilde{\mathbf{d}} = \mathbf{B}\mathbf{d}, \tag{2}$$

where $\mathbf{B}$ is the so-called blending (Berkhout, 2008) or encoding (Romero et al., 2000; Tang, 2008a) operator, and $\widetilde{\mathbf{d}}$ is the set of super-areal shot records after blending. Substituting equation 1 into equation 2 leads to the modeling equation for the blended acquisition geometry:

$$\widetilde{\mathbf{d}} = \mathbf{BLm} = \widetilde{\mathbf{L}}\mathbf{m}, \tag{3}$$

where $\widetilde{\mathbf{L}} = \mathbf{BL}$ is defined as the combined Born modeling operator.

There are many choices of the blending operator; which one produces the optimal imaging result might be case-dependent and is beyond the scope of this paper. In this paper, we mainly consider two different blending operators: a linear-time-delay blending operator and a random-time-delay blending operator. The first operator seems to be common and easy to implement in practice for acquiring marine data, while the second one is interesting and has been partially adopted in acquiring both land and marine data with simultaneous shooting (Hampson et al., 2008). For example, Figure 1 shows a scattering reflectivity model with a constant velocity of 2000 m/s. Figure 2 and Figure 3 show the snapshots of the corresponding blended source wavefields. For both cases, 41 point sources with an equal spacing of 100 m are blended into one composite source. Figure 4 shows the modeled blended data. Given the complexity of the super-areal shot gathers shown in Figure 4, it might be very difficult or even impossible to deblend them.

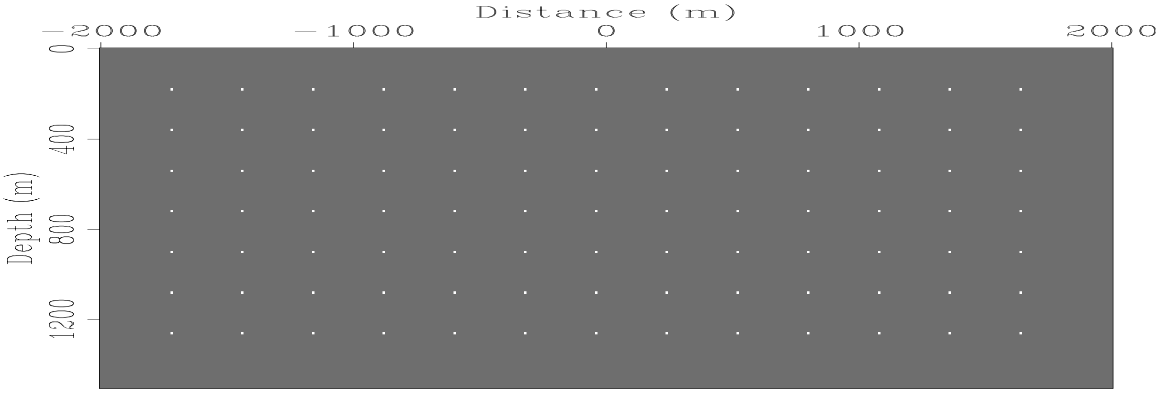


Figure 1: A reflectivity model containing many point scatterers. [**ER**] yaxun1/. pts-refl

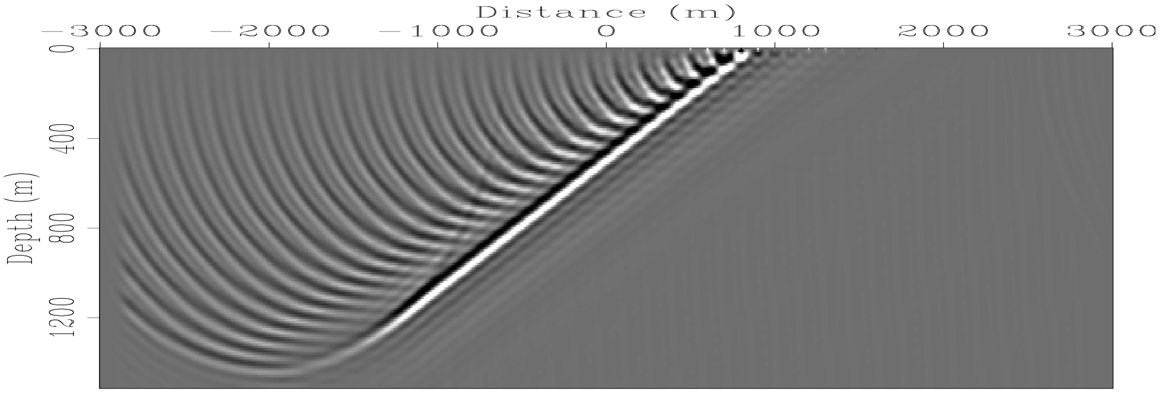


Figure 2: Source wavefield after linear-time-delay blending. [**CR**] yaxun1/. pts-wfds-planes-2

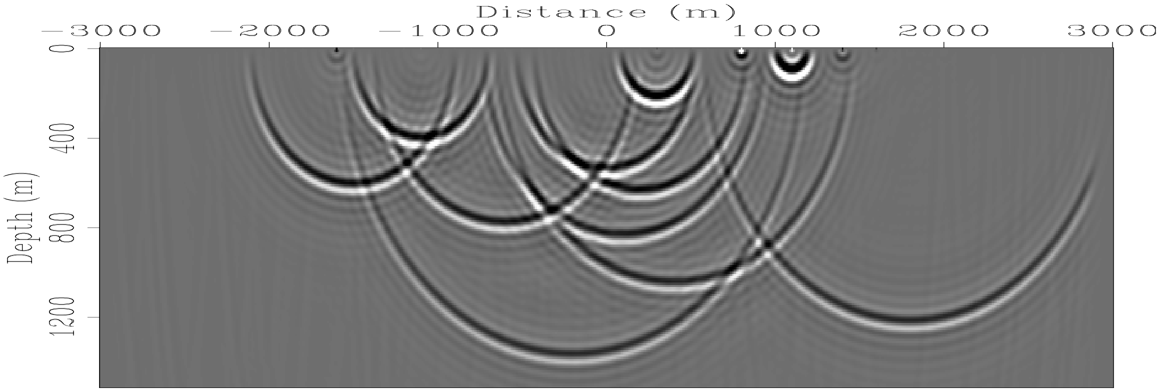
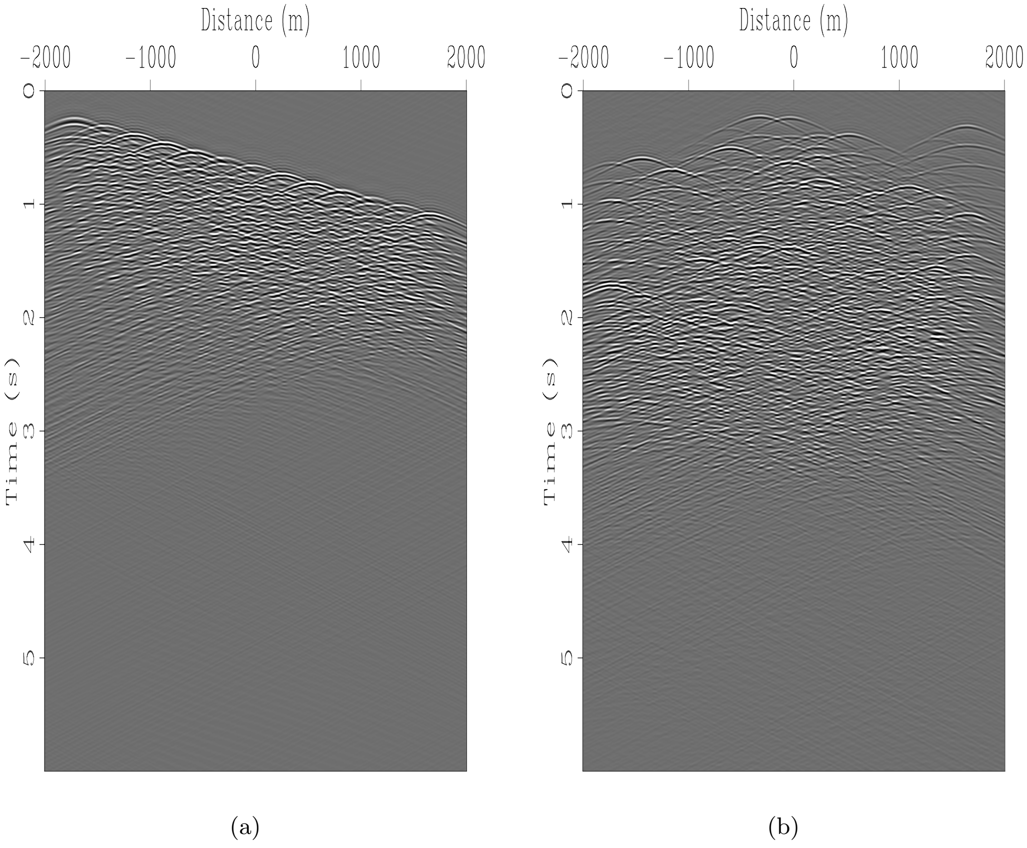


Figure 3: Source wavefield after random-time-delay blending. [**CR**] yaxun1/. pts-wfds-randts



(a)

(b)

Figure 4: Modeled blended shot gather. (a) Linear-time delay blending and (b) random-time-delay blending. [**CR**] yaxun1/. pts-trec-planes-2,pts-trec-randts

We can directly use the adjoint of the combined modeling operator, which is also widely known as the migration operator, to reconstruct the reflectivity as follows:

$$\widetilde{\mathbf{m}}_{\text{mig}} = \widetilde{\mathbf{L}}'\widetilde{\mathbf{d}}_{\text{obs}} = \mathbf{L}'\mathbf{B}'\mathbf{B}\mathbf{d}_{\text{obs}}, \tag{4}$$

where the superscript $'$ denotes the conjugate transpose and the subscript $_{\text{obs}}$ denotes observed data. Contrary to the imaging formula in conventional acquisition geometry, now we have an extra $\mathbf{B}'\mathbf{B}$ in our imaging formula, which has a direct impact on the imaging quality of blended data. If $\mathbf{B}'\mathbf{B}$ is close to unitary, i.e., $\mathbf{B}'\mathbf{B} \approx \mathbf{I}$ with $\mathbf{I}$ being the identity matrix, then direct migration of blended data would produce exactly the same results as migration of conventional data, and the blending process would produce little impact on the final image we obtain. However, in reality, $\mathbf{B}'\mathbf{B}$ is often far from unitary, because $\mathbf{B}$ is usually a short matrix (its number of rows is much smaller than its number of columns); thus its normal operator, $\mathbf{B}'\mathbf{B}$, is rank deficient. In other words, there are many non-negligible off-diagonal elements in $\mathbf{B}'\mathbf{B}$. As a consequence, direct migration using equation 4 would produce crosstalk artifacts. An example is demonstrated in Figure 5 and Figure 6, which illustrate the migrated images for the blended data shown in Figure 4; the images are severely degraded by the crosstalk artifacts. For comparison, Figure 7 shows the crosstalk-free image by migrating the data synthsized with the conventional acquisition geometry (when $\mathbf{B} = \mathbf{I}$).

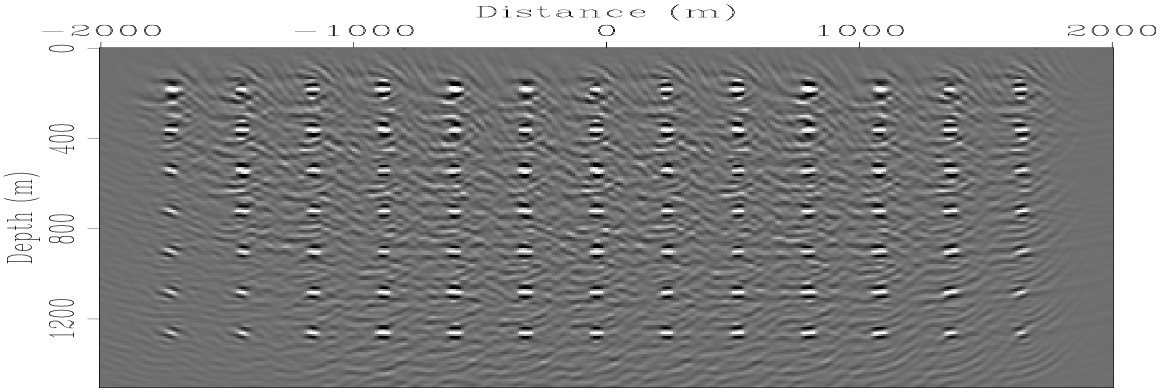


Figure 5: Migration of linear-time-delay blended data. [**CR**] yaxun1/. pts-mig-planes-2

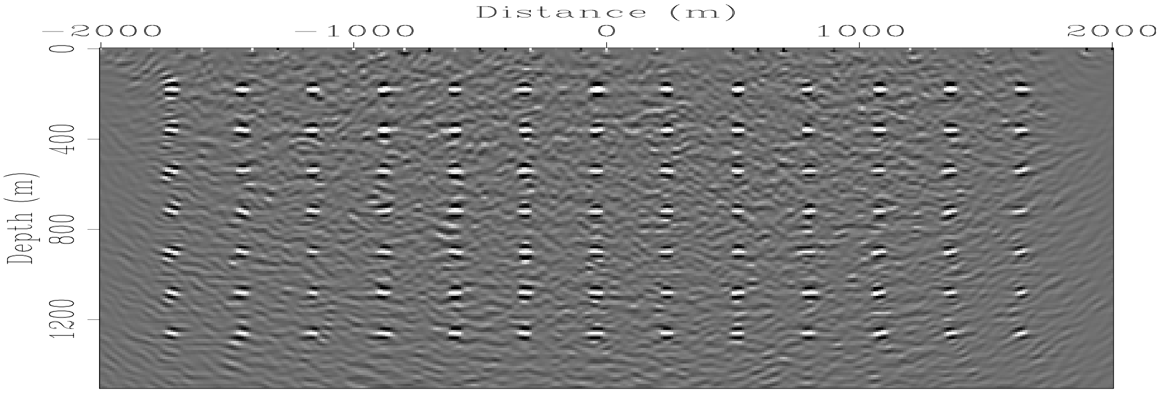


Figure 6: Migration of random-time-delay blended data. [**CR**] yaxun1/. pts-mig-randts

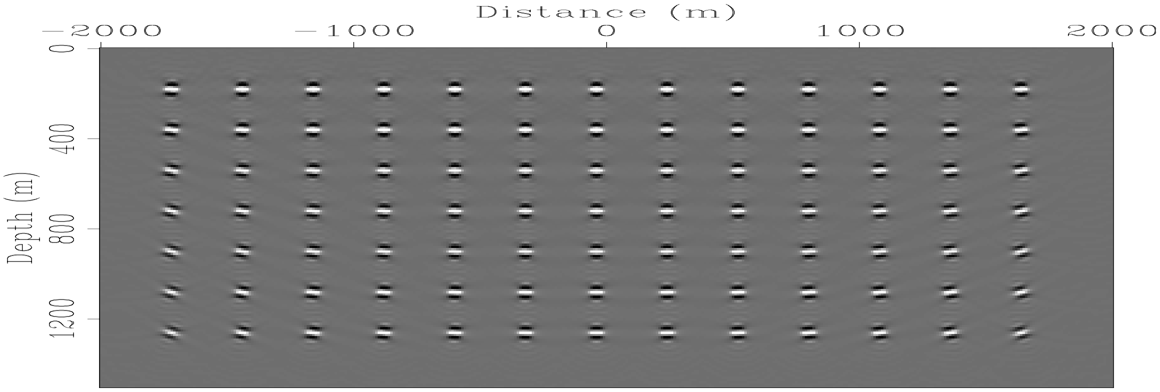


Figure 7: Migration of the data acquired with the conventional acquisition geometry. [**CR**] yaxun1/. pts-mig-shtpro

## DIRECT IMAGING THROUGH INVERSION

A way to reduce the crosstalk is to go beyond migration by formulating the imaging problem as a LSI problem. The motivation behind LSI is that the pseudo-inverse of the combined modeling operator $\widetilde{\mathbf{L}}$ should be able to optimally reconstruct the reflectivity; hence the image would be minimally affected by the crosstalk artifacts. The LSI can be performed either in the model space or in the data space, each of which has its own pros and cons. We will analyze both of them for imaging the blended data.

### LSI in the model space

The least-squares solution of equation 3 can be formally written as follows:

$$\mathbf{m} = \widetilde{\mathbf{H}}^{-1}\widetilde{\mathbf{L}}'\widetilde{\mathbf{d}}_{\text{obs}} = \widetilde{\mathbf{H}}^{-1}\widetilde{\mathbf{m}}_{\text{mig}}, \tag{5}$$

where $\widetilde{\mathbf{H}} = \widetilde{\mathbf{L}}'\widetilde{\mathbf{L}} = \mathbf{L}'\mathbf{B}'\mathbf{B}\mathbf{L}$ is the Hessian for the blended acquisition geometry. However, equation 5 has only symbolic meaning, because the Hessian is often singular and its inverse is not easy to obtain directly. A more practical method is to reconstruct the reflectivity $\mathbf{m}$ through iterative inverse filtering by minimizing a model-space objective function defined as follows:

$$J(\mathbf{m}) = ||\widetilde{\mathbf{H}}\mathbf{m} - \widetilde{\mathbf{m}}_{\text{mig}}||_2^2 + \epsilon||\mathbf{A}\mathbf{m}||_2^2, \tag{6}$$

where $||\cdot||_2$ stands for the $\ell_2$ norm, $\mathbf{A}$ is a regularization operator that imposes prior information that we know about the model $\mathbf{m}$, and $\epsilon$ is a trade-off parameter that controls the strength of regularization.

The advantage of the model-space formulation is that it can be implemented in a target-oriented fashion, which can substantially reduce the size of the problem and hence the computational cost (Valenciano, 2008). However, it requires explicitly computing the Hessian operator, which is expensive without certain approximations. As demonstrated by Valenciano (2008), for a typical conventional acquisition geometry, i.e., when $\mathbf{B} = \mathbf{I}$, the Hessian operator $\mathbf{L}'\mathbf{L}$ is diagonally dominant for areas of good illumination, but for areas of poor illumination, the diagonal energy spreads along its off-diagonals. The spreading is limited and can almost be captured by a limited number of off-diagonal elements. That is why Valenciano (2008) suggests computing a truncated Hessian filter to approximate the exact Hessian for inverse filtering. Doing this makes the cost of the model-space inversion scheme affordable for practical applications. Figure 8(a) shows the local Hessian operator located at $x = 0$ m and $z = 750$ m in the subsurface (a row of the entire Hessian matrix) for the previous scattering model with the conventional acquisition geometry. The origin of this plot denotes the diagonal element of the Hessian, while locations not at the origin denote the off-diagonal elements of the Hessian. As expected, the Hessian is well focused around its diagonal part, and hence can be approximated by a filter with a small size. However, for the blended acquisition geometry, the combined modeling operator $\widetilde{\mathbf{L}}$ becomes far from unitary, and hence the Hessian $\widetilde{\mathbf{H}}$ has non-negligible off-diagonal energy, which can spread over many of the off-diagonal elements. This phenomenon is confirmed by Figure 8(b) and Figure 8(c), which show the local Hessian operators at the same image point for different blended acquisition geometries. It is clear that a filter with a small size could not capture all the important characteristics of the crosstalk in the migrated image; therefore inverse filtering would fail to remove the crosstalk. Figure 9 shows the model-space inversion result with a small Hessian filter (41 × 41 in size) for both blended acquisition geometries. The crosstalk is not removed at all, and the inverted images become even worse.

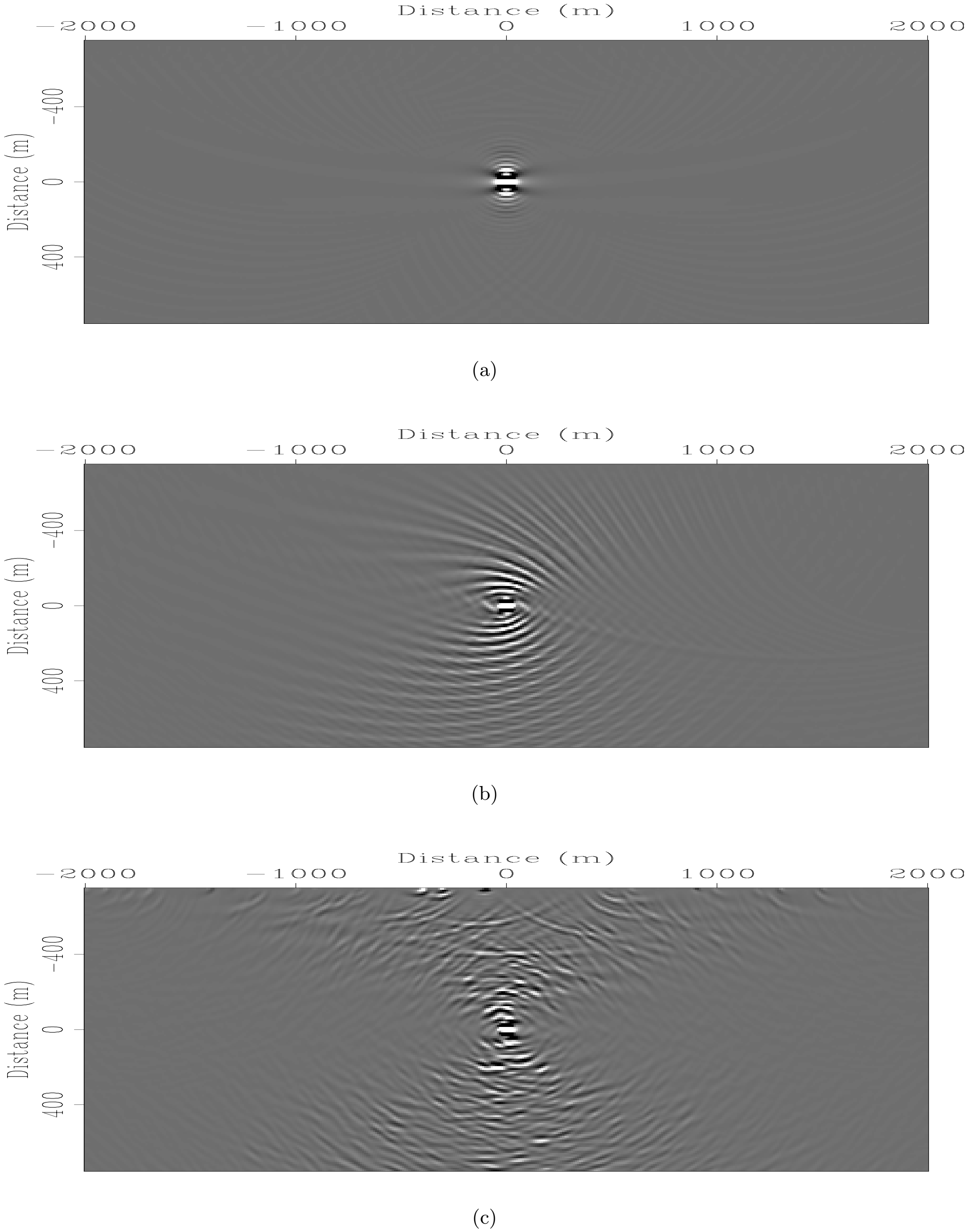
For comparison, Figure 10 shows the inversion result with the full Hessian for a model with only one scatterer in the subsurface (the blending parameters are the same as those for the multiple scattering model). The full Hessian includes all possible off-diagonal elements, so it accurately predicts the crosstalk pattern. The inversion successfully removes the crosstalk. However, the full Hessian is too expensive to compute even though it is target-oriented, and the cost of computing many off-diagonals can quickly outweight the achieved savings of performing the inversion in a target-oriented fashion. Therefore, we seek an inversion approach that does not require explicitly computing the Hessian, so that we do not have to worry about the size of the Hessian filter. This important consideration leads us to the following data-space inversion approach.

## LSI in the data space

The data-space LSI minimizes the following objective function:

$$F(\mathbf{m}) = ||\widetilde{\mathbf{L}}\mathbf{m} - \widetilde{\mathbf{d}}_{\text{obs}}||_2^2 + \epsilon||\mathbf{A}\mathbf{m}||_2^2. \tag{7}$$

The data-space objective function $F(\mathbf{m})$ can be minimized through gradient-based optimization schemes, which iteratively reconstruct the model parameters. The advantage of this data-space formulation is that it does not require explicitly building the Hessian operator; hence all crosstalk information is captured implicitly. However, the data-space formulation lacks flexibility and can not be implemented in a target-oriented fashion. Its cost is another concern, because each iteration costs about the same as two migrations, making it challenging for large scale applications. The cost can be significantly reduced by using proper preconditioners, which may speed up the convergence considerably.

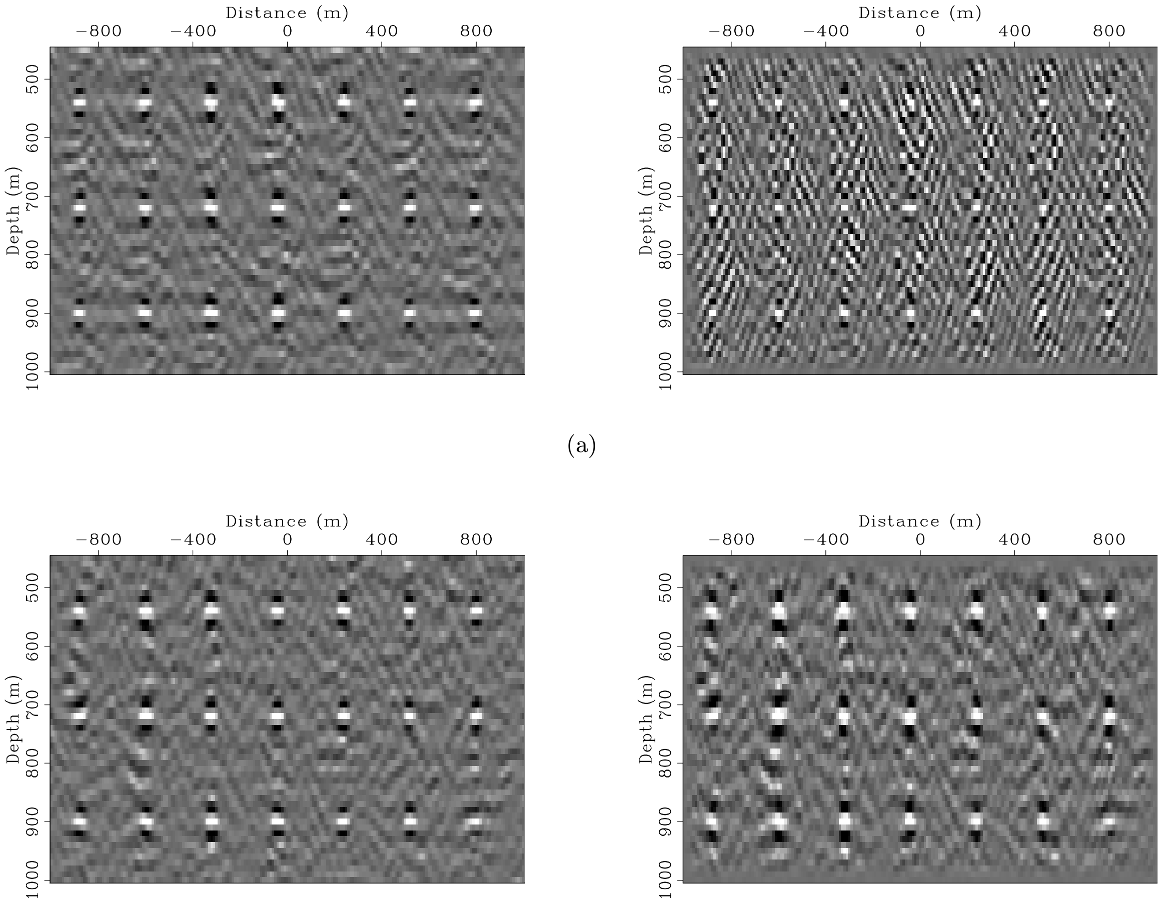


(a)

(b)

(c)

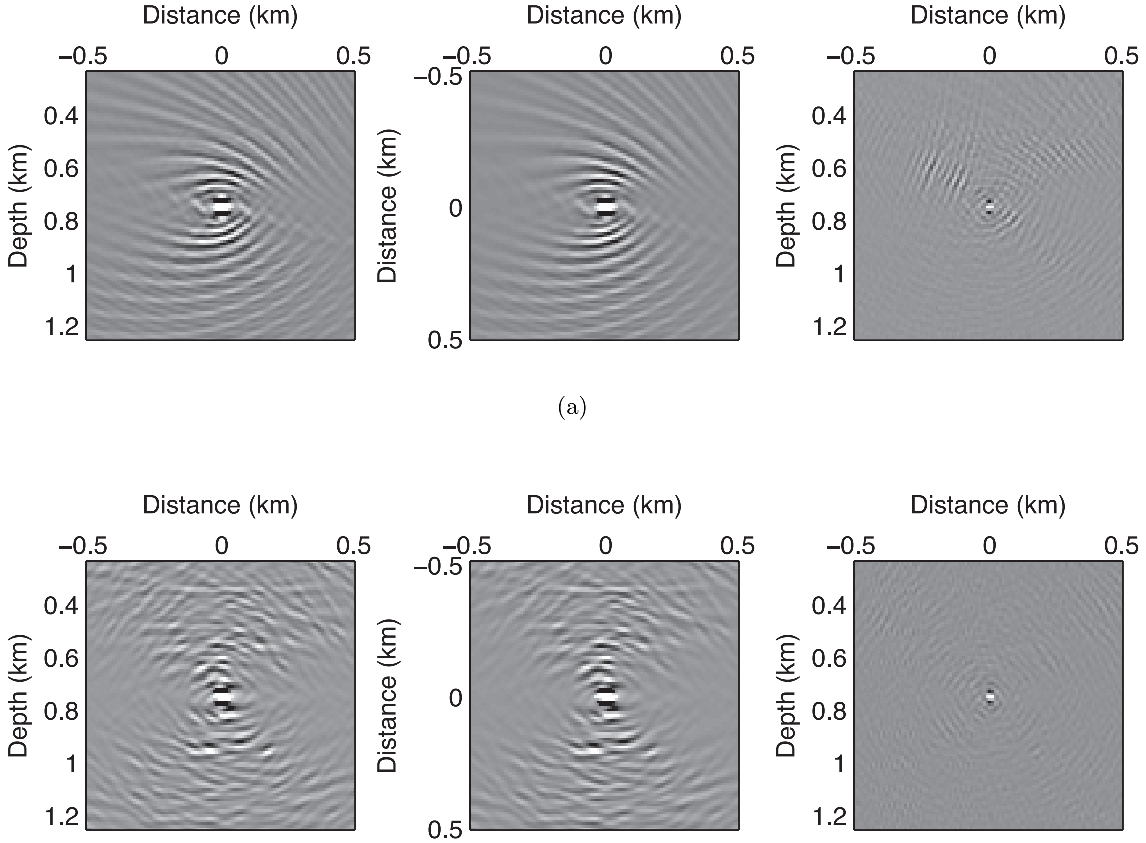
Figure 8: The local Hessian operator located at $x = 0$ m and $z = 750$ m in the subsurface. (a) Conventional acquisition geometry, (b) linear-time-delay blended acquisition geometry, and (c) random-time-delay blended acquisition geometry. [**CR**] yaxun1/. pts-hess-super-shtpro,pts-hess-super-planes-2,pts-hess-super-randts



(a)

(b)

Figure 9: Comparison between migration and model-space LSI with a small Hessian ($41 \times 41$ in size) for the model containing multiple scatters. (a) Linear-time-delay blended acquisition geometry and (b) random-time-delay blended acquisition geometry. In both (a) and (b), the left panel shows the migrated result, while the right panel shows the inverted result. [**CR**] yaxun1/. pts-imag-invt-planes-2,pts-imag-invt-randts



(a)

(b)

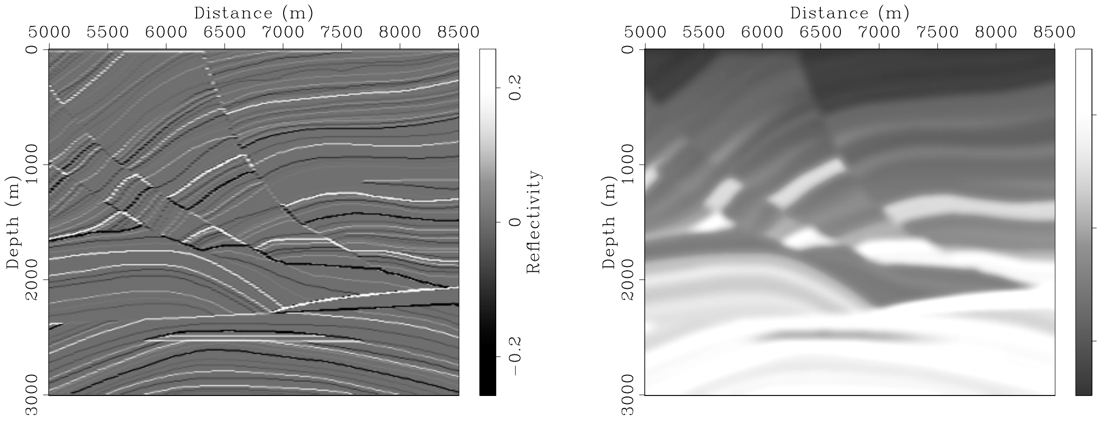
Figure 10: Comparison between migration and model-space LSI with a full Hessian (401 × 151 in size) for a single-scatterer model. (a) Linear-time-delay blended acquisition geometry and (b) random-time-delay blended acquisition geometry. In both (a) and (b), the left panel shows the migrated result, the center panel shows the local Hessian operator (a row of the full Hessian), and the right panel shows the inverted result. [**CR**] yaxun1/. pts-single-decon-planes-2,pts-single-decon-randts

## NUMERICAL EXAMPLES

We test our data-space inversion scheme on the Marmousi model. Figure 11 shows the reflectivity and velocity model used for the Born modeling under the blended acquisition geometry. A total of 51 shots are modeled with a uniform spacing of 100 m, ranging from 4000 m to 9000 m. The receiver spread ranges from 4000 m to 9000 m with a 10 m sampling and is fixed for all shots. As in the previous example of the scattering model, we also simulate linear-time-delay and random-time-delay blended acquisition geometries. Figure 12(a) and Figure 12(b) show the corresponding blended gathers. In both cases, all 51 gathers are blended into one super-areal shot gather.

Figure 13 shows the local Hessian operators for the blended acquisition geometries at different spatial locations in the subsurface. Note that the Hessian operators are far from unitary and contain many non-negligible off-diagonal elements, which contribute to the crosstalk in the migrated images.

The migration and data-space LSI results are shown in Figure 14 and Figure 15. In the inversion results, a horizontal Laplacian operator that imposes horizontal continuities of the reflectivity is used as the regularization operator **A**. For both blended acquisition geometries, migration produces poor images (Figure 14(a) and Figure 15(a)), which are seriously contaminated by crosstalk artifacts. The data-space LSI, on the contrary, successfully removes the crosstalk and we get good reconstruction of the reflectivity in the subsurface (Figure 14(b) and Figure 15(b)).



(a) (b)

Figure 11: The Marmousi model. (a) Reflectivity model; (b) background velocity model. [**ER**] yaxun1/. marm-refl,marm-vmod

## DISCUSSION

Imaging through LSI in blended acquisition geometries is an underdetermined problem; hence there are an infinite number of solutions that fit the observed data equally well. Therefore, regularization is very important for constraining the corresponding solution. In

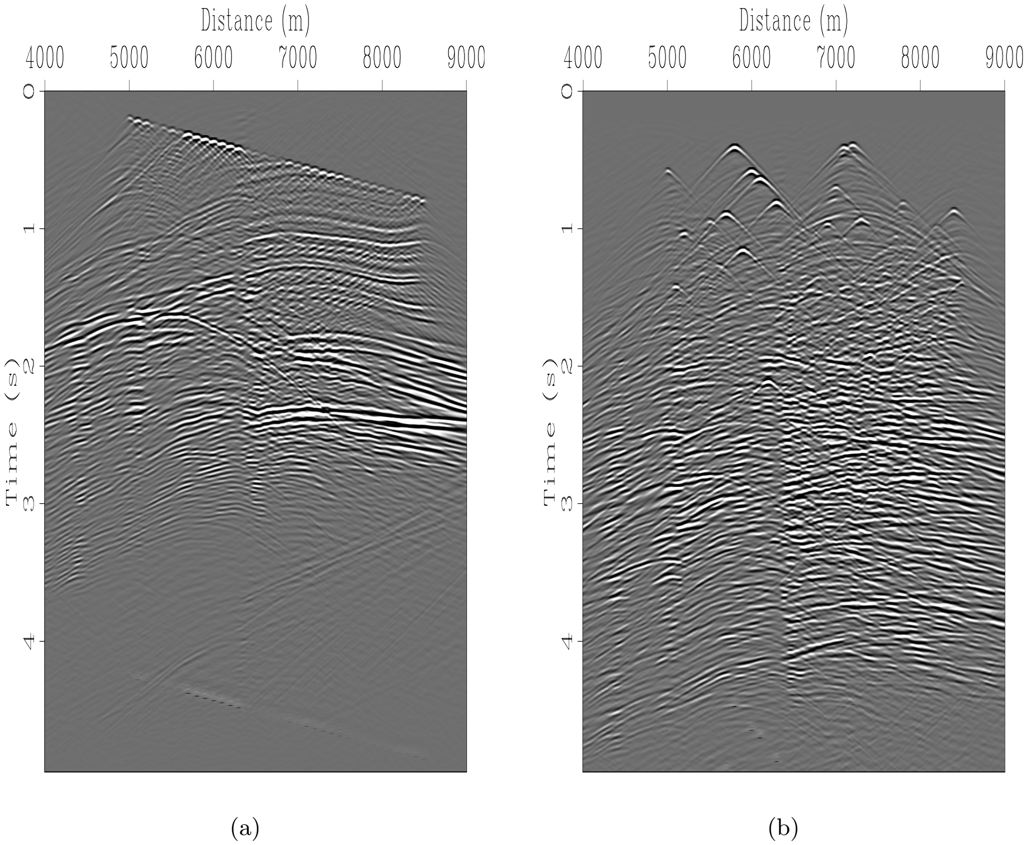
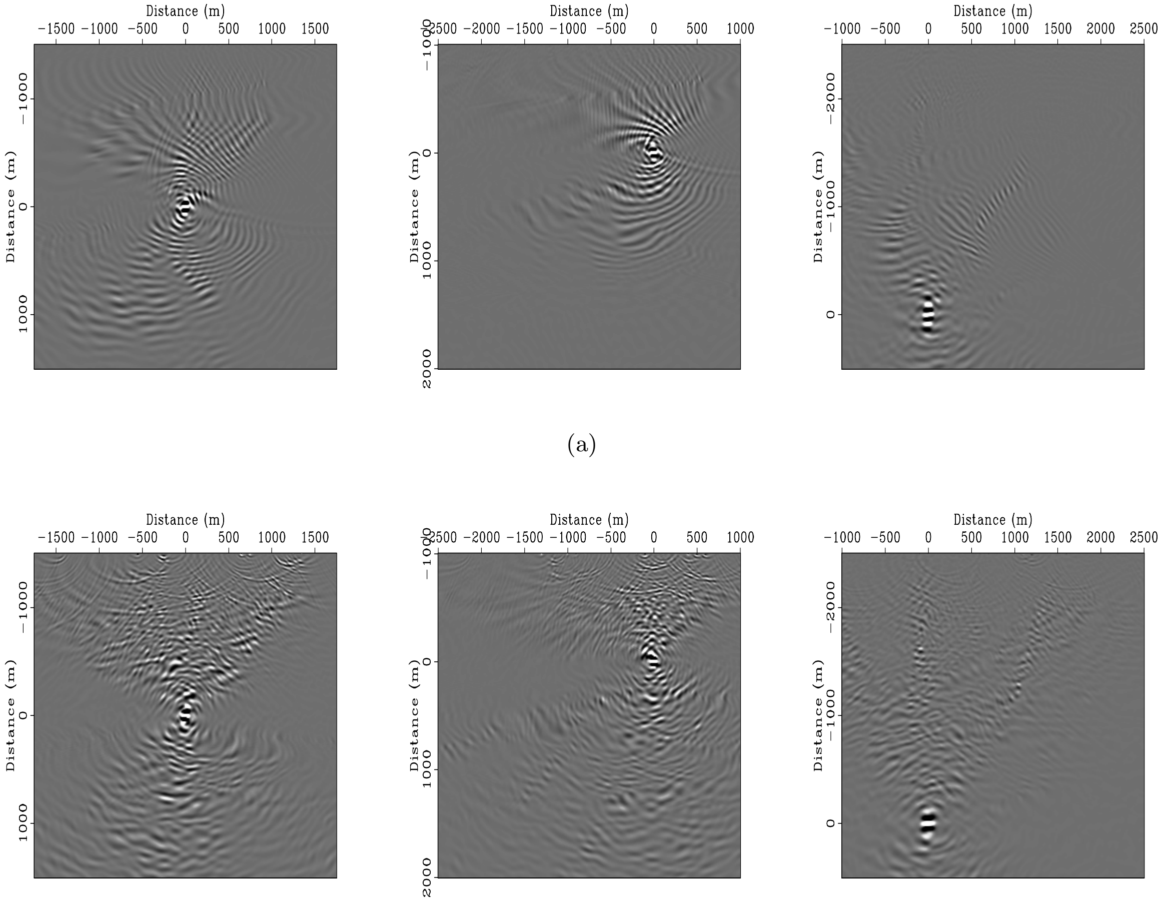


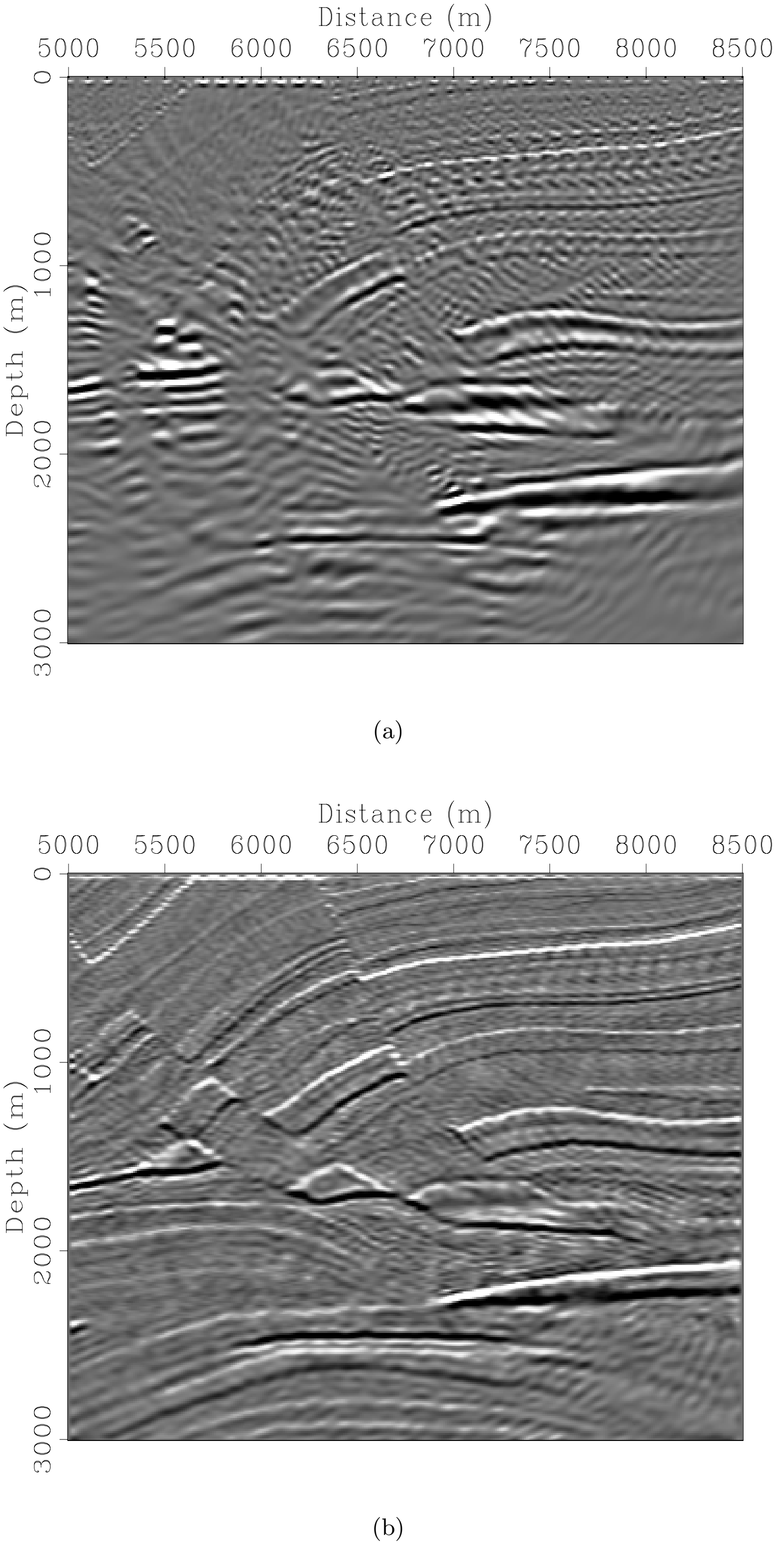
Figure 12: Synthesized blended shot gather for (a) linear-time-delay blended acquisition and (b) random-time-delay blended acquisition. [**CR**] yaxun1/. marm-trec-planes-2,marm-trec-randts



(a)

(b)

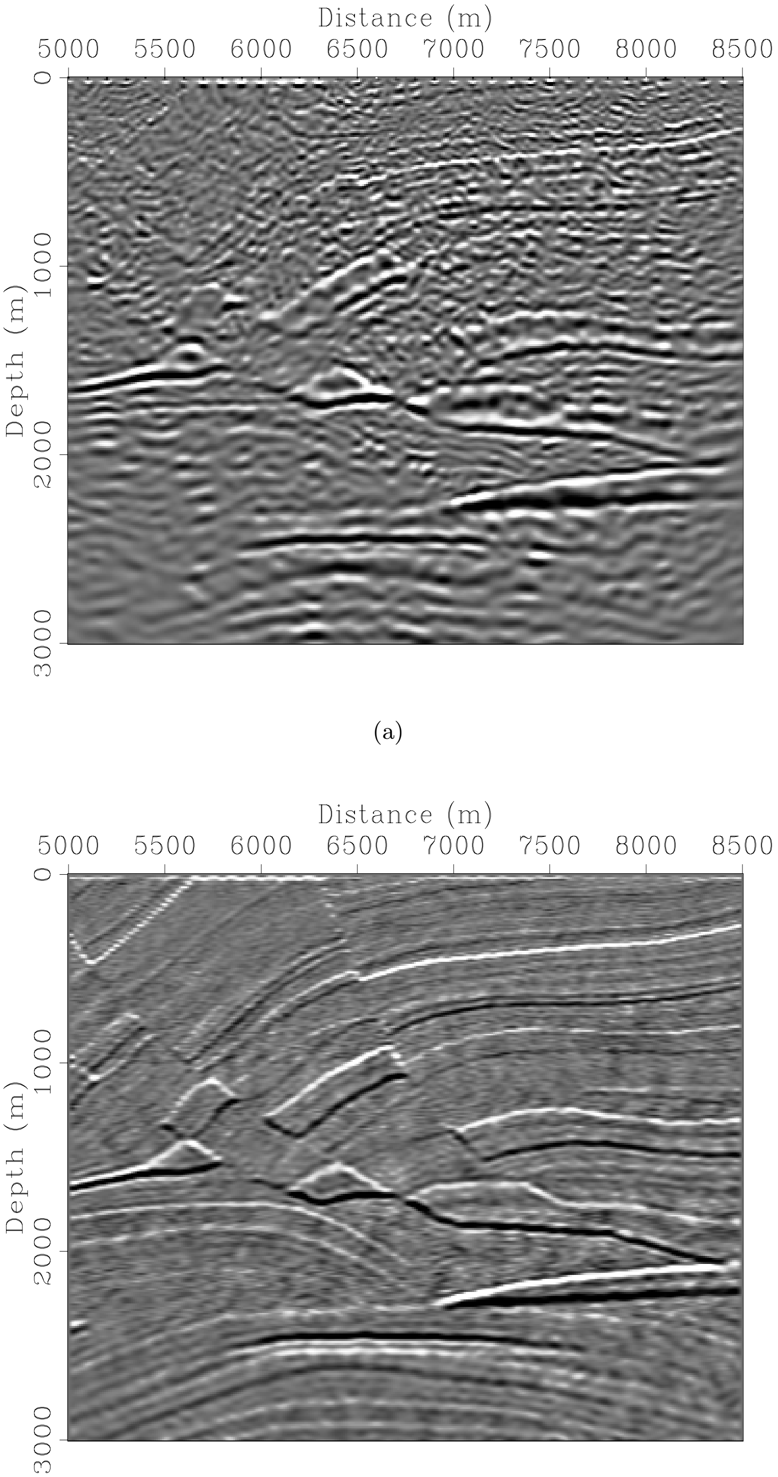
Figure 13: The local Hessian operators (rows of the full Hessian) at different locations for blended acquisition geometries. Panel (a) shows the results for linear-time-delay blending, while panel (b) shows the results for random-time-delay blending. In both (a) and (b), left, center and right panels are the local Hessians located at $x$ = 6750 m, $z$ = 1500 m; $x$ = 7500 m, $z$ = 1000 m; and $x$ = 6000 m, $z$ = 2500 m, respectively. [**CR**] yaxun1/. marm-hess-super-planes-2,marm-hess-super-randts



(a)

(b)

Figure 14: Comparison between (a) migration and (b) data-space LSI for the linear-time-delay blended acquisition geometry. [**CR**] yaxun1/. marm-imag-planes-2,marm-lsm-planes-2



(a)

(b)

Figure 15: Comparison between (a) migration and (b) data-space LSI for the random-time-delay blended acquisition geometry. [**CR**] yaxun1/. marm-imag-randts,marm-lsm-randts

the Marmousi example, a simple regularization operator that imposes horizontal smoothness seems to be working well. However, this choice may not be optimal, because it may wash out dipping reflectors. Regularization operators that better predict the inverse model covariance, for example, by imposing continuities along reflection angles and geological dips (Clapp, 2005), or promoting sparseness in the image domain (Tang, 2009), should be able to reduce the null space and further improve the inversion result. How to incorporate accurate prior information to constrain the inversion remains a research area for further investigation.

## CONCLUSIONS

We present a method based on LSI to directly image the subsurface using blended data sets. This method does not require any pre-separation of the blended shot gathers, and the crosstalk is effectively removed by formulating the imaging problem as a least-squares inverse problem. The inversion examples on the Marmousi model show that LSI can successfully remove the crosstalk that migration generates and optimally reconstruct the reflectivity in the subsurface.

## ACKNOWLEDGMENTS

We thank Guus Berkhout, whose invited talk at Stanford University inspired us to think about the potential opportunities of the blended acquisition geometry.

## REFERENCES

Akerberg, P., G. Hampson, J. Rickett, H. Martin, and J. Cole, 2008, Simultaneous source separation by sparse radon transform: SEG Technical Program Expanded Abstracts, **27**, 2801–2805.

Ayeni, G., Y. Tang, and B. Biondi, 2009, Joint preconditioned inversion of simultaneous source time-lapse seismic data sets: SEG Technical Program Expanded Abstracts, **28**, submitted.

Beasley, C. J., 2008, A new look at marine simultaneous sources: The Leading Edge, **27**, 914–917.

Beasley, C. J., R. E. Chambers, and Z. Jiang, 1998, A new look at simultaneous sources: SEG Technical Program Expanded Abstracts, **17**, 133–135.

Berkhout, A. J. G., 2008, Changing the mindset in seismic data acquisition: The Leading Edge, **27**, 924–938.

Clapp, M. L., 2005, Imaging Under Salt: Illumination Compensation by Regularized Inversion: PhD thesis, Stanford University.

Hampson, G., J. Stefani, and F. Herkenhoff, 2008, Acquisition using simultaneous sources: The Leading Edge, **27**, 918–923.

Howard, M. S. and N. Moldoveanu, 2006, Marine survey design for rich-azimuth seismic using surface streamers: SEG Technical Program Expanded Abstracts, **25**, 2915–2919.

Keggin, J., T. Manning, W. Rietveld, C. Page, E. Fromyr, and R. van Borselen, 2006, Key aspects of multi-azimuth acquisition and processing: SEG Technical Program Expanded Abstracts, **25**, 2886–2890.

Michell, S., E. Shoshitaishvili, D. Chergotis, J. Sharp, and J. Etgen, 2006, Wide azimuth streamer imaging of mad dog; have we solved the subsalt imaging problem?: SEG Technical Program Expanded Abstracts, **25**, 2905–2909.

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

Romero, L. A., D. C. Ghiglia, C. C. Ober, and S. A. Morton, 2000, Phase encoding of shot records in prestack migration: Geophysics, **65**, 426–436.

Spitz, S., G. Hampson, and A. Pica, 2008, Simultaneous source separation: A prediction-subtraction approach: SEG Technical Program Expanded Abstracts, **27**, 2811–2815.

Tang, Y., 2008a, Modeling, migration and inversion in the generalized source and receiver domain: **SEP-136**, 97–112.

———, 2008b, Wave-equation Hessian by phase encoding: SEG Technical Program Expanded Abstracts, **27**, 2201–2205.

———, 2009, Target-oriented least-squares migration/inversion with sparseness constraints: **SEP-138**.

Valenciano, A., 2008, Imaging by Wave-equation Inversion: PhD thesis, Stanford University.

# Joint inversion of simultaneous source time-lapse seismic data sets

*Gboyega Ayeni, Yaxun Tang, and Biondo Biondi*

## ABSTRACT

We present a joint inversion method, based on iterative least-squares migration, for imaging simultaneous source time-lapse seismic data sets. Non-repeatable shot and receiver positions introduce undesirable artifacts into time-lapse seismic images. We conjecture that when data sets are acquired with several simultaneously shooting sources, additional artifacts will result from *relative shot-timing non-repeatability.* These artifacts can be attenuated by joint inversion of such data sets without any need for initial separation. Preconditioning with non-stationary dip filters and with temporal smoothness constraints ensures stability and geologically consistent time-lapse images. Results from a modified Marmousi 2D model show that the proposed method yields reliable time-lapse images.

## INTRODUCTION

Time-lapse (4D) seismic is an established technology for monitoring hydrocarbon reservoirs. It is central to most field development and management plans, and many successful applications have been published (Whitcombe et al., 2004; Zou et al., 2006). However, in many time-lapse seismic applications, inaccuracies in replication of acquisition geometries for different surveys (*non-repeatability*) is a recurring problem. Although modern acquisition techniques can improve repeatability of shot-receiver positions, field conditions usually prevent perfect repeatability.

Recently, several authors have suggested acquisition with multiple simultaneously shooting seismic sources. Although, not a new technology (Womack et al., 1990; Beasley et al., 1998), modern acquisition and imaging techniques now make simultaneous source (or blended) acquisition both appealing and practical. Some advantages of this acquisition method include:

- Improved shot-sampling: *reduces shot-interpolation requirements in conventional narrow-azimuth data.*
- Lower acquisition cost: *enables acquisition of several azimuths in 3D wide-azimuth data sets at lower cost.*
- Longer offsets and full-azimuth: *enables better imaging and improved AVO information.*
- Shorter acquisition time-window: *makes acquisition practical where operational, climatic, political or other uncontrollable factors could have prevented it.*

Different processing schemes have been proposed for simultaneous source data sets. Most of these schemes rely on separation of the data sets into different shot components before standard processing (Hampson et al., 2008; Spitz et al., 2008). Processing schemes that require no separation have also been suggested (Berkhout et al., 2008; Tang and Biondi, 2009). However, there has been little discussion on the implications of this acquisition technique for time-lapse seismic.

We introduce the term *relative shot-timing non-repeatability* to describe a potential source of artifacts in simultaneous source time-lapse seismic data sets. Because current simultaneous source acquisition designs generally rely on randomized shot-timings, it will be difficult to accurately reproduce the relative shot-receiver positions and at the same time maintain the relative shot-timing for different surveys. Shot-receiver non-repeatability, together with the predicted relative shot-timing non-repeatability, will lead to strong degradation of time-lapse seismic images. Because of the complexity introduced by non-repeatability of both shot-receiver positions and relative shot-timing, *conventional* cross-equalization methods for time-lapse seismic data sets will fail. Therefore, we explore least-squares inversion methods of such data sets.

Iterative *data-space* linear least-squares migration/inversion can improve structural and amplitude information in seismic images (Nemeth et al., 1999; Kühl and Sacchi, 2003; Plessix and Mulder, 2004; Clapp, 2005). An extension of *image-space* least-squares inversion (Valenciano, 2008) to time-lapse imaging has been shown to improve time-lapse seismic images (Ayeni and Biondi, 2008). In this paper, we propose a *data-space* joint inversion method for imaging simultaneous source time-lapse seismic data sets. The proposed method combines the cost-saving advantages of both simultaneous source acquisition and phase encoded migration (Romero et al., 2000). We further demonstrate that preconditioning with non-stationary dip filters and temporal smoothness constraints further improves the time-lapse seismic images.

We assume a known, slowly changing background baseline velocity. Because a close approximation of the background velocity is essential, we propose baseline data acquisition with separate or few simultaneous sources and monitor data acquisition with several simultaneous sources. We assume careful processing of the baseline data such that the data can be used for velocity estimation and the image can be used for dip estimation or interpretation. Furthermore, we assume that the shot-receiver positions and relative shot-timing are known for all surveys. Integration of background velocity and geomechanical changes into the joint inversion formulation is ongoing and will be discussed elsewhere.

In this paper, we first discuss Born modeling of phase-encoded data as an approximation of simultaneous source acquisition. Then, using a phase-encoded modeling/migration formulation, we discuss joint linear least-squares inversion of multiple simultaneous source seismic data sets. We also summarize a spatio-temporal preconditioning scheme based on spatial non-stationary dip-filters and temporal leaky integration. Finally, using a modified version of the 2D Marmousi model (Versteeg, 1994), we show that solving the preconditioned joint inversion problem yields optimal time-lapse seismic images.

## LINEAR PHASE-ENCODED BORN MODELING

Within limits of the Born approximation of the acoustic wave equation, the seismic data $\mathbf{d}$ recorded by a receiver at $\mathbf{x_r}$ due to a shot at $\mathbf{x_s}$ is given by

$$d(\mathbf{x_s},\mathbf{x_r},\omega) = \omega^2 \sum_{\mathbf{x}} f_{\mathbf{s}}(\omega) G(\mathbf{x_s},\mathbf{x},\omega) G(\mathbf{x},\mathbf{x_r},\omega) m(\mathbf{x}), \tag{1}$$

where $\omega$ is frequency, $m(x)$ is the *reflectivity* at image points $\mathbf{x}$, $f_{\mathbf{s}}(\omega)$ is the source waveform, and $G(\mathbf{x_s},\mathbf{x},\omega)$ and $G(\mathbf{x},\mathbf{x_r},\omega)$ are respectively the Green's functions from shot $\mathbf{x_s}$ to $\mathbf{x}$ and from $\mathbf{x}$ to $\mathbf{x_r}$.

By considering randomized simultaneous source data as a special case of linear phase-encoded shot gathers, equation (1) is modified to include a concatenation of phase-shifted shots, from $\mathbf{s}=\mathbf{q}$ to $\mathbf{s}=\mathbf{p}$:

$$d(\mathbf{x_{s_{pq}}},\mathbf{x_r},\omega) = \sum_{\mathbf{s}=\mathbf{p}}^{\mathbf{q}} a(\gamma_{\mathbf{s}}) \omega^2 \sum_{\mathbf{x}} f_{\mathbf{s}}(\omega) G(\mathbf{x_s},\mathbf{x},\omega) G(\mathbf{x},\mathbf{x_r},\omega) m(\mathbf{x}), \tag{2}$$

where $\mathbf{x_{s_{pq}}}$ defines the positions of the encoded sources, and

$$a(\gamma_{\mathbf{s}}) = e^{i\gamma_{\mathbf{s}}} = e^{i\omega t_{\mathbf{s}}}, \tag{3}$$

and $\gamma_{\mathbf{s}}$, the linear time-delay function, depends on the delay time $t_{\mathbf{s}}$ at shot $\mathbf{s}$.

Relative shot-timing non-repeatability arises due to the uncertainty (Folland and Sitaram, 1997) associated with correct positioning of shots and receivers while maintaining the correct time delays $t_{\mathbf{s}}$ between shots. This is particularly true for the blended acquisition geometry (Berkhout, 2008), where several (20 or more) shots are encoded into a single record.

## LINEAR LEAST-SQUARES MIGRATION/INVERSION

We re-write the linear modeling operation in equation (1) in matrix-vector form as follows:

$$\mathbf{d} = \mathbf{Lm}, \tag{4}$$

where $\mathbf{L}$ is the modeling operator and $\mathbf{m}$ is the earth reflectivity. The encoding (or blending) operation in equation (2) is then defined as:

$$\widetilde{\mathbf{d}} = \mathbf{BLm} = \widetilde{\mathbf{L}}\mathbf{m}, \tag{5}$$

where $\widetilde{\mathbf{d}}$ is the encoded data, $\mathbf{B}$ is the encoding (or blending) operator, and $\widetilde{\mathbf{L}}$ is the combined modeling and encoding operator.

Given two surveys (baseline and monitor), acquired over an evolving earth model at times $\mathbf{t=0}$ and $\mathbf{t=1}$ respectively, we can write

$$\begin{aligned} \widetilde{\mathbf{d}}_0 &= \widetilde{\mathbf{L}}_0\mathbf{m}_0, \\ \widetilde{\mathbf{d}}_1 &= \widetilde{\mathbf{L}}_1\mathbf{m}_1, \end{aligned} \tag{6}$$

where $\mathbf{m}_0$ and $\mathbf{m}_1$ are the baseline and monitor reflectivities, and $\widetilde{\mathbf{d}}_0$ and $\widetilde{\mathbf{d}}_1$ are the encoded seismic data sets. Note that the modeling operators $\widetilde{\mathbf{L}}_0$ and $\widetilde{\mathbf{L}}_1$ in equation (6) can define both different acquisition geometries and different relative shot-timings.

By applying the adjoint operators to the data sets, we obtain the migrated images:

$$\begin{aligned}\acute{\mathbf{m}}_0 &= \widetilde{\mathbf{L}}_0^*\widetilde{\mathbf{d}}_0,\\ \acute{\mathbf{m}}_1 &= \widetilde{\mathbf{L}}_1^*\widetilde{\mathbf{d}}_1,\end{aligned} \tag{7}$$

where $\acute{\mathbf{m}}_0$ and $\acute{\mathbf{m}}_1$ are the migrated baseline and monitor images respectively, and the symbol $^*$ denotes the conjugate transpose of the modeling operators. The *raw* time-lapse seismic image $\Delta\tilde{\mathbf{m}}$ is the difference between the migrated images:

$$\Delta\acute{\mathbf{m}} = \acute{\mathbf{m}}_1 - \acute{\mathbf{m}}_0. \tag{8}$$

Because of differences in relative shot-timings, cross-term artifacts (Romero et al., 2000; Tang and Biondi, 2009) will be different for each migrated data set. Conventional equalization methods (Rickett and Lumley, 2001; Calvert, 2005) will be inadequate to remove these artifacts.

The quadratic cost functions for equation (6) are

$$\begin{aligned}S(\mathbf{m_0}) &= \|\widetilde{\mathbf{L}}_0\mathbf{m}_0 - \widetilde{\mathbf{d}}_0\|^2,\\ S(\mathbf{m_1}) &= \|\widetilde{\mathbf{L}}_1\mathbf{m}_1 - \widetilde{\mathbf{d}}_1\|^2,\end{aligned} \tag{9}$$

which when minimized gives the inverted baseline $\hat{\mathbf{m}}_0$ and monitor $\hat{\mathbf{m}}_1$ images:

$$\begin{aligned}\hat{\mathbf{m}}_0 &= (\widetilde{\mathbf{L}}_0^*\widetilde{\mathbf{L}}_0)^{-1}\widetilde{\mathbf{L}}_0^*\mathbf{d}_0,\\ \hat{\mathbf{m}}_1 &= (\widetilde{\mathbf{L}}_1^*\widetilde{\mathbf{L}}_1)^{-1}\widetilde{\mathbf{L}}_1^*\mathbf{d}_1,\end{aligned} \tag{10}$$

This is the so-called *data-space* least-squares migration/inversion method.

### Joint-inversion

Instead of solving the two equations in equation (6) independently, we combine them to form a joint system of equations

$$\begin{bmatrix}\widetilde{\mathbf{L}}_0 & \mathbf{0}\\ \mathbf{0} & \widetilde{\mathbf{L}}_1\end{bmatrix}\begin{bmatrix}\mathbf{m}_0\\ \mathbf{m}_1\end{bmatrix} = \begin{bmatrix}\widetilde{\mathbf{d}}_0\\ \widetilde{\mathbf{d}}_1\end{bmatrix}, \tag{11}$$

for which a solution is obtained by minimizing the objective function

$$S(\mathbf{m_0}, \mathbf{m_1}) = \left\|\begin{bmatrix}\widetilde{\mathbf{L}}_0 & \mathbf{0}\\ \mathbf{0} & \widetilde{\mathbf{L}}_1\end{bmatrix}\begin{bmatrix}\mathbf{m}_0\\ \mathbf{m}_1\end{bmatrix} - \begin{bmatrix}\widetilde{\mathbf{d}}_0\\ \widetilde{\mathbf{d}}_1\end{bmatrix}\right\|^2. \tag{12}$$

Neglecting numerical stability issues, the computational cost of minimizing equations 12 is the same as the cost of minimizing the two objective functions in equation 5. Because several shots are encoded and directly imaged, the computational cost of this approach is considerably reduced relative to non-encoded (or single source) data sets. Equivalent formulations for conventional time-lapse seismic data sets have been shown by previous authors (Ajo-Franklin et al., 2005; Ayeni and Biondi, 2008).

### Regularization and Preconditioning

Seismic inversion is an ill-posed problem. Therefore, regularization operators are required to stabilize the inversion and to prevent divergence to unrealistic solutions. A regularized least squares solution $\hat{\mathbf{m}}$ is obtained by minimizing a modified objective function:

$$S(\mathbf{m}) = \|\widetilde{\mathbf{L}}\mathbf{m} - \widetilde{\mathbf{d}}\|^2 + \epsilon^2\|\mathbf{R}\mathbf{m}\|^2, \tag{13}$$

where $\epsilon$ is a damping factor that determines the strength of the regularization operator $\mathbf{R}$. In this paper, we consider a fixed, heuristically determined, damping factor computed as a function of the data as follows:

$$\epsilon = \frac{max|\widetilde{\mathbf{d}}|}{100}. \tag{14}$$

Relevant examples of regularization criteria for geophysical inverse problems include model smoothness (Tikhonov and Arsenin, 1977), temporal smoothness (Ajo-Franklin et al., 2005), and horizontally smooth angle gathers (Clapp, 2005).

Minimizing equation (13) is equivalent to solving the problem

$$\begin{bmatrix} \widetilde{\mathbf{L}} \\ \epsilon\mathbf{R} \end{bmatrix} \mathbf{m} \; = \begin{bmatrix} \widetilde{\mathbf{d}} \\ \mathbf{0} \end{bmatrix}. \tag{15}$$

Fast iterative convergence can be obtained by *preconditioning the regularization* (Claerbout and Fomel, 2008). This is equivalent to making the variable substitution

$$\mathbf{m} = \mathbf{R}^{-\mathbf{1}}\mathbf{p} = \mathbf{A}\mathbf{p}, \tag{16}$$

so that equation (15) becomes

$$\begin{bmatrix} \widetilde{\mathbf{L}}\mathbf{A} \\ \epsilon\mathbf{I} \end{bmatrix} \mathbf{p} \; = \begin{bmatrix} \widetilde{\mathbf{d}} \\ \mathbf{0} \end{bmatrix}, \tag{17}$$

where $\mathbf{A}$ is the preconditioner, and $\mathbf{p}$ is the preconditioned variable. By selecting an invertible regularization operator $\mathbf{R} = \mathbf{A}^{-\mathbf{1}}$, we can solve the preconditioned problem (equation (17)) at fewer iterations than the regularized problem (equation (15)).

For the current problem, we require two regularization constraints (spatial and temporal). The spatial regularization operator is a system of non-stationary dip filters applied on the helix (Claerbout and Fomel, 2008). These symmetric filters, built from *puck* filters (Hale, 2007; Claerbout and Fomel, 2008), are then factored into causal dip filters using Wilson-Burg factorization (Fomel et al., 2003). The preconditioner, implemented as a helical polynomial division, uses dips estimated from plane-wave destruction (Fomel, 2002) to determine the appropriate filters for each model point. The temporal preconditioner is a bi-directional leaky integration operator which penalizes sudden changes over time.

The preconditioned joint inverse problem is

$$\begin{bmatrix} \text{Ł}\mathring{\mathbf{A}} \\ \epsilon\mathbf{I} \end{bmatrix} \mathbf{p} \; = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix}, \tag{18}$$

where

$$\text{\L} = \begin{bmatrix} \widetilde{\mathbf{L}}_0 & \mathbf{0} \\ \mathbf{0} & \widetilde{\mathbf{L}}_1 \end{bmatrix}, \tag{19}$$

$$\mathring{\mathbf{A}} = \mathbf{AT}, \tag{20}$$

with

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_1 \end{bmatrix}, \tag{21}$$

and

$$\mathbf{T} = \begin{bmatrix} \mathbf{I} & \Lambda \\ \Lambda & \mathbf{I} \end{bmatrix}. \tag{22}$$

The operators $\mathbf{A}_0$ and $\mathbf{A}_1$ are preconditioners for the baseline and monitor images, respectively, while $\mathbf{I}$ is identity and $\Lambda$ is a diagonal operator containing the leak rates $\lambda$. Equation (18) is directly extendable to an arbitrary number of surveys. The proposed method, joint preconditioned least squares inversion (J-PLSI ) refers to the definition in equation (18). We solve equation (18) using a conjugate gradient algorithm.

### Cascaded covariance-based preconditioning

We have specialized the spatial and temporal preconditioners such that the dip-discrimination (or range) of the filters decreases as a function of iteration, while the temporal integration leak rate increases as a function of iteration. This preconditioning approach (which should be applicable to other inversion problems) ensures that close to the solution, the data fitting goal is given more weight relative to the regularization goal.

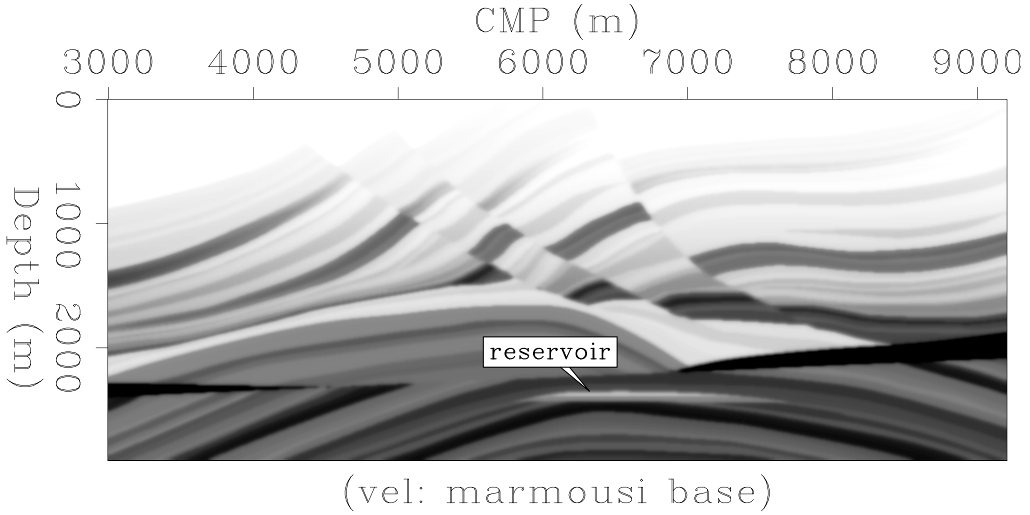
In addition, because non-stationary deconvolution by polynomial division can become unstable at sharp boundaries, the filter range at any image point is a function of dip contrast-dependent covariance. Details of this preconditioning approach is outside the scope of this paper and will be discussed elsewhere.

## NUMERICAL EXAMPLE

We demonstrate J-PLSI using a modified section of 2D Marmousi model (Figure 1). The objective is to image the seismic amplitude changes at the reservoir using simultaneous source data sets. We assume seismic amplitude changes only within the reservoir and neglect geomechanical changes.

The data consist of two sets of 29 encoded shot records over the 8x8 m grid model at different production stages. In this example, because we use a known background baseline velocity model, both data sets are fully encoded. The random encoding function, with a maximum delay of 1 s, is different for each data set. Shot positions vary randomly between surveys with a maximum displacement of 32 m, whereas the receiver array is the same for both surveys and fixed for all shots. Dips were computed from a single source migrated baseline image (not shown).

Figures 2 shows the baseline and monitor data sets. The downgoing baseline and monitor source wavefields at time 0.52 s are shown Figure 3. The single and simultaneous source



(a)

Figure 1: 2D Marmousi velocity model.[**ER**] gayeni1/. marm-vel

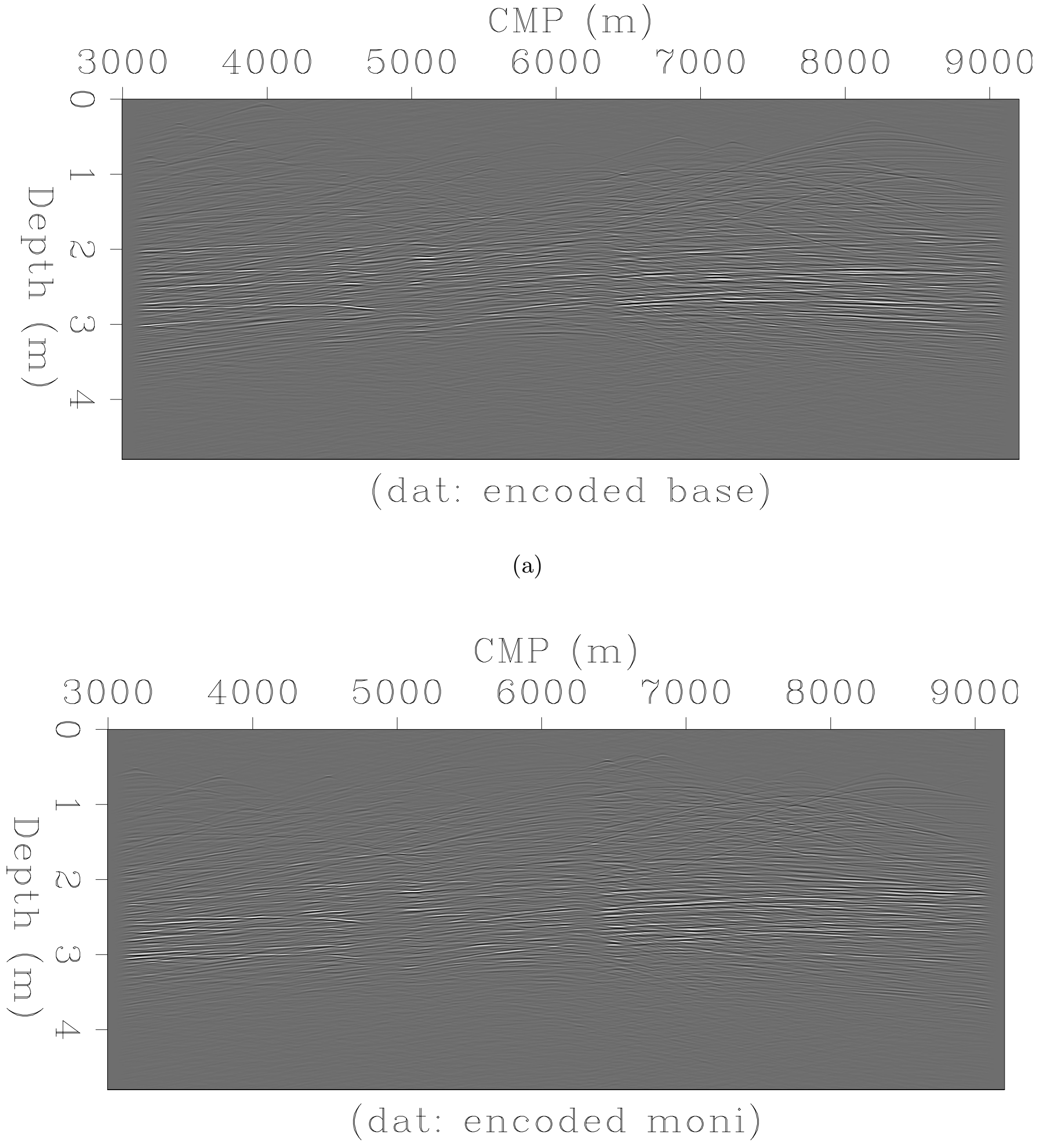
migrated monitor images are shown in Figures 4(a) and 4(b), respectively. The inverted simultaneous source monitor image obtained from J-PLSI is shown in Figure 4(c). Figure 5 shows the migrated and inverted time-lapse images. Note that the cross-term artifacts in the migrated time-lapse image (Figure 5(b)) are significantly attenuated in the inverted image (Figure 5(c)). The weighted RMS difference (non-repeatability) between the migrated and inverted images are shown in Figure 6.

## DISCUSSION AND CONCLUSIONS

We have presented a joint least-squares wave-equation inversion method (J-PLSI) for imaging simultaneous source (or blended) time-lapse seismic data sets. Because it is difficult to repeat both the shot-receiver positions and relative shot-timing for different surveys, there will be significant non-repeatability in simultaneous source data sets. J-PLSI directly inverts the simultaneous source data sets without need for prior separation, combining the cost savings advantages of simultaneous source acquisition and phase-encoded migration.

Direct migration of simultaneous source data generates strong cross-term artifacts (Figure 4(b)) relative to conventional single source data (Figure 4(a)). It is unlikely that *conventional* cross-equalization methods will adequately attenuate these cross-term artifacts while preserving the production-related time-lapse amplitude changes. J-PLSI attenuates the artifacts giving images with better resolution and more balanced amplitudes than migration (Figure 4). This translates to relatively high-quality, high-resolution time-lapse images (Figure 5(c)).

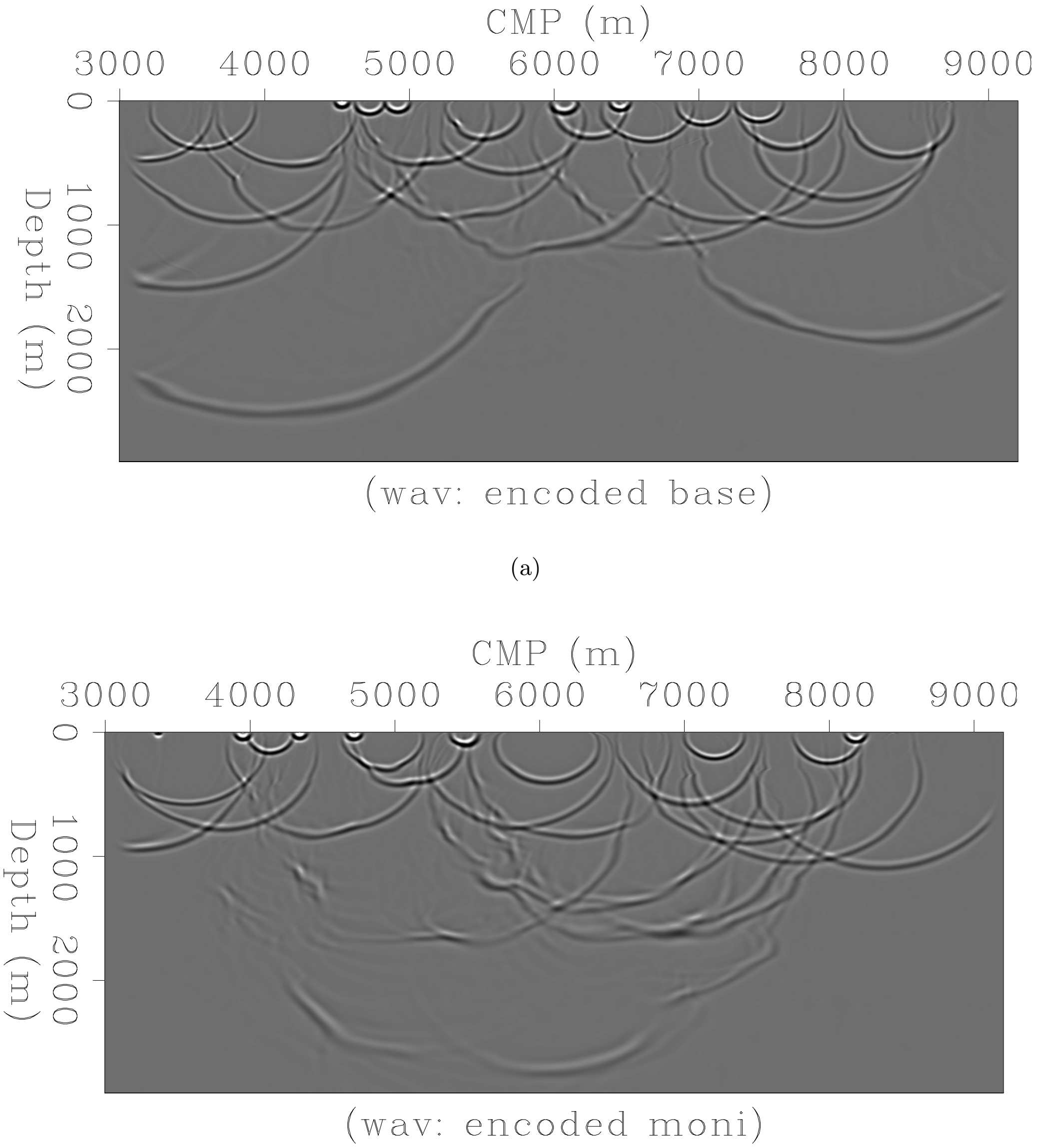
Because simultaneous source acquisition reduces the overall data acquisition cost and the acquired data can be efficiently processed, we recommend shorter survey intervals. By



(a)

(b)

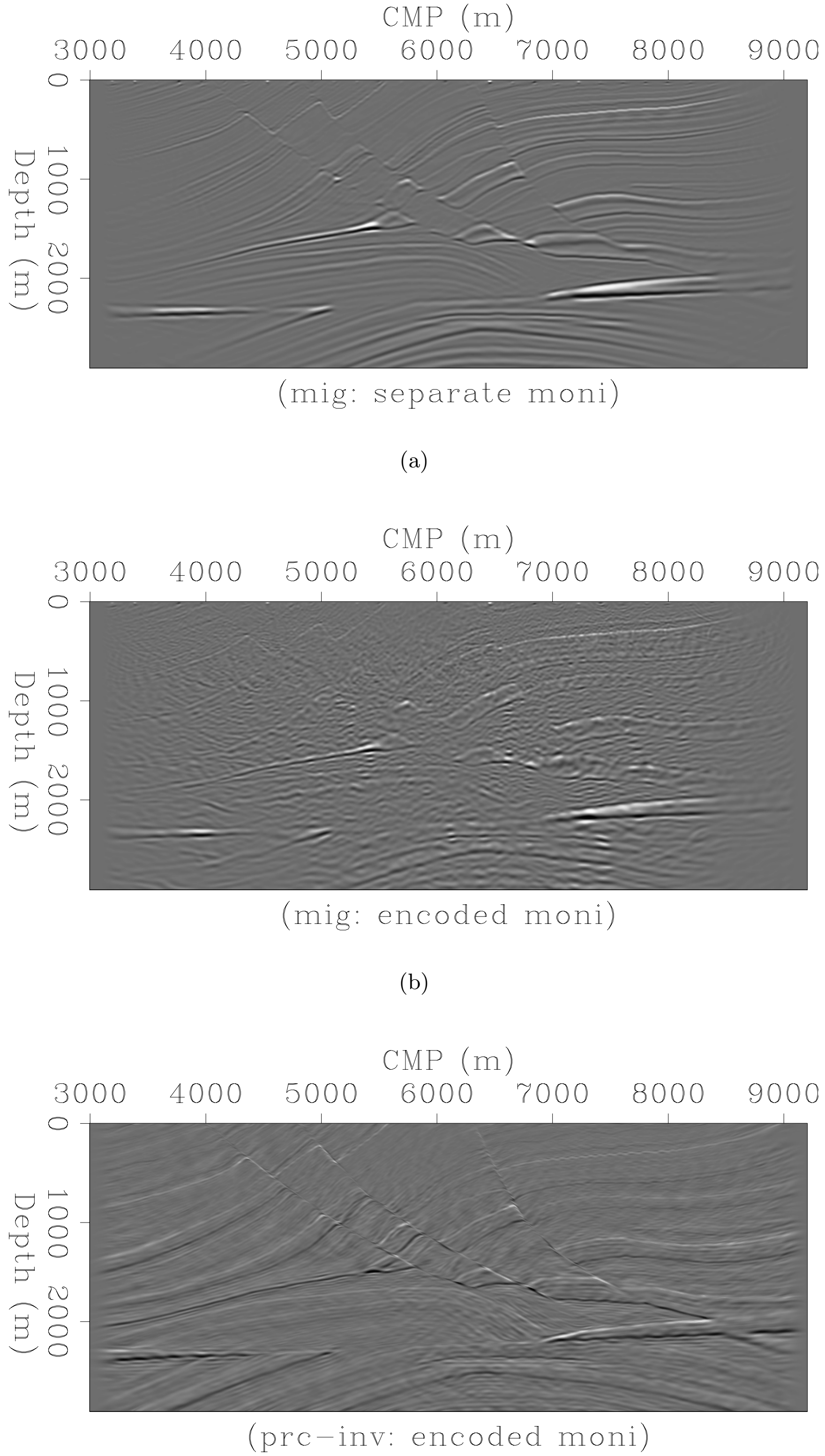
Figure 2: Encoded (a) baseline and (b) monitor data sets.[**CR**] gayeni1/. marm-encoded-dat-1,marm-encoded-dat-2



(a)

(b)

Figure 3: Downgoing source wavefields at 0.52 s for (a) the baseline and (b) monitor data sets. Note the difference in shot-timing.[**CR**] gayeni1/. marm-encoded-wav-1,marm-encoded-wav-2



(a)

(b)

(c)

Figure 4: (a) Single source migrated monitor image. Simultaneous source (b) migrated and (c) inverted monitor images.[**CR**] gayeni1/. marm-nocoded-mig-2,marm-encoded-mig-2,marm-encoded-inv-2

acquiring time-lapse seismic data sets in this manner and processing them using the J-PLSI method, we can get closer to the goal of *continuous* seismic reservoir monitoring. The J-PLSI formulation also provides a suitable framework for simultaneous inversion of multiple seismic and other reservoir monitoring data (e.g. production).

J-PLSI provides gives good-quality inverted time-lapse images at a fraction of cost of least-squares migration of single source data sets. Therefore, it can be applied to conventional single source time-lapse data sets, specifically encoded for computational cost savings. Where such an approach is taken, similar encoding schemes can be used for all data sets and the suitable encoding scheme (Romero et al., 2000) can be chosen to minimize the cross-term artifacts.

## FUTURE DIRECTIONS

A more robust minimization such as L1 will likely yield better quality results than those shown in this paper. We anticipate that a combination of L1-minimization (including both data and model re-weighting) with our preconditioners will lead to a robust inversion scheme for simultaneous source time lapse data sets.

## ACKNOWLEDGEMENTS

We thank Bob Clapp for useful suggestions. We thank sponsors of the Stanford Exploration Project for their financial support.

## REFERENCES

Ajo-Franklin, J. B., J. Urban, and J. M. Harris, 2005, Temporal integration of seismic traveltime tomography: SEG Technical Program Expanded Abstracts, **24**, 2468–2471.

Ayeni, G. and B. Biondi, 2008, Joint wave-equation inversion of time-lapse seismic data: Stanford Exploration Project Report, **136**, 71–96.

Beasley, C. J., R. E. Chambers, and Z. Jiang, 1998, A new look at simultaneous sources: SEG Technical Program Expanded Abstracts, **17**, 133–135.

Berkhout, A. J. G., 2008, Changing the mindset in seismic data acquisition: The Leading Edge, **27**, 924–938.

Berkhout, A. J. G., G. Blacquière, and E. Verschuur, 2008, From simultaneous shooting to blended acquisition: SEG Technical Program Expanded Abstracts, **27**, 2831–2838.

Calvert, R., 2005, Insights and methods for 4D reservoir monitoring and characterization: SEG/EAGE DISC (Distinguished Instructor Lecture Course).

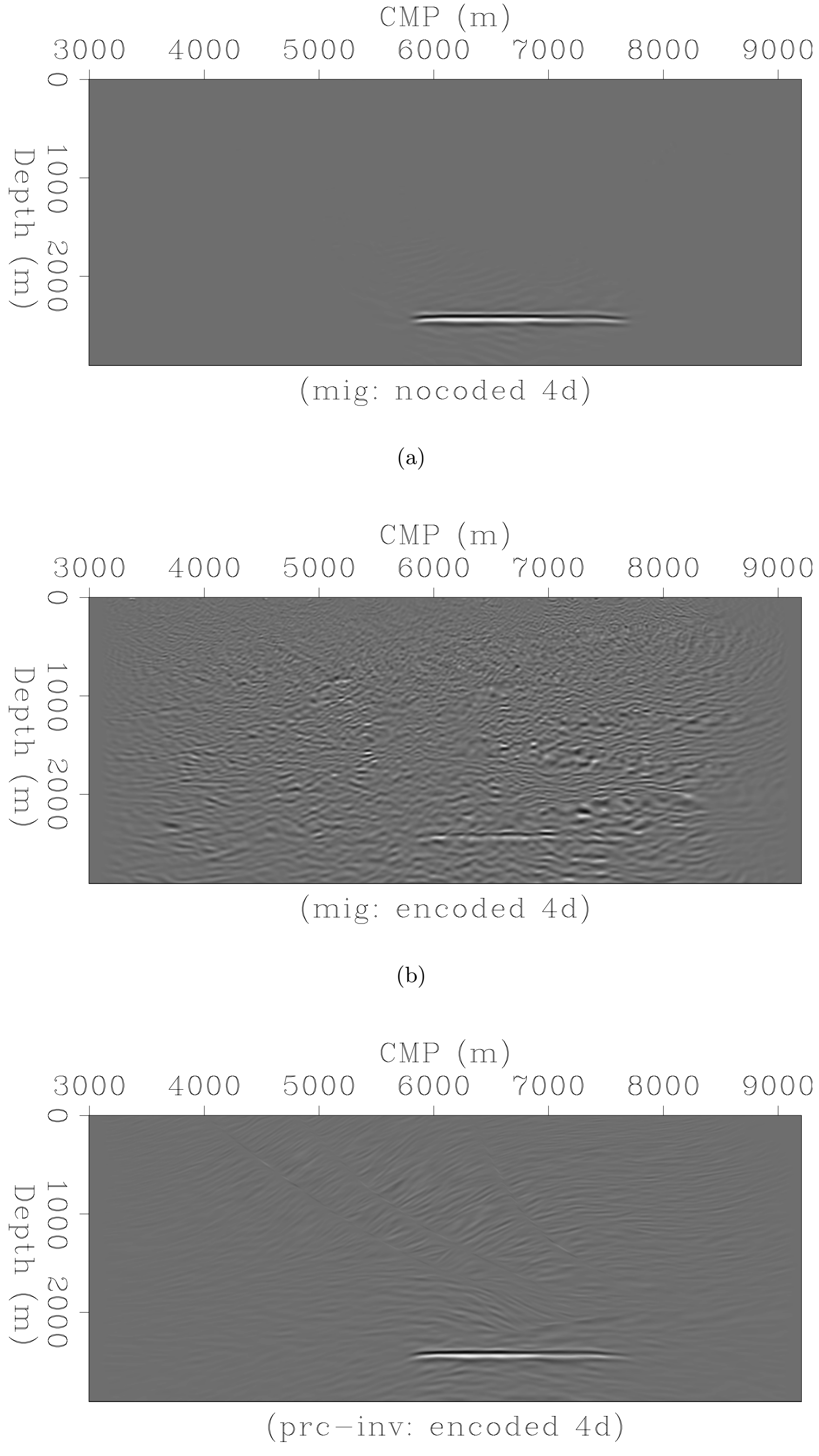
Claerbout, J. F. and S. Fomel, 2008, Image estimation by example: Geophysical soundings image construction: multidimensional autoregression.

Clapp, M. L., 2005, Imaging under salt: illumination compensation by regularized inversion: PhD thesis, Stanford University.

Folland, G. B. and A. Sitaram, 1997, The uncertainty principle: A mathematical survey: Journal of Fourier Analysis and Applications, **3**, 207–238.

Fomel, S., 2002, Applications of plane-wave destruction filters: Geophysics, **67**, 1946–1960.

Fomel, S., P. Sava, J. Rickett, and J. Claerbout, 2003, The Wilson-Burg method of spectral factorization with application to helical filtering: Geophysical Prospecting, **51**, 409–420.

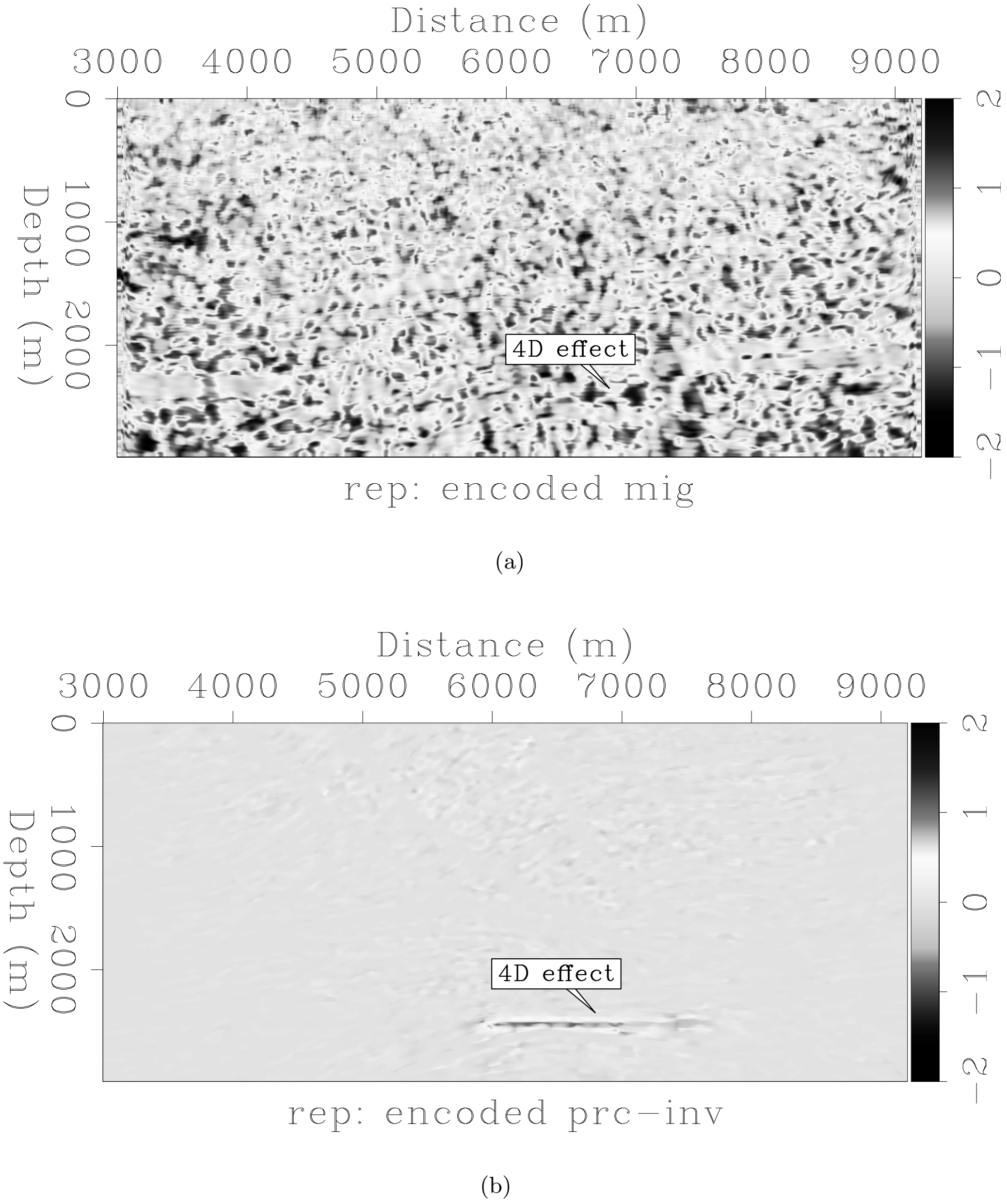


(a)

(b)

(c)

Figure 5: (a) Single source migrated time-lapse image. Simultaneous source (b) migrated and (c) inverted time-lapse images. Note the cross-term artifacts in (b).[**CR**] gayeni1/. marm-nocoded-mig4d,marm-encoded-mig4d,marm-encoded-inv4d



(a)

(b)

Figure 6: Weighted RMS difference between the baseline and monitor images obtained via (a) migration and (b) inversion. Note that inversion attenuates the non-repeatbaility RMS energy and preserves preserving the production-related change.[**CR**]

gayeni1/. marm-encoded-mig-rep-lab,marm-encoded-inv-rep-lab

Hale, D., 2007, Local dip filtering with directional laplacians: CWP Project Reiew, **567**, 91–102.

Hampson, G., J. Stefani, and F. Herkenhoff, 2008, Acquisition using simultaneous sources: SEG Technical Program Expanded Abstracts, **27**, 2816–2820.

Kühl, H. and M. D. Sacchi, 2003, Least-squares wave-equation migration for avp/ava inversion: Geophysics, **68**, 262–273.

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

Plessix, R.-E. and W. Mulder, 2004, Frequency-domain finite-frequency amplitude-preserving migration: Geophysical Journal International, **157**, 975–985.

Rickett, J. E. and D. E. Lumley, 2001, Cross-equalization data processing for time-lapse seismic reservoir monitoring: A case study from the Gulf of Mexico: Geophysics, **66**, 1015–1025.

Romero, L. A., D. C. Ghiglia, C. C. Ober, and S. A. Morton, 2000, Phase encoding of shot records in prestack migration: Geophysics, **65**, 426–436.

Spitz, S., G. Hampson, and A. Pica, 2008, Simultaneous source separation: A prediction-subtraction approach: SEG Technical Program Expanded Abstracts, **27**, 2811–2815.

Tang, Y. and B. Biondi, 2009, Least-squares migration/inversion of blended data: SEG Technical Program Expanded Abstracts, **28**, submitted.

Tikhonov, A. and V. Arsenin, 1977, Solutions of ill-posed problems: V.H. Winston and Sons.

Valenciano, A., 2008, Imaging by Wave-Equation Inversion: PhD thesis, Stanford University.

Versteeg, R., 1994, The Marmousi experience: Velocity model determination on a synthetic complex data set: The Leading Edge, **13**, 927–936.

Whitcombe, D. N., J. M. Marsh, P. J. Clifford, M. Dyce, C. J. S. McKenzie, S. Campbell, A. J. Hill, R. S. Parr, C. Pearse, T. A. Ricketts, C. P. Slater, and O. L. Barkved, 2004, The systematic application of 4D in BP's North-West Europe operations — 5 years on: SEG Technical Program Expanded Abstracts, **23**, 2251–2254.

Womack, J. E., J. R. Cruz, H. K. Rigdon, and G. M. Hoover, 1990, Encoding techniques for multiple source point seismic data acquisition: Geophysics, **55**, 1389–1396.

Zou, Y., L. R. Bentley, L. R. Lines, and D. Coombe, 2006, Integration of seismic methods with reservoir simulation, Pikes Peak heavy-oil field, Saskatchewan: The Leading Edge, **25**, 764–781.

# Target-oriented least-squares migration/inversion with sparseness constraints

*Yaxun Tang*

## ABSTRACT

I pose the seismic imaging problem as an inverse problem and present a regularized inversion scheme that tries to overcome three main practical issues with the standard least-squares migration/inversion (LSI) approach, i.e., the high computational cost, the operator mismatch, and the poorly constrained solution due to a limited surface acquisition geometry. I show that the computational cost is considerably reduced by formulating the LSI problem in a target-oriented fashion and computing a truncated Hessian operator using the phase-encoding method. The second and third issues are mitigated by introducing a non-quadratic regularization operator that imposes sparseness to the model parameters. Numerical examples on the Marmousi model show that the sparseness constraint has the potential to effectively reduce the null space and produce an image with high resolution, but it also has the risk of over-penalizing weak reflections.

## INTRODUCTION

Migration is an important and robust tool for imaging subsurface structures using reflection seismic data. However, migration operator is only the adjoint of the forward Born modeling operator (Lailly, 1983), which produces reliable structural information of the subsurface (assuming an accurate background velocity is known), but blurs the image because of the non-unitary nature of the Born modeling operator. To deblur the migrated image and correct the effects of limited acquisition geometry, complex overburden and bandlimited wavefields, the imaging problem can be treated as an inverse problem, which, instead of using the adjoint operator, uses the pseudo-inverse of the Born modeling operator to optimally reconstruct the reflectivity. This inversion-based imaging mehtod is also widely known as least-squares migration (Nemeth et al., 1999; Kuhl and Sacchi, 2003; Clapp, 2005; Valenciano, 2008).

The standard least-squares migration/inversion (LSI) approach tries to minimize an objective function defined in the data space, which compares the mismatch between the modeled and the observed primaries (Nemeth et al., 1999; Kuhl and Sacchi, 2003; Clapp, 2005). The objective function is then minimized with a gradient-based optimization solver, which iterates until an acceptable image is obtained. However, the data-space inversion scheme lacks flexibility and cannot be implemented in a target-oriented fashion. Full-domain migration/demigration has to be carried out within each iteration; and the optimization converges slowly without a proper preconditioner. Therefore, the data-space inversion scheme is computationally challenging for large-scale applications.

One way to reduce the computational cost is by solving the LSI problem in a target-oriented fashion (Yu et al., 2006; Valenciano, 2008). This can be achieved by minimizing an objective function defined in the model space, instead of the data space. The target-oriented model-space formulation allows us to invert only areas of particular interest, such as subsalt regions, where potential reservoirs are located and migration often fails to provide reliable images. Solving the LSI in the model space requires explicitly computing the Hessian, the normal operator of the forward Born modeling operator. The full Hessian, however, is expensive to compute without certain approximations. Fortunately, as demonstrated by Valenciano (2008) and Tang and Biondi (2009), for a typical conventional acquisition geometry (shot records do not interfer), the Hessian matrix is sparse and diagonally dominant for most areas. Thus a truncated Hessian with a limited number of off-diagonal elements (the number is usually very small) can be used to approximate the exact Hessian for inverse filtering.

The truncated Hessian can be computed by storing the Green's functions (Valenciano, 2008), which, however, may bring considerable computational issues (e.g. disk storage, I/O and etc.), because the Green's functions can be huge for practical applications, especially in 3-D. To reduce the computational overburden, this paper computes the Hessian using the phase-encoding method (Tang, 2008b). As demonstrated by Tang (2008b) and Tang (2008a), computing the phase-encoded Hessian does not require storing any Green's functions and it is also more efficient: the cost for computing the receiver-side randomly phase-encoded Hessian is about one shot-profile migration, and if a mixed simultaneous phase-encoding strategy is used, the cost is about one plane-wave source migration.

Besides the computational cost, two main issues, i.e., the operator mismatch and the underdetermined nature of the seismic inverse problem, make the practical application of LSI less effective. The first issue often arises when our modeling operator is not sufficient to predict the physics of the data, for example, anisotropy or elasticity presents in the data but is not accurately modeled by our numerical operators. This can cause data-inconsistency problems. The second issue is due to the limited surface seismic acquisition geometry, which makes the inversion have an infinite number of solutions that fit the observed data equally well. Regularization is therefore important to stabilize the inversion and make it converge to geologically reasonable solutions. In this paper, I exploit the application of a non-quadratic regularization operator that imposes sparsness to the model space (Sacchi and Ulrych, 1995; Ulrych et al., 2001). The model-space sparsity is achieved by minimizing the model residual in the $\ell_1$ or Cauchy norm, whose distribution is longer-tailed than the Gaussian distribution (the $\ell_2$ norm), hence it penalizes weak energy and leads to spiky solutions (Amundsen, 1991). The application of the sparseness constraint to seismic imaging has also been reported by Tang (2006) and Wang and Sacchi (2007), who use it to regularize prestack image gathers. In this paper, however, I use it to regularize the prestack image (zero subsurface offset) to enhance the resolution of the inverted reflectivity. I compare the one-way wave-equation inversion results on the Marmousi model regularized using the sparseness constraint with those regularized using a standard $\ell_2$ norm damping. The experiments have been carried out on data sets synthesized using both one-way wave-equation Born modeling and two-way acoustic wave-equation finite-difference modeling. The first case represent the ideal scenario, where our modeling operator (one-way wave-equation propagator for this case) matches all the physics in the data. I show that both inversion schemes work well under this situation, and sparseness constrained inversion can offer slightly higher resolution. The second case is much more challenging for both schemes, because our modeling operator can not model all the complexities present in the data (e.g., amplitudes, multiples and etc.). My experiments show that under this difficult situation, the sparseness-constrained approach provides us a better inversion result for the Marmousi model, which suggests the importance of accurate model covariance (or the *a priori* information) for the LSI problem.

This paper is organized as follows: I first briefly review the theory of target-oriented LSI and phase-encoded Hessian, then I discuss the sparseness constraint which minimizes the model residual in the Cauchy norm. Finally I apply the regularized inversion scheme to the Marmousi model.

## TARGET-ORIENTED LEAST-SQUARES MIGRATION

Within limits of the Born approximation of the acoustic wave equation, the seismic data can be modeled with a linear operator as follows

$$\mathbf{d} = \mathbf{L}\mathbf{m}, \tag{1}$$

where $\mathbf{d}$ is the modeled data, $\mathbf{L}$ is the Born modeling operator and $\mathbf{m}$ denotes the reflectivity of the subsurface (a perturbed quantity from the background velocity). The simplest way is to use the adjoint of the Born modeling operator to image the reflectivity $\mathbf{m}$ as follows:

$$\mathbf{m}_{\mathrm{mig}} = \mathbf{L}'\mathbf{d}_{\mathrm{obs}}, \tag{2}$$

where the superscript denotes the conjugate transpose and the subscript $_{\mathrm{obs}}$ denotes observed data. However, migration produces unreliable images in areas of poor illumination. To get an optimally reconstructed image, we can invert equation 1 in the least-squares sense. The least-squares soltuion of equation 1 can be formally written as follows

$$\mathbf{m} = \mathbf{H}^{-1}\mathbf{m}_{\mathrm{mig}}, \tag{3}$$

where $\mathbf{H} = \mathbf{L}'\mathbf{L}$ is the Hessian operator. Equation 3 has only symbolic meaning, because the Hessian is often singular and its inverse is not easy to obtain directly. A more practical method is to reconstruct the reflectivity $\mathbf{m}$ through iterative inverse filtering by minimizing a model-space objective function defined as follows:

$$J(\mathbf{m}) = ||\mathbf{H}\mathbf{m} - \mathbf{m}_{\mathrm{mig}}||_2^2, \tag{4}$$

where $||\cdot||_2$ denotes the $\ell_2$ norm. Each component of the Hessian matrix $\mathbf{H}$ can be computed with the following equation, which is obtained by evaluating the operator $\mathbf{L}'\mathbf{L}$ (Plessix and Mulder, 2004; Valenciano, 2008):

$$\begin{aligned} H(\mathbf{x},\mathbf{y}) = \quad & \sum_{\omega} \omega^4 \sum_{\mathbf{x}_s} |f_s(\omega)|^2 G(\mathbf{x},\mathbf{x}_s,\omega) G'(\mathbf{y},\mathbf{x}_s,\omega) \\ \times \quad & \sum_{\mathbf{x}_r} w(\mathbf{x}_r,\mathbf{x}_s) G(\mathbf{x},\mathbf{x}_r,\omega) G'(\mathbf{y},\mathbf{x}_r,\omega), \end{aligned} \tag{5}$$

where $\omega$ is the angular frequency, and $f_s(\omega)$ is the source function; $G(\mathbf{x},\mathbf{x}_s,\omega)$ and $G(\mathbf{x},\mathbf{x}_r,\omega)$ denote Green's functions connecting the source location $\mathbf{x}_s = (x_s, y_s, 0)$ and receiver location $\mathbf{x}_r = (x_r, y_r, 0)$ to the image point $\mathbf{x}$, respectively. We have similar definitions for $G(\mathbf{y},\mathbf{x}_s,\omega)$ and $G(\mathbf{y},\mathbf{x}_r,\omega)$, except that they define the Green's functions connecting the source and receiver locations to another image point $\mathbf{y}$ in the subsurface. Throughout this paper, we assume the Green's functions are computed by means of one-way wavefield extrapolation (Claerbout, 1985; Stoffa et al., 1990; Ristow and Rühl, 1994). But Green's functions obatined with other methods, such as the ray-based approach, the two-way wave-equation-based approah and etc., can also be used under this framework. The weighting factor $w(\mathbf{x}_s, \mathbf{x}_r)$ denotes the acquisition mask matrix (Tang, 2008a) defined as follows:

$$w(\mathbf{x}_r, \mathbf{x}_s) = \begin{cases} 1 & \text{if } \mathbf{x}_r \text{ is within the recording} \\ & \text{range of a shot at } \mathbf{x}_s; \\ 0 & \text{otherwise .} \end{cases} \tag{6}$$

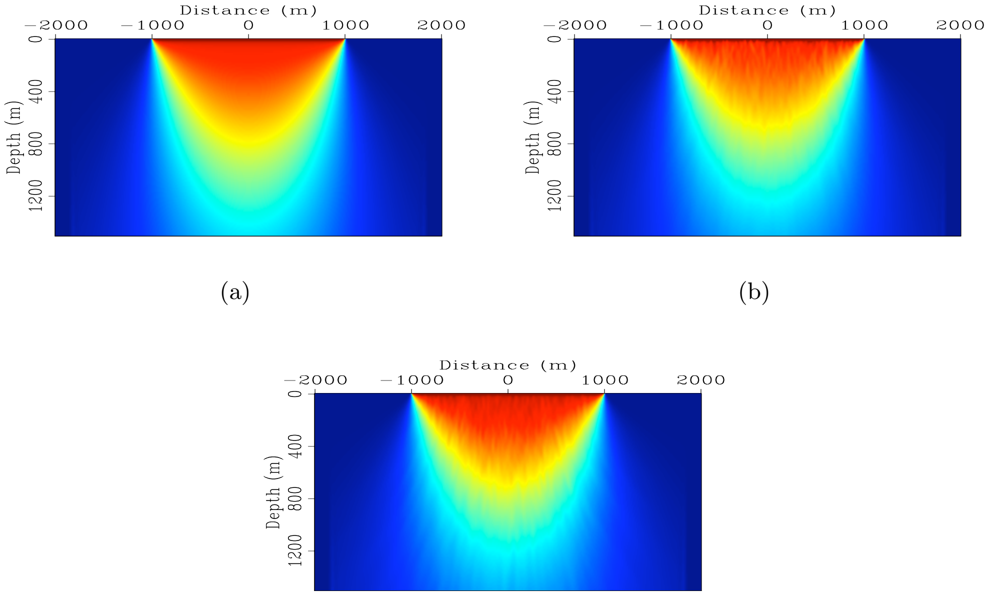
When $\mathbf{x} = \mathbf{y}$, we obtain the diagonal elements of the Hessian; when $\mathbf{x} \neq \mathbf{y}$, we obtain the off-diagonal elements. A target-oriented truncated Hessian is obtained by computing the Hessian for $\mathbf{x}$'s that are within the target zone and a small number of $\mathbf{y}$'s that are close to each $\mathbf{x}$ (Valenciano, 2008).

## HESSIAN BY PHASE ENCODING

The truncated Hessian operator can be computed by using equation 5, but direct implementation of equation 5 requires storing a huge number of Green's functions (especially in 3-D), which may bring computational challenges for large-scale applications. An alternative and also more efficient way is to compute the Hessian using the so-called phase-encoding method (Tang, 2008b,a), where equation 5 is structured into a similar form as that of the wave-equation migration, except for a modified boundary condition for the receiver wavefield and a modified imaging condition which correlates four wavefields instead of two. Doing so makes storing Green's functions unnecessary, and the cost for computing a target-orineted wave-equation Hessian becomes comparable to one migration.

As further discussed by Tang (2008a), the phase-encoded Hessian is equivalent to the imaging Hessian in the generalized source and receiver domain, a transformed domian that is obtained by linear combination of the encoded sources and receivers. Different phase-encoded Hessian therefore can be obtained through different encoding strategies: if the encoding is performed in the source domain, we get the source-side encoded Hessian; if the encoding is performed in the receiver domain, we get the receiver-side encoded Hessian; if the encoding is performed in both source and receiver domain, we get the source- and receiver-side simultaneously encoded Hessian. One shortcoming of the encoding method, however, is that it also introduces undesired crosstalk artifacts, which may affect the convergence of the model-space based inversion (Tang, 2008b). The crosstalk artifacts can be effectively suppressed by carefully choosing the phase-encoding functions. As demonstrated by Tang (2008b,a), plane-wave-phase encoding or random-phase encoding or a combination of the two can effectively attenuate the crosstalk.

Figure 1 compares diagonal parts of the exact Hessian (Figure 1(a)) obtained using equation 5 and the phase-encoded Hessians (Figure 1(b) for the receiver-side randomly phase-encoded Hessian and Figure 1(c) for the simultaneously phase-encoded Hessian with a mixed encoding strategy) for a simple model with a constant velocity of 2000 m/s. The acquisition geometry consists of 201 shots from −1000 m to 1000 m with a 10 m sampling and 201 receivers also spanning from −1000 m to 1000 m with a 10 m sampling. Figure 2 compares the off-diagonal elements (a row of the truncted Hessian matrix) for image point at $x = 0$ m, $z = 800$ m. The size of the filter is $21 \times 21$ in $x$ and $z$ directions. The comparisons show that besides lower computational cost, the phase-encoded Hessians are good approximations to the exact truncated Hessian.



(a)

(b)

(c)

Figure 1: The diagonal part of the Hessian for a constant-velocity model. (a) The exact Hessian; (b) the receiver-side randomly phase-encoded Hessian and (c) the simultaneously phase-encoded Hessian with a mixed phase encoding which combines both random and plane-wave encoding functions. [**CR**] yaxun2/. hess-exact,hess-random,hess-simul-mixed

## REGULARIZATION WITH SPARSENESS CONSTRAINTS

Inverting the linear system defined by equation 4 is difficult, because it is underdetermined due to the incomplete subsurface illumination caused by the limited surface acquisition and complex overburden. Another difficulty arises when our Born modeling operator $\mathbf{L}$ is not sufficient to model all the complexities in the observed data $\mathbf{d}_{\text{obs}}$. For example, the commonly used one-way wave-equation propagator is based on acoustic assumption and cannot handle waves beyond 90 degrees; its amplitude is also not accurate for wide angles propagations (Zhang et al., 2005). The operator mismatch can make the inversion unstable. Of course, adding more data and using more accurate modeling operators can always help, but a more cost effective way would be introducing regularization operators that impose the *a priori* information to stabilize the inversion and make it converge to a geologically reasonable solution. A widely used regularization is the $\ell_2$-norm damping, which minimizes the energy of the model parameters by introducing a secondary objective function, and the overal objective function to minimize becomes

$$J(\mathbf{m}) = ||\mathbf{H}\mathbf{m} - \mathbf{m}_{\text{mig}}||_2^2 + \epsilon||\mathbf{m}||_2^2, \tag{7}$$

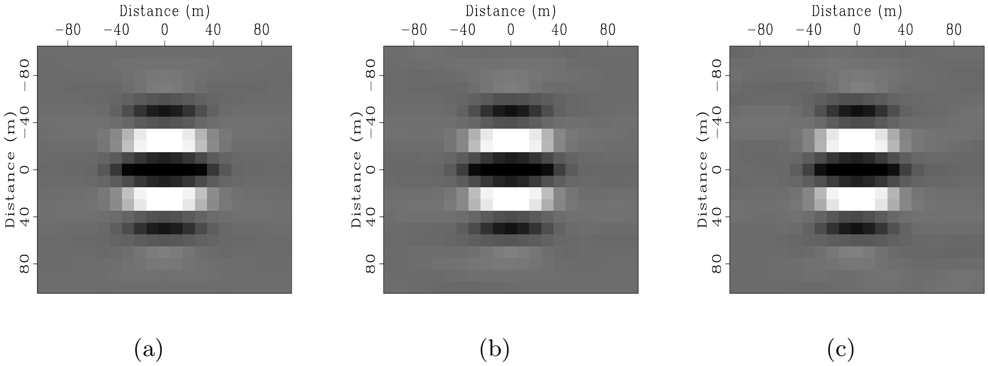


Figure 2: The off-diagonal elements of the Hessian for a image point (a row of the Hessian). (a) The exact Hessian; (b) the receiver-side randomly phase-encoded Hessian and (c) the simultaneously phase-encoded Hessian with a mixed phase encoding which combines both random and plane-wave encoding functions. [**CR**] yaxun2/. hess-exact-offd1,hess-random-offd1,hess-simul-mixed-offd1

where $\epsilon$ is a trade-off parameter that controls the strength of regularization. The $\ell_2$-norm damping assumes the statistic of the reflectivity has a Gaussian distribution, which often leads to a relatively smooth solution. If we assume that the reflectivity is made up of spikes (Oldenburg et al., 1981), then the short-tailed Gaussian distribution assumption becomes unappropriate. To obtain a spiky or sparse solution, a long-tailed distribution such as exponential (the $\ell_1$ norm) or Cauchy (the Cauchy norm) distribution should be used (Sacchi and Ulrych, 1995). The objective function with a regularization in the Cauchy norm reads

$$J(\mathbf{m}) = ||\mathbf{Hm} - \mathbf{m}_{\text{mig}}||_2^2 + \epsilon S(\mathbf{m}), \tag{8}$$

where $S(\mathbf{m})$ is a non-quadratic regularization function defined as follows:

$$S(\mathbf{m}) = \sum_{\mathbf{x}} \log(1 + m^2(\mathbf{x})/\sigma^2), \tag{9}$$

in which $\sigma^2$ is a scalar parameter of the Cauchy distribution that controls the sparsity of the model. The objective function 8 can be minimized under $\ell_2$ norm with the iterative reweighted least-squares (IRLS) technique (Darche, 1989; Nichols, 1994; Scales and Smith, 1994; Guitton, 2000), which equivalently minimizes the following non-linear objective function:

$$J(\mathbf{m}) = ||\mathbf{Hm} - \mathbf{m}_{\text{mig}}||_2^2 + \epsilon||\mathbf{Qm}||_2^2, \tag{10}$$

where $\mathbf{Q}$ is a model dependent diagonal operator defined as follows:

$$\mathbf{Q} = \mathbf{diag}\left(\frac{1}{\sqrt{1 + m^2(\mathbf{x})/\sigma^2}}\right). \tag{11}$$

The detailed implementation of IRLS can be found in Darche (1989); Nichols (1994); Scales and Smith (1994); Guitton (2000).

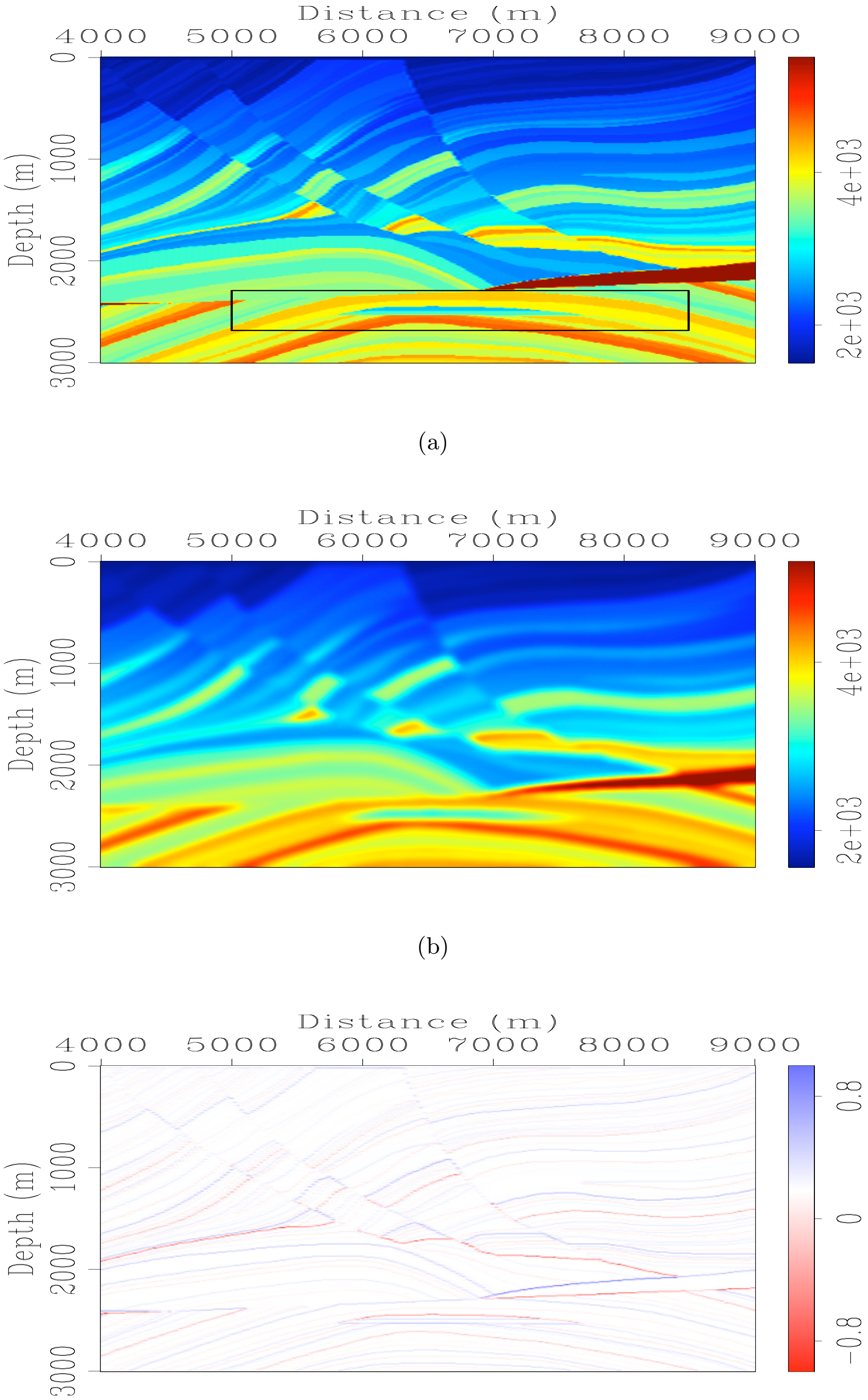
## NUMERICAL EXAMPLES

I test both regularized target-oriented inversion schemes (equation 7 and 8) on the Marmousi model. Two data sets are synthesized: the first one is generated using one-way wave-equation Born modeling, while the second one is generated using two-way acoustic wave-equation finite-difference modeling. Figure 3(a) shows the stratigraphic velocity model used for the two-way wave-equation modeling. Figure 3(b) and Figure 3(c) show the corresponding background velocity model (the low frequency component of Figure 3(a)) and the reflectivity model (the high frequency component of Figure 3(a)) for the one-way wave-equation Born modeling. For both data sets, I model 251 shots ranging from 4000 m to 9000 m with a 20 m sampling. The receiver spread is fixed for all shots and spans from 4000 m to 9000 m with a 10 m sampling. Figure 4 compares the modeled shot gathers located at 6500 m. Note the amplitude differences between both data. Also note that some complexities present in two-way finite-difference modeled data are not modeled using the one-way Born modeling.

The target zone selected for inversion tests is outlined with a small box in Figure 3(a), a close-up look is also shown in Figure 5(a). The target zone is where the reservoir locates. The target-oriented Hessian is computed using the receiver-side random-phase encoding (Tang, 2008b,a). The smooth background velocity model (Figure 3(b)) and the Fourier finite-difference (FFD) one-way extrapolator (Ristow and Rühl, 1994) are used for migrating both one-way and two-way data and also for the Hessian computation. Figure 5(b) illustrates the diagonal elements of the phase-encoded Hessian for the target area (the amplitude is normalized). Note the uneven illumination due to the limited acquisition geometry and complex velocity model. Figure 6 shows the truncated local Hessian filters for three different image points (three rows of the truncted Hessian). The size of the filter is $31 \times 31$ in $x$ and $z$ directions, which seems to be big enough to capture most of the energy in the Hessian matrix.

Figure 7 shows the inversion results on the one-way wave-equation Born-modeled data. This example represents the ideal case for one-way wave-equation inversion, since our modeling operator can "explain" all the physics present in the "observed" data (Figure 4(a)). As expected, migration produces a blurred image (Figure 7(b)); the regularized inversion schemes optimally deblur the migrated image, and the reflectivity is better recovered (Figure 7(c) and Figure 7(d)). Note that both inversion schemes enhance the spatial resolution. Also note that regularization with the sparseness constraint produces slightly higher resolution than regularization with the standard $\ell_2$-norm damping and Figure 7(d) is closer to the true reflectivity shown in Figure 8(a). This suggests that the sparseness constraint better predicts the model covariance, so that it more effectively reduces the null space and provides more accurate inversion result.

More interesting and also more instructive examples are shown in Figure 8, where both regularized inversion schemes are applied to the data synthesized using the two-way wave-equation finite-difference modeling (Figure 4(b)). In this case, the one-way wave-equation migrated image (Figure 8(b)) is much noisier than the corresponding result using the one-way Born data (Figure 7(b)); the amplitudes are also more distorted. This phenomenon is due to the operator mismatch, where the internal multiples and wide angle propagations cannot be modeled by the one-way Born modeling operator. Consequently, they contribute to the artifacts shown in Figure 8(b). The operator mismatch also influences the inversion



(a)

(b)

(c)

Figure 3: The Marmousi model. Panel (a) is the stratigraphic velocity model used for two-way wave-equation finite-difference modeling. Panels (b) and (c) are the background velocity model and reflectivity model used for one-way wave-equation Born modeling. [**ER**] yaxun2/. marm-vmod-stra,marm-vmod,marm-refl

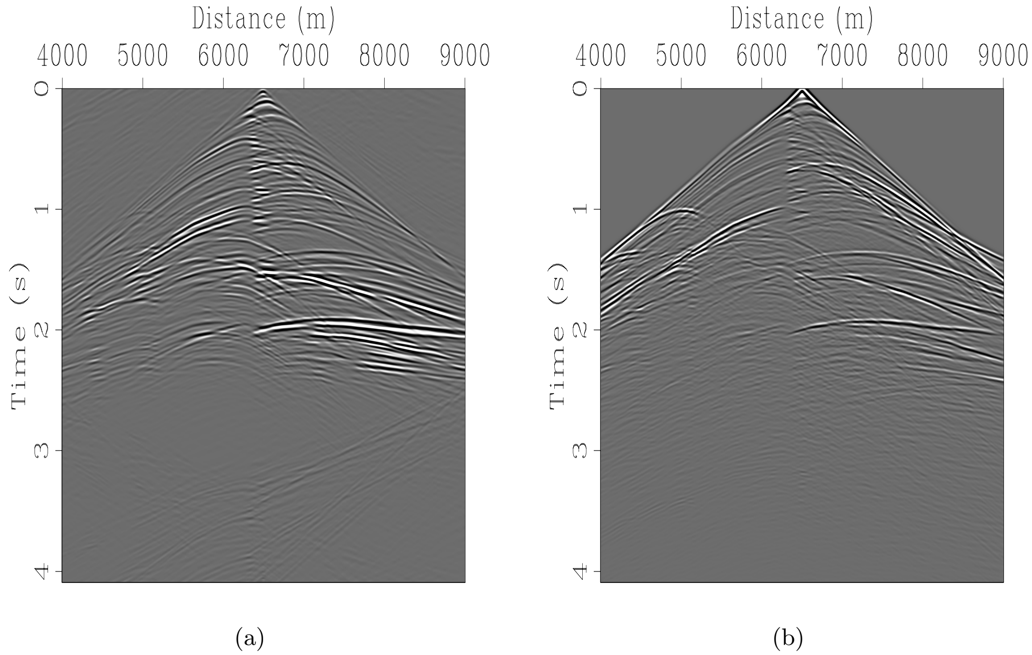
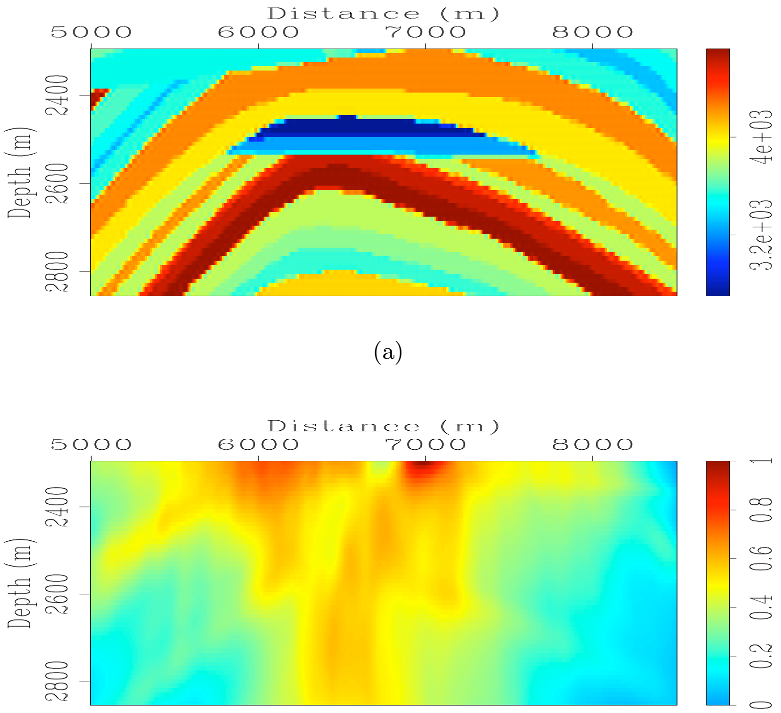


Figure 4: Comparison between shots synthesized using (a) one-way wave-equation Born modeling and (b) two-way wave-equation finite-difference modeling. [**CR**] yaxun2/. marm-trec-one-way,marm-trec-two-way

results, as shown in Figure 8(c) and Figure 8(d). The inverted images are noisier and have more artifacts compared to the results obtained on the one-way Born data. But noticeable improvement on resolution over migrated image (Figure 8(b)) can still be identified. Note that inversion regularized with the sparseness constraint seems to provide a less noisy image with slightly higher spatial resolution than the inverted image regularized with the $\ell_2$-norm damping. This example suggests that when we have operator mismatch issues for inverse problems, it is important to add regularization operators that more accurately predict the model covariance. In this particular example, although promoting sparsity may not be the best regularization, it does better predicts the model covariance than the $\ell_2$-norm damping, hence it produces a better result even when our operator is not able to fully explain the observed data.

## DISCUSSION

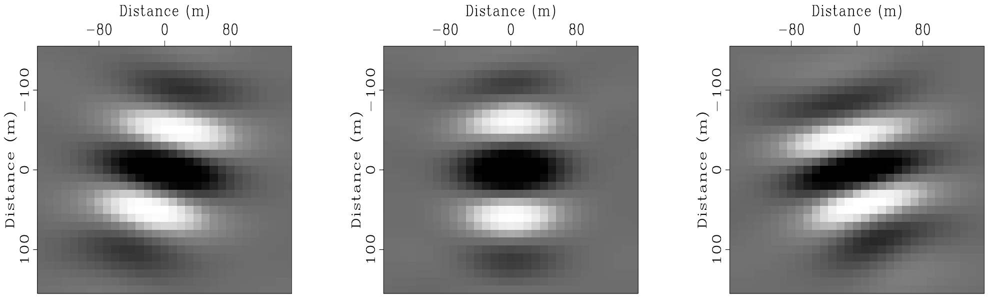
This paper presents a sparseness constrained LSI scheme that promotes sparsity of the reflectivity. This is a reasonable assumption if the reflectivity is indeed spiky; however, if the reflectivity changes smoothly, the sparseness constraint may lead to a biased solution. The parameters $\sigma$ and $\epsilon$ that control the strength of sparsity and the amount of regularization should also be chosen with extreme care. Because by promoting sparsity, we run the risk of penalizing true reflections that have very weak energy, over-regularization may lead to too-sparse solutions, forfeiting the ability to image weak reflections.



(a)

(b)

Figure 5: (a) The stratigraphic velocity model for the target zone. (b) The diagonal of the Hessian for the target zone. [**CR**] yaxun2/. marm-stra-target,marm-hess-diag-target

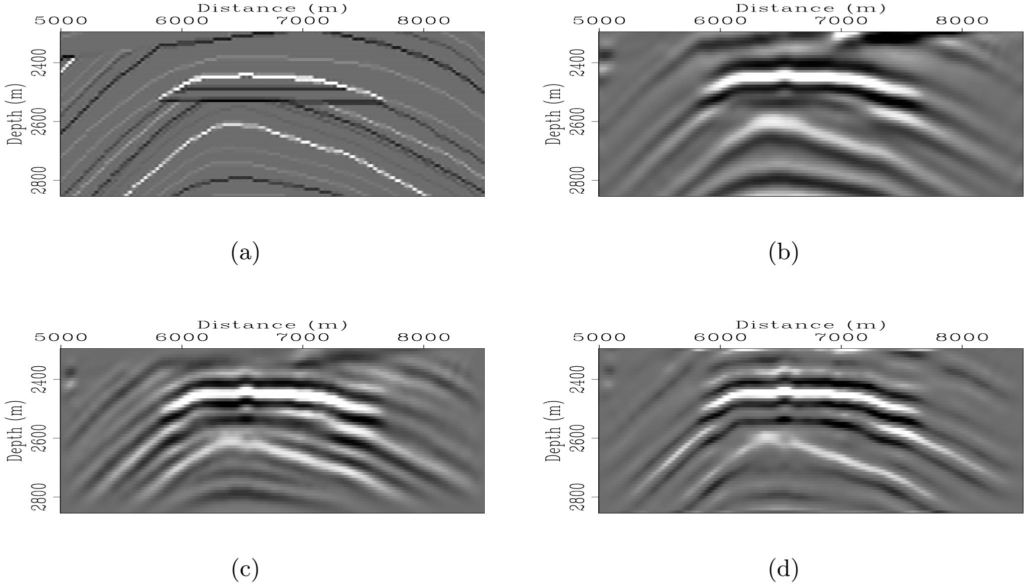


(a)

(b)

(c)

Figure 6: The local Hessian filters at (a) $x = 5250$ m, $z = 2800$ m, (b) $x = 6500$ m, $z = 2600$ m and (c) $x = 8400$ m, $z = 2400$ m. [**CR**] yaxun2/. marm-hess-offd-target-1,marm-hess-offd-target-2,marm-hess-offd-target-3



(a)

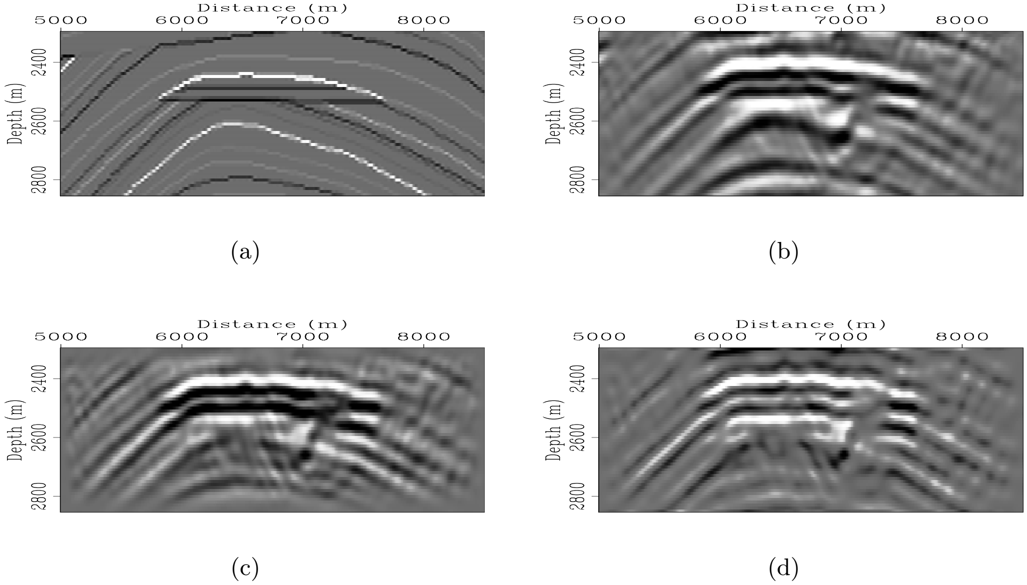
(b)

(c)

(d)

Figure 7: Target-orineted inversion of the one-way wave-equation Born-modeled data. (a) The true reflectivity, (b) migration, (c) inversion regularized with $\ell_2$ norm damping (equation 7) and (d) inversion regularized with the sparseness constraint (equation 8). [**CR**]

yaxun2/. marm-refl-target,marm-imag-one-way-target,marm-invt-one-way-target-l2,marm-invt-one-way-targe



(a)

(b)

(c)

(d)

Figure 8: Target-orineted inversion of the two-way wave-equation finite-difference modeled data. (a) The true reflectivity, (b) migration, (c) inversion regularized with $\ell_2$ norm damping (equation 7) and (d) inversion regularized with the sparseness constraint (equation 8). [**CR**]

yaxun2/. marm-refl-target,marm-imag-two-way-target,marm-invt-two-way-target-l2,marm-invt-two-way-targe

Recent study in curvelet (Kumar and Herrmann, 2008) and seislet (Fomel, 2006) transforms show that seismic images tend to have a sparse representation in these new domains, where a few number of coefficients are sufficient to describe images with complex structures. This feature makes these new domains good candidates for adding sparseness constraints. Therefore, promoting sparsity in either curvelet or seislet domain may potentially avoid the issues discussed before and lead to geologically more reasonable solutions. This remains a research area for further investigation.

## CONCLUSIONS

I have presented a regularized least-squares inversion scheme to image the reflectivity. This inversion scheme allows us to perform inversion in a target-oriented fashion, and the total cost is about two migrations (one for computing the migrated image, the other for computing the phase-encoded Hessian). Examples on the Marmousi model show that regularization that promotes sparsity in the image domain help to reduce the null space and to mitigate the effects of operator mismatch. Inversion with the sparseness constraint can lead to a better solution with higher resolution than that regularized with the standard $\ell_2$-norm damping.

## REFERENCES

Amundsen, L., 1991, Comparison of the least-squares criterion and the Cauchy criterion in frequency-wavenumber inversion: Geophysics, **56**, 2027–2035.

Claerbout, J. F., 1985, Imaging the earth's interior: Blackwell Scientific Publication.

Clapp, M. L., 2005, Imaging Under Salt: Illumination Compensation by Regularized Inversion: PhD thesis, Stanford University.

Darche, G., 1989, Iterative L1 deconvolution: **SEP-61**, 281–301.

Fomel, S., 2006, Towards the seislet transform: SEG Technical Program Expanded Abstracts, **25**, 2847–2851.

Guitton, A., 2000, Huber solver versus IRLS algorithm for quasi L1 inversion: **SEP-103**, 205–266.

Kuhl, H. and M. D. Sacchi, 2003, Least-squares wave-equation migration for AVP/AVA inversion: Geophysics, **68**, 262–273.

Kumar, V. and F. J. Herrmann, 2008, Deconvolution with curvelet-domain sparsity: SEG Technical Program Expanded Abstracts, **27**, 1996–2000.

Lailly, P., 1983, The seismic inverse problem as a sequence of before stack migration: Proc. Conf. on Inverse Scattering, Theory and Applications, Expanded Abstracts, Philadelphia, SIAM.

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

Nichols, D., 1994, Velocity-stack inversion using Lp norms: **SEP-82**, 1–16.

Oldenburg, D. W., S. Levy, and K. P. Whittall, 1981, Wavelet estimation and deconvolution: Geophysics, **46**, 1528–1542.

Plessix, R.-E. and W. A. Mulder, 2004, Frequency-domain finite-difference amplitude-preserving migration: Geophys. J. Int., **157**, 975–987.

Ristow, D. and T. Rühl, 1994, Fourier finite-difference migration: Geophysics, **59**, 1882–1893.

Sacchi, M. D. and T. J. Ulrych, 1995, High-resolution velocity gathers and offset space reconstruction: Geophysics, **60**, 1169–1177.

Scales, J. A. and M. L. Smith, 1994, Introductory geophysical inverse theory: Samizdat Press.

Stoffa, P. L., J. T. Fokkema, R. M. de Luna Freire, and W. P. Kessinger, 1990, Split-step fourier migration: Geophysics, **55**, 410–421.

Tang, Y., 2006, Least-squares migration of incomplete data sets with regularization in the subsurface-offset domain: **SEP-125**, 159–175.

———, 2008a, Modeling, migration and inversion in the generalized source and receiver domain: **SEP-136**, 97–112.

———, 2008b, Wave-equation Hessian by phase encoding: **SEP-134**, 1–24.

Tang, Y. and B. Biondi, 2009, Least-squares migration/inversion of blended data: **SEP-138**.

Ulrych, T. J., M. D. Sacchi, and A. Woodbury, 2001, A Bayes tour of inversion: A tutorial: Geophysics, **66**, 55–69.

Valenciano, A., 2008, Imaging by Wave-equation Inversion: PhD thesis, Stanford University.

Wang, J. and M. D. Sacchi, 2007, High-resolution wave-equation amplitude-variation-with-ray-parameter (AVP) imaging with sparseness constraints: Geophysics, **72**, S11–S18.

Yu, J., J. Hu, G. T. Schuster, and R. Estill, 2006, Prestack migration deconvolution: Geophysics, **71**, no.2, S53–S62.

Zhang, Y., G. Zhang, and N. Bleistein, 2005, Theory of true-amplitude one-way wave equations and true-amplitude common-shot migration: Geophysics, **70**, E1–E10.

# Target-oriented joint inversion of incomplete time-lapse seismic data sets

*Gboyega Ayeni and Biondo Biondi*

## ABSTRACT

We propose a joint inversion method, based on linear least-squares wave-equation inversion, for imaging incomplete time-lapse seismic data sets. Such data sets can arise from presence of production facilities or intentional *sparse sampling.* These data sets generate undesirable artifacts that degrade the quality of time-lapse seismic images, making them unreliable indicators of production-related changes in reservoir properties. To solve this problem, we pose time-lapse imaging as a joint linear inverse problem that utilizes concatenations of target-oriented approximations to the least-squares imaging Hessian. Using a subset of the 2D Marmousi model, we show that the proposed method gives reliable time-lapse seismic images from incomplete seismic data sets.

## INTRODUCTION

There is a wide range of published work on the most important aspects of time-lapse seismic imaging. Some of these works include studies of seismic properties of reservoir fluids (Batzle and Wang, 1992), processing and practical applications (Rickett and Lumley, 2001; Calvert, 2005), and successful case studies (Lefeuvre et al., 2003; Whitcombe et al., 2004; Zou et al., 2006). Because of many successful applications, time-lapse seismic imaging is now an integral part of many reservoir management projects.

A recurring problem in many field time-lapse seismic applications is the presence (and sometimes changing locations) of production and development facilities. Such facilities prevent perfect geometry repetition for different surveys and can pose a major challenge when they are directly located above producing reservoirs. In order to circumvent this problem, it is common practice to *undershoot* the facilities using two or more boats. However, the *undershoot* approach does not work in all situations, mainly because the shot/receiver offset distributions cannot be perfectly matched.

Incomplete time-lapse seismic data sets also arise from intentional subsampling of seismic data sets. Such regularly (Calvert and Wills, 2003; Smit et al., 2006) or randomly (Arogunmati and Harris, 2007) subsampled data sets reduce the overall acquisition cost requirement for multiple seismic surveys. Successful field application of regularly sub-sampled time-lapse data sets has been demonstrated by previous authors (Calvert and Wills, 2003; Smit et al., 2006). Although regularly sampled data sets removes unnecessary redundancy in time-lapse data sets and can sufficiently sample low frequency spatial changes in reservoir properties, high frequency changes will likely not be captured. Acquiring seismic data sets randomly can ensure that all parts of the evolving reservoir are sampled, but with different densities/folds for any given survey. Randomly sampled data sets can be interpolated and then processed as full-volume data sets (Arogunmati and Harris, 2007), or they can be directly used to reconstruct the reservoir using compressive sampling (Candes and Romberg, 2007; Candes and Wakin, 2008) principles.

We propose a joint inversion method, based on an iterative least-squares inversion of the linearized wave-equation, for direct imaging of randomly sparse/incomplete time-lapse seismic data sets. The method utilizes a system of non-stationary filters derived from an explicitly computed target-oriented approximation (Valenciano, 2008) to the linear least-squares wave-equation Hessian. A joint inversion scheme enables incorporation of structural constraints (e.g., reservoir location and geometry) and temporal constraints (e.g., smooth temporal changes) in time-lapse image estimation. The proposed method, regularized joint inversion of multiple images (RJMI), and related methods have been applied to other time-lapse seismic imaging problems (Ajo-Franklin et al., 2005; Ayeni and Biondi, 2008; Ayeni et al., 2009).

We assume that the background baseline velocity model is known and that it changes slowly between surveys. Large velocity changes and geomechanical shifts can be handled by including an event alignment step prior to or during inversion. Integration of geomechanical shifts into the joint inversion formulation is ongoing and will be discussed elsewhere. A solution of the joint inversion problem using a robust (reweighted least-squares) L1-framework is also ongoing.

In this paper, using matrix-vector notations, we first review linear wave-equation modeling, iterative least-squares migration/inversion, and the RJMI method. Then, using a subset of the 2D Marmousi model (Versteeg, 1994), we show that RJMI gives good quality time-lapse images from incomplete seismic data sets.

## Least-squares inversion of time-lapse seismic data sets

Within limits of the Born approximation of the linearized acoustic wave equation, synthetic seismic data set $\mathbf{d}$ is obtained by the action of a modeling operator $\mathbf{L}$ on the earth reflectivity $\mathbf{m}$:

$$\mathbf{d} = \mathbf{L}\mathbf{m}. \tag{1}$$

Given two data sets (baseline and monitor), acquired over an evolving earth model at times $\mathbf{0}$ and $\mathbf{1}$ respectively, we can write

$$\begin{aligned} \mathbf{d}_0 &= \mathbf{L}_0\mathbf{m}_0, \\ \mathbf{d}_1 &= \mathbf{L}_1\mathbf{m}_1, \end{aligned} \tag{2}$$

where $\mathbf{m}_0$ and $\mathbf{m}_1$ are the baseline and monitor reflectivities, and $\mathbf{d}_0$ and $\mathbf{d}_1$ are the data sets modeled by $\mathbf{L}_0$ and $\mathbf{L}_1$.

Applying the adjoint operators $\bar{\mathbf{L}}_0^T$ and $\bar{\mathbf{L}}_1^T$ to $\mathbf{d}_0$ and $\mathbf{d}_1$ respectively, we obtain the migrated baseline $\tilde{\mathbf{m}}_0$ and monitor $\tilde{\mathbf{m}}_1$ images:

$$\begin{aligned} \tilde{\mathbf{m}}_0 &= \bar{\mathbf{L}}_0^T\mathbf{d}_0, \\ \tilde{\mathbf{m}}_1 &= \bar{\mathbf{L}}_1^T\mathbf{d}_1, \end{aligned} \tag{3}$$

where $\bar{\mathbf{L}}_i^T$ denotes conjugate transpose of $\mathbf{L}_i$. The *raw* time-lapse image $\Delta\tilde{\mathbf{m}}$ is the difference between the migrated images:

$$\Delta\tilde{\mathbf{m}} = \tilde{\mathbf{m}}_1 - \tilde{\mathbf{m}}_0. \tag{4}$$

Because incomplete seismic data sets leads to high non-repeatability, $\tilde{\mathbf{m}}_\mathbf{0}$ and $\tilde{\mathbf{m}}_\mathbf{1}$ must be *cross-equalized* before $\Delta\tilde{\mathbf{m}}$ is computed. The high level of non-repeatability makes it difficult to adapt existing cross-equalization methods (Rickett and Lumley, 2001; Calvert, 2005; Hall, 2006) to randomly sampled time-lapse seismic data sets. The RJMI method takes the data acquisition geometry and sampling into account and hence can correct for the non-repeatability of the data sets.

We define two quadratic cost functions for the modeling experiments (equation 2):

$$\begin{aligned} S(\mathbf{m_0}) &= \|\mathbf{L}_0\mathbf{m}_0 - \mathbf{d}_0\|_2^2, \\ S(\mathbf{m_1}) &= \|\mathbf{L}_1\mathbf{m}_1 - \mathbf{d}_1\|_2^2, \end{aligned} \tag{5}$$

which, when minimized, give the least-squares solutions $\hat{\mathbf{m}}_\mathbf{0}$ and $\hat{\mathbf{m}}_\mathbf{1}$, where

$$\begin{aligned} \hat{\mathbf{m}}_0 &= (\bar{\mathbf{L}}_0^T\mathbf{L}_0)^\dagger\bar{\mathbf{L}}_0^T\mathbf{d}_0, \\ \hat{\mathbf{m}}_1 &= (\bar{\mathbf{L}}_1^T\mathbf{L}_1)^\dagger\bar{\mathbf{L}}_1^T\mathbf{d}_1, \end{aligned} \tag{6}$$

and $(\cdot)^\dagger$ denotes approximate inverse.

Because seismic inversion is ill-posed, model regularization is often required to ensure stability and convergence to a geologically consistent solution. For many seismic monitoring objectives, the known geology and reservoir architecture provide useful regularization information. Including baseline and monitor regularization operators ($\mathbf{R}_0$ and $\mathbf{R}_1$ respectively) in the cost functions gives

$$\begin{aligned} S(\mathbf{m_0}) &= \|\mathbf{L}_0\mathbf{m}_0 - \mathbf{d}_0\|_2^2 + \epsilon_0^2\|\mathbf{R}_0\mathbf{m}_0\|^2, \\ S(\mathbf{m_1}) &= \|\mathbf{L}_1\mathbf{m}_1 - \mathbf{d}_1\|_2^2 + \epsilon_1^2\|\mathbf{R}_1\mathbf{m}_1\|^2, \end{aligned} \tag{7}$$

which have the solutions

$$\begin{aligned} \hat{\mathbf{m}}_0 &= (\bar{\mathbf{L}}_0^T\mathbf{L}_0 + \epsilon_0^2\mathbf{R}_0^T\mathbf{R}_0)^\dagger\bar{\mathbf{L}}_0^T\mathbf{d}_0, \\ \hat{\mathbf{m}}_1 &= (\bar{\mathbf{L}}_1^T\mathbf{L}_1 + \epsilon_1^2\mathbf{R}_1^T\mathbf{R}_1)^\dagger\bar{\mathbf{L}}_1^T\mathbf{d}_1. \end{aligned} \tag{8}$$

where $\epsilon_i$ is a regularization parameter that determines the strength of the regularization relative to the data fitting goal. Although there is a wide range of suggested methods for selecting $\epsilon_i$, in most practical applications, the final choice of the parameter is subjective. Unless otherwise stated, we use a fixed, heuristically determined, data-dependent regularization parameter given by

$$\epsilon_i = \frac{max|\mathbf{d}_i|}{50}. \tag{9}$$

Estimating $\hat{\mathbf{m}}_\mathbf{0}$ or $\hat{\mathbf{m}}_\mathbf{1}$ by minimizing equation 7 is the so-called *data-space* least-squares migration/inversion method (Clapp, 2005).

Substituting equation 3 into equation 8, and re-arranging the terms, we get

$$\begin{aligned} [\mathbf{H}_0 + \mathbf{R}_{00}]\,\hat{\mathbf{m}}_0 &= \tilde{\mathbf{m}}_0, \\ [\mathbf{H}_1 + \mathbf{R}_{11}]\,\hat{\mathbf{m}}_1 &= \tilde{\mathbf{m}}_1, \end{aligned} \tag{10}$$

where $\mathbf{H}_i = \bar{\mathbf{L}}_i^T\mathbf{L}_i$ is the Hessian, and $\mathbf{R}_{ii} = \epsilon_i^2\mathbf{R}_i^T\mathbf{R}_i$ is the regularization term. Equation 10 can be solved using iterative inverse filtering leading to the so-called *model-space* least-squares migration/inversion method (Valenciano, 2008). We summarize linearized (Born) wave-equation data modeling and the least-squares Hessian derivation in Appendix *A*.

Throughout this paper, our discussion of the Hessian refers to its target-oriented approximation defined in equation A-5.

An inverted time-lapse image, $\Delta\hat{\mathbf{m}}$, can be obtained as the difference between the two images, $\hat{\mathbf{m}}_0$ and $\hat{\mathbf{m}}_1$:

$$\Delta\hat{\mathbf{m}} = \hat{\mathbf{m}}_1 - \hat{\mathbf{m}}_0. \tag{11}$$

In this paper, we refer to the method of computing the time-lapse image using equation 11 as *separate inversion.*

## Joint inversion of multiple images

In order to solve a single joint inversion problem in which the baseline and monitor images are simultaneously estimated, we combine the two expressions in equation 2 to get

$$\left[\begin{array}{c} \mathbf{d}_0 \\ \mathbf{d}_1 \end{array}\right] = \left[\begin{array}{cc} \mathbf{L}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_1 \end{array}\right] \left[\begin{array}{c} \mathbf{m}_0 \\ \mathbf{m}_1 \end{array}\right], \tag{12}$$

which can be solved by minimizing the cost function

$$S(\mathbf{m_0}, \mathbf{m_1}) = \left|\left|\left[\begin{array}{cc} \mathbf{L}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_1 \end{array}\right] \left[\begin{array}{c} \mathbf{m}_0 \\ \mathbf{m}_1 \end{array}\right] - \left[\begin{array}{c} \mathbf{d}_0 \\ \mathbf{d}_1 \end{array}\right]\right|\right|_2^2, \tag{13}$$

to obtain the solution

$$\left[\begin{array}{c} \hat{\mathbf{m}}_0 \\ \hat{\mathbf{m}}_1 \end{array}\right] = \left[\begin{array}{cc} \bar{\mathbf{L}}_0^T\mathbf{L}_0 & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{L}}_1^T\mathbf{L}_1 \end{array}\right]^{\dagger} \left[\begin{array}{cc} \bar{\mathbf{L}}_0^T & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{L}}_1^T \end{array}\right] \left[\begin{array}{c} \mathbf{d}_0 \\ \mathbf{d}_1 \end{array}\right], \tag{14}$$

where † is the pseudo refers to the pseudo-inverse.

The RJMI method differs from separate inversion, because it enables inclusion of both spatial regularization (as in separate inversion) and temporal regularization (e.g., Tikhonov) so that the cost function becomes

$$\begin{aligned} S(\mathbf{m_0}, \mathbf{m_1}) = & \left|\left|\left[\begin{array}{cc} \mathbf{L}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_1 \end{array}\right] \left[\begin{array}{c} \mathbf{m}_0 \\ \mathbf{m}_1 \end{array}\right] - \left[\begin{array}{c} \mathbf{d}_0 \\ \mathbf{d}_1 \end{array}\right]\right|\right|^2 + \\ & \left|\left|\left[\begin{array}{cc} \epsilon_0\mathbf{R}_0 & \mathbf{0} \\ \mathbf{0} & \epsilon_1\mathbf{R}_1 \end{array}\right] \left[\begin{array}{c} \mathbf{m}_0 \\ \mathbf{m}_1 \end{array}\right]\right|\right|^2 + \left|\left|\left[\begin{array}{cc} -\zeta_0\mathbf{\Lambda_0} & \zeta_1\mathbf{\Lambda_1} \end{array}\right] \left[\begin{array}{c} \mathbf{m}_0 \\ \mathbf{m}_1 \end{array}\right]\right|\right|^2, \end{aligned} \tag{15}$$

where $\mathbf{\Lambda_i}$ is the temporal regularization, and $\zeta_i$ is a relative temporal regularization parameter that determines the strength of the temporal constraint. Similar formulations have been applied to seismic tomography (Ajo-Franklin et al., 2005) and medical imaging problems (Zhang et al., 2005). However, for our problem, a direct minimization of equation 15 with an iterative solver is computationally expensive:

$$cost \propto 2 \times N_{surv} \times N_{iter} \times C_{mig}, \tag{16}$$

where $N_{surv}$ is the number of data sets, $N_{iter}$ is the number of iterations, and $C_{mig}$ is the cost of on migration. Although it is possible to reduce the computational cost by encoding the data sets (Ayeni et al., 2009), conventional single-record shot-profile implementation is too expensive for practical applications. Because several iterations are usually required to reach a useful solution, and because inversion is usually repeated several times to fine-tune parameters, the overall cost of this scheme makes it impractical. One advantage of the RJMI method is that modifications can be made to inversion parameters and the inversion repeated at several orders of magnitude more cheaply than iterative least-squares *data-space* migration/inversion. This cost reduction comes because the migration and modeling (demigration) operations are replaced by a single sparse-matrix convolution.

Minimizing equation 15 leads to the solutions $\hat{\mathbf{m}}_0$ and $\hat{\mathbf{m}}_1$:

$$\begin{bmatrix} \hat{\mathbf{m}}_0 \\ \hat{\mathbf{m}}_1 \end{bmatrix} = \left( \begin{bmatrix} \bar{\mathbf{L}}_0^T \mathbf{L}_0 & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{L}}_1^T \mathbf{L}_1 \end{bmatrix} + \begin{bmatrix} \mathbf{R}_0^T \mathbf{R}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_1^T \mathbf{R}_1 \end{bmatrix} + \begin{bmatrix} \mathbf{\Lambda_0'^T \Lambda_0} & -\mathbf{\Lambda_0'^T \Lambda_1} \\ -\mathbf{\Lambda_1'^T \Lambda_0} & \mathbf{\Lambda_1'^T \Lambda_1} \end{bmatrix} \right)^{\dagger} \begin{bmatrix} \tilde{\mathbf{m}}_0 \\ \tilde{\mathbf{m}}_1 \end{bmatrix}, \tag{17}$$

which can be obtained via iterative recursive filtering:

$$\left( \begin{bmatrix} \mathbf{H}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_1 \end{bmatrix} + \begin{bmatrix} \mathbf{R}_{00} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{11} \end{bmatrix} + \begin{bmatrix} \mathbf{\Lambda_{00}} & -\mathbf{\Lambda_{01}} \\ -\mathbf{\Lambda_{10}} & \mathbf{\Lambda_{11}} \end{bmatrix} \right) \begin{bmatrix} \hat{\mathbf{m}}_0 \\ \hat{\mathbf{m}}_1 \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{m}}_0 \\ \tilde{\mathbf{m}}_1 \end{bmatrix}, \tag{18}$$

where

$$\begin{array}{c} \mathbf{R_{ij}} = \epsilon_{\mathbf{i}} \mathbf{R'}_{\mathbf{i}} \epsilon_{\mathbf{j}} \mathbf{R_j} \\ \Lambda_{\mathbf{ij}} = \zeta_{\mathbf{i}} \mathbf{\Lambda'}_{\mathbf{i}} \zeta_{\mathbf{j}} \mathbf{\Lambda_j} \end{array} . \tag{19}$$

Following the same procedure, equation 18 can be directly extended to an arbitrary number of surveys (Ayeni and Biondi, 2008). Note that it is unnecessary to explicitly form the Hessian operators in equations 18 because they are composed of simple combinations of $\mathbf{H_0}$ to $\mathbf{H_N}$ for $N$ surveys. Also, $\mathbf{R_{ij}}$ and $\Lambda_{\mathbf{ij}}$ are not explicitly computed, but instead, the regularization operators $\mathbf{R_i}$ and $\mathbf{\Lambda_i}$ (and their adjoints) are applied at each inversion step. Depending on the problem size, computational domain and available a priori information, the spatial and temporal regularization operators can be applied over several dimensions (e.g., stacked-image, subsurface offset, subsurface scattering-angles, etc.). We have implemented these operators for any arbitrary number of surveys using sparse convolution operators. Unless otherwise stated, equation 19 is solved with a conjugate gradient algorithm.

## NUMERICAL EXAMPLES

We consider two incomplete synthetic time-lapse seismic examples aimed at imaging seismic amplitude changes using incomplete time-lapse seismic data sets. Both examples are based on a modified section of the Marmousi model (Figure 1) with the target reservoir located at a shallower depth than the original Marmousi reservoir. In both examples, we neglect geomechanical changes above the reservoir.

In both cases, the baseline data set consists of 111 surface shots spaced at 80 m and 551 receivers spaced at 16 m. In the first example, the monitor data sets were modeled with gaps in data created by obstructions along the survey line (Figure 2). The monitor data sets in the second example consist of randomly sampled shot and receiver axis (Figure 3). We avoid a multiple attenuation requirement by using a Born single-scattering modeling algorithm. We migrated the data sets with the oneway wave-equation, using 184 frequencies and computed the target-oriented Hessian with 72 frequencies.

We compare the results from migration, normalization with the Hessian diagonal, separate, and joint inversion for the target area in Figure 1. Normalization with the Hessian diagonal is one implementation of the so-called true-amplitude migration (Gray, 1997). In both examples, the same spatial regularization parameters were used for the separately and jointly inverted results. Where applicable, the regularization operators are Laplacian in x-direction and first-order gradient in time.

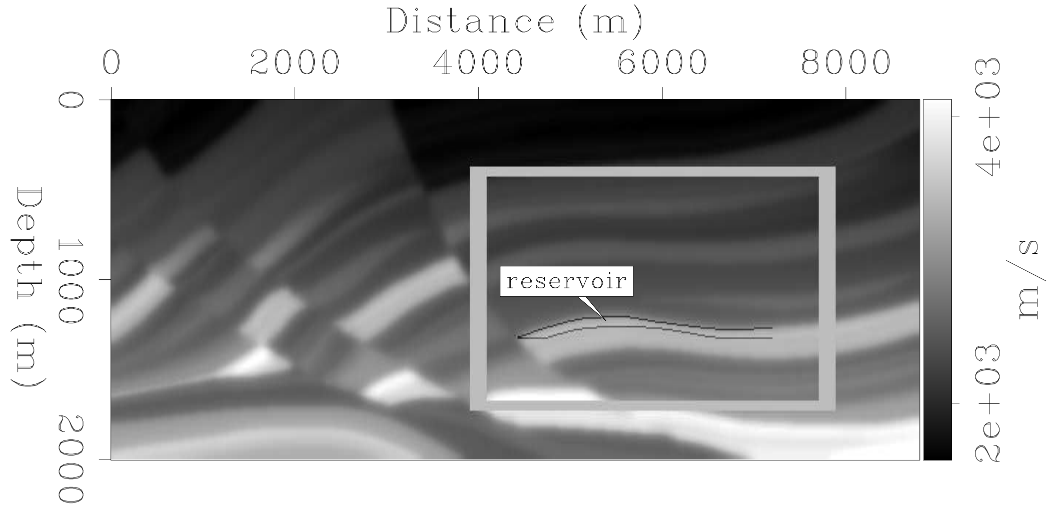
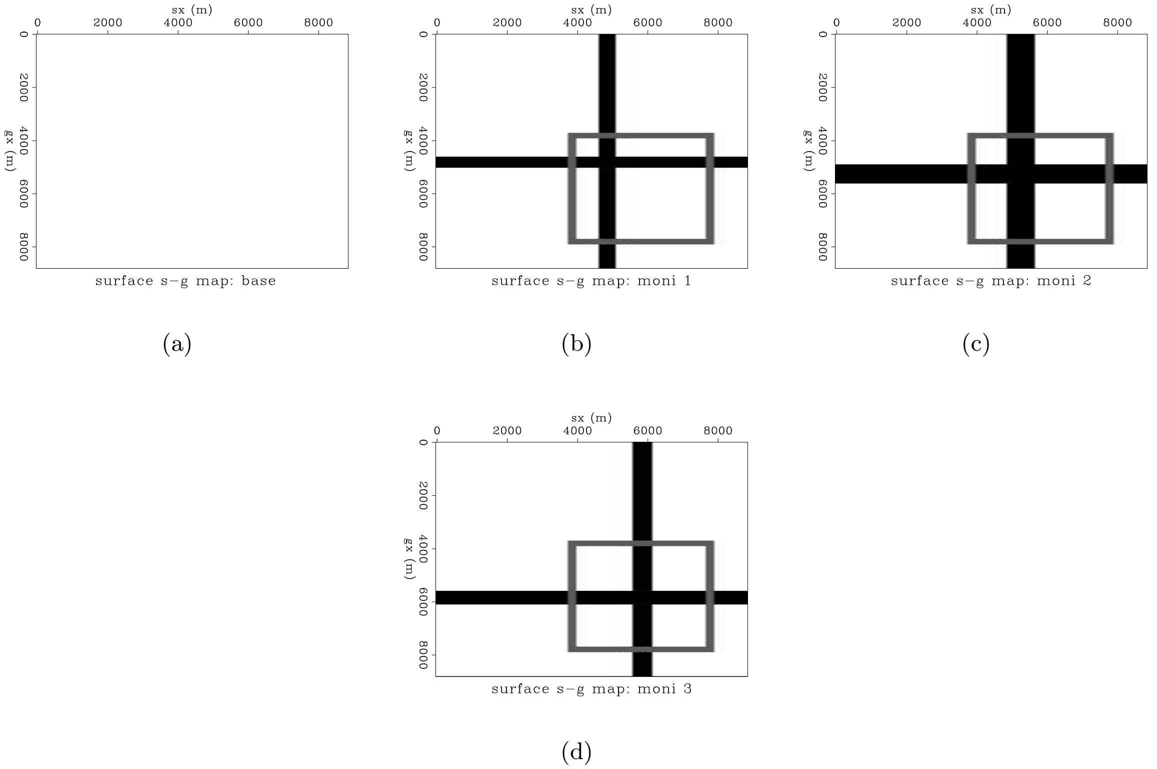


Figure 1: Modified section of the Marmousi velocity model showing the reservoir location. The gray box shows the target area.[**ER**] gayeni2/. vel-pick3

## Undershoot problem

This example demonstrates the *undershoot* problem , where obstructions caused by latter development facilities prevent complete recording of monitor data sets. Here, we consider an obstruction with changes in its location and size (Figure 2) for the three monitor data sets. This is a common scenario in seismic monitoring applications where the construction, addition and alteration of production facilities create obstructions. In each monitor survey, neither shot nor receivers were located within the undershoot area (Figure 2).

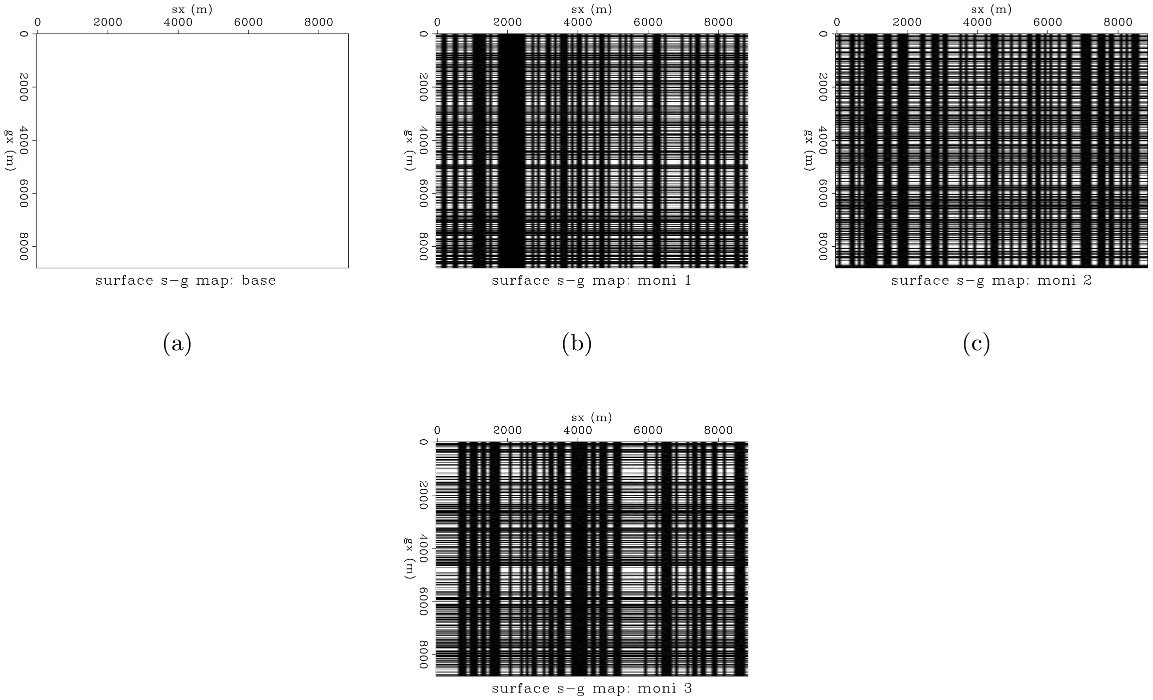
Migrated images of the target area for all four data sets are shown in Figure 4(a), and the corresponding Hessian diagonals (so-called *illumination*) in Figures 4(b). Figure 4(c) shows the cumulative time-lapse image of the target area obtained from the full data sets, and Figure 4(d) shows the illumination ratio between monitor and baseline data sets. The migrated, normalized, separately and jointly inverted time-lapse images are shown in Figure 5. Note that time-lapse images obtained from joint inversion [Figure 5(d)] contain fewer artifacts relative to those from migration, normalization and separate invertion [Figures 5(a) to 5(c)].



(a) (b) (c)

(d)

Figure 2: Surface shot-geophone coverage maps for the (a) baseline and (b)-(d) monitor surveys. White indicates locations with shot-receiver coverage whereas black gaps indicate the undershoot positions. The gray box indicates the surface location of the target area in both this Figure and also in Figure 3.[**ER**] gayeni2/. map0-h,map1-h,map2-h,map3-h



(a)

(b)

(c)

(d)

Figure 3: Surface shot-geophone coverage maps for the (a) baseline and (b)-(d) monitor surveys. White indicates locations with shot-receiver coverage whereas black indicates no coverage.[**ER**] gayeni2/. map0-r,map1-r,map2-r,map3-r

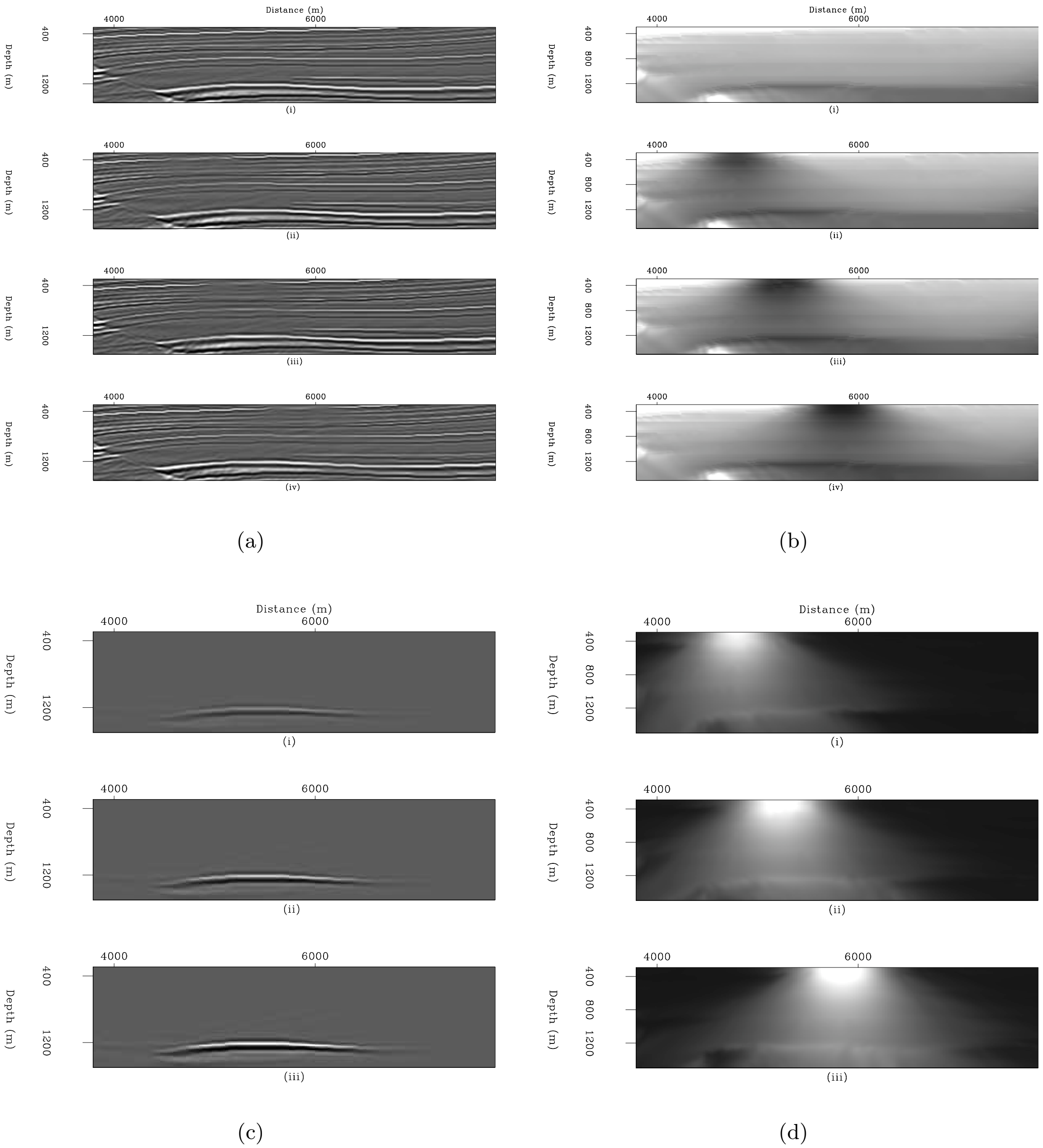
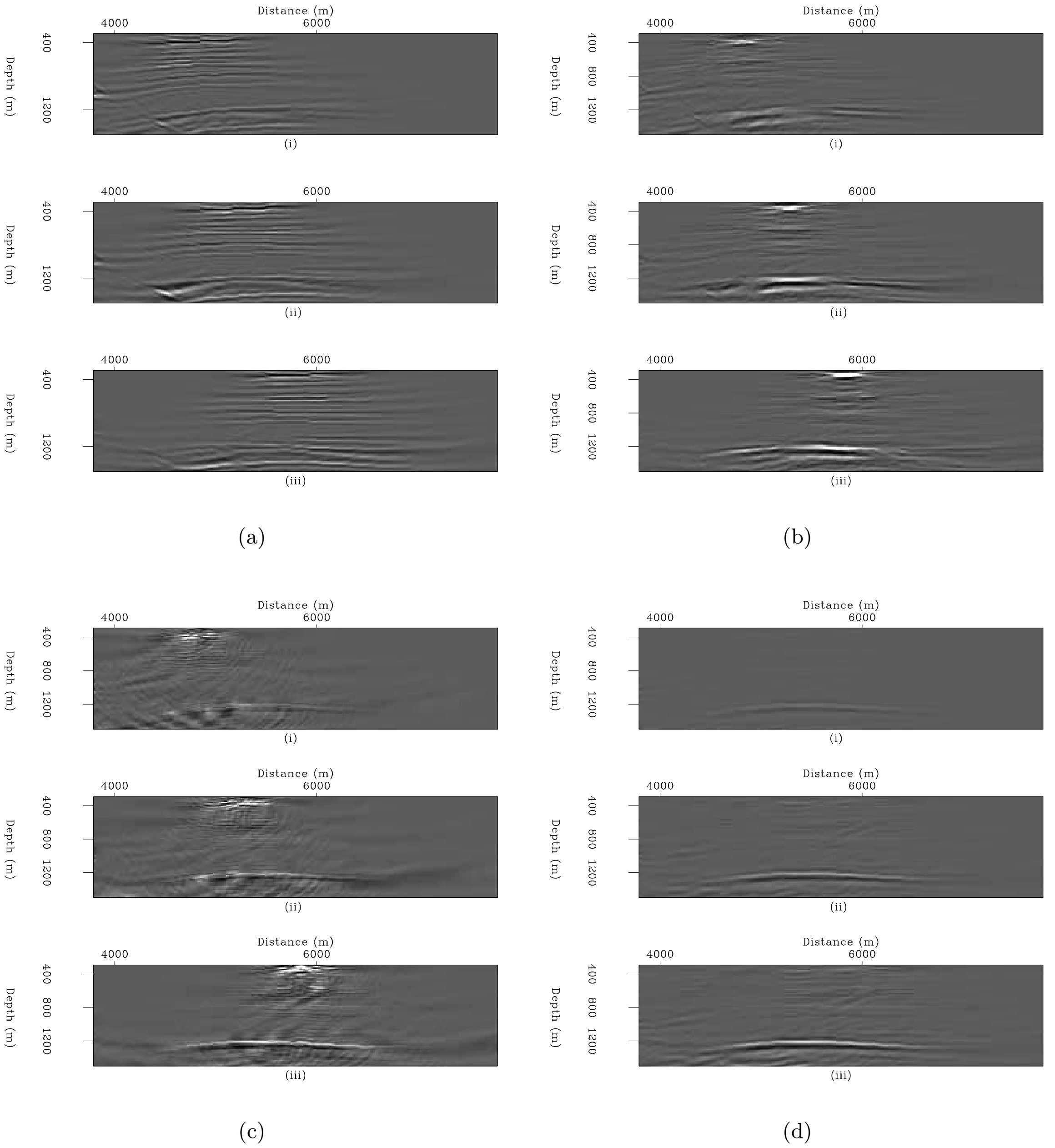


Figure 4: (a) Migrated (i) baseline and (ii)-(iv) monitor images. (b) Hessian diagonal corresponding to images in (a). (c) Cumulative time-lapse images from full baseline and monitor shot-receiver coverage [Figure 2(a)]. (d) Illumination-ratio for each monitor survey [Figures b(ii)-(iv)] relative to the baseline [Figure b(i)].[**CR**] gayeni2/. migs-1w-h,illums-hole-1w-h,migs-4d-1w,illums-hole-r-1w-h



(a) (b)

(c) (d)

Figure 5: Cumulative time-lapse images at four production stages (with increasing production from top to bottom) obtained from (a) migration, (b) Hessian-diagonal illumination correction, (c) separate inversion, and (d) RJMI. Compare these results to those from the full data sets [Figure 4(c)].[**CR**] gayeni2/. migs-4d-1w-h,invd-4d-1w-h,invs-4d-1w-h,inv2-4d-1w-h

### Sparse data problem

This example demonstrates a particular sparse time-lapse seismic monitoring problem (Arogunmati and Harris, 2007), where sparse randomly-sampled monitor data sets are acquired at a fraction of the cost of the full survey. Here, we consider a full baseline data set and three randomly sampled monitor surveys each constituting only 25 percent—at 50 per cent source and receiver sampling—of the full data set (Figure 3).

Migrated images of the target area are shown in Figure 6(a), and the corresponding Hessian diagonals in Figure 6(b). Figure 6(c) shows the cumulative time-lapse images of the target area obtained from the full data sets, and Figure 6(d) the illumination ratio between monitor and baseline data sets. The migrated, normalized, separately and jointly inverted time-lapse images are shown in Figure 7. Note that the migrated and normalized time-lapse images [Figures 7(a) and 7(b)] show no resemblance to the full data results [Figure 6(c)]. Also, note that time-lapse images obtained from joint inversion [Figure 7(d)] contain fewer artifacts than those from separate invertion [Figure 7(c)].

## DISCUSSION

From the numerical examples, we see that incomplete time-lapse seismic data-sets degrade time-lapse images [Figures 5(a) and 7(a)]. This image degradation is expected because the migration does not compensate for the resulting geometry (and hence illumination) differences in the migrated images (Figures 4(a) and 6(a)).

Normalization with the Hessian diagonal is insufficient to adequately attenuate undershoot artifacts [Figure 5(b)] and is markedly insufficient in the sparse data example [Figure 7(b)]. Although it is possible that specialized regularization methods can attenuate some of these artifacts, we suspect that most conventional cross-equalization methods will be inadequate.

Although separate inversion improves the quality of the time-lapse images relative to migration and normalization, several relatively high-amplitude artifacts persist [Figures 5(c) and 7(c)]. The high-amplitude artifacts in Figures 5(c) and 7(c) result from a mismatch of residual artifacts from the independent inversion of the data sets. Recall that the target-oriented approximation captures limited information contained in a poorly-conditioned full Hessian matrix. Although the spatial regularization improves the conditioning of the problem, residual artifacts in final results from each inversion differ and do not tend to cancel out.

In the undershoot example, joint inversion of all the data sets (using the RJMI method), improves the time-lapse image quality substantially [Figure 7(d)]. The improvement in the time-lapse images obtained via RJMI vs. separate inversion result from an inclusion of temporal constraints in the RJMI inversion. There is also significant reduction in artifacts in the RJMI sparse data results [Figure 7(d)] relative to separate inversion [Figure 7(c)]. However, in this sparse data example, there are still several residual artifacts in joint inversion results. These artifacts can be further attenuated using stronger regularization (at the cost of the data-fitting) or choosing a more robust minimization (e.g., L1-minimization by iterative re-weighting).

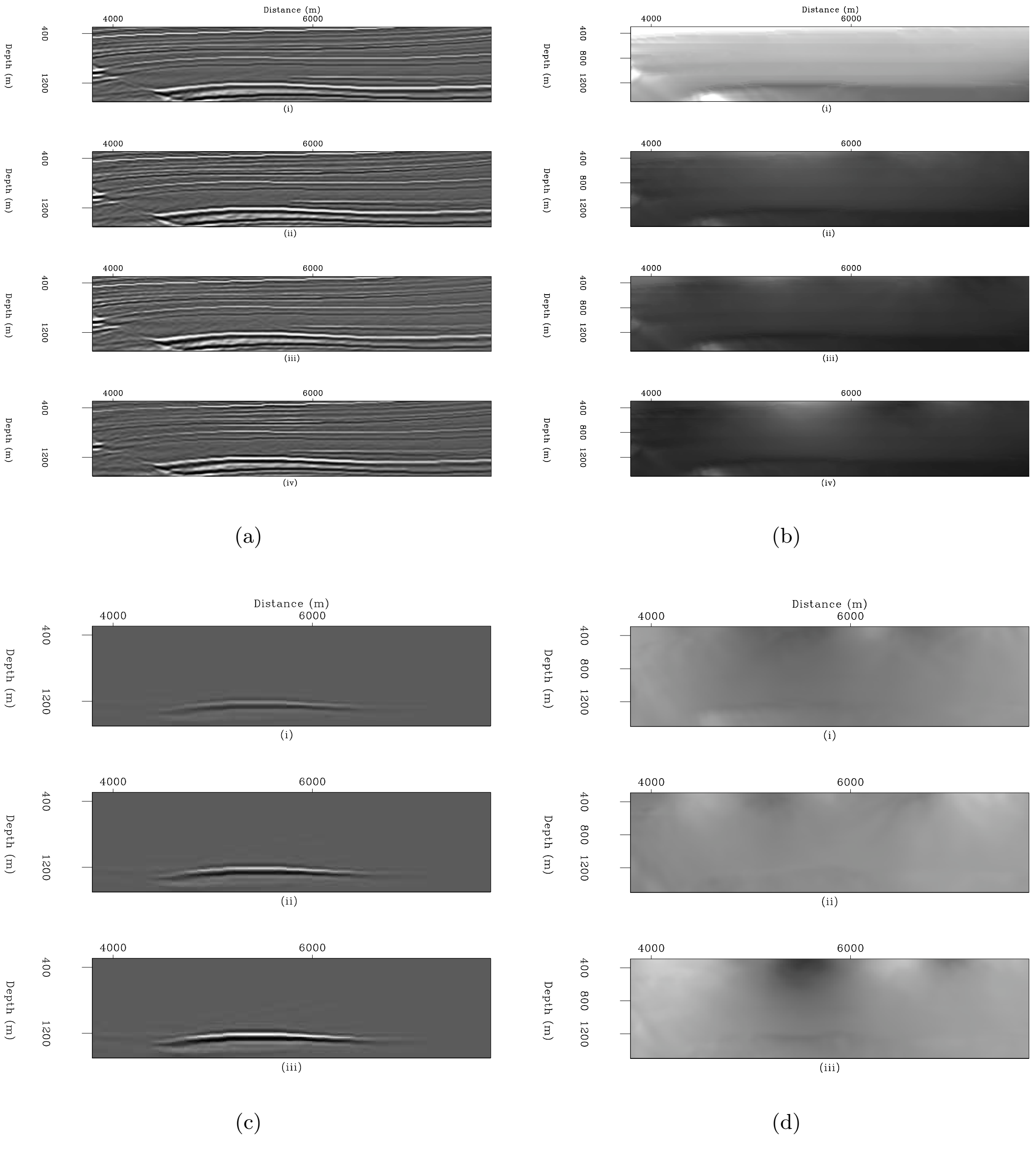
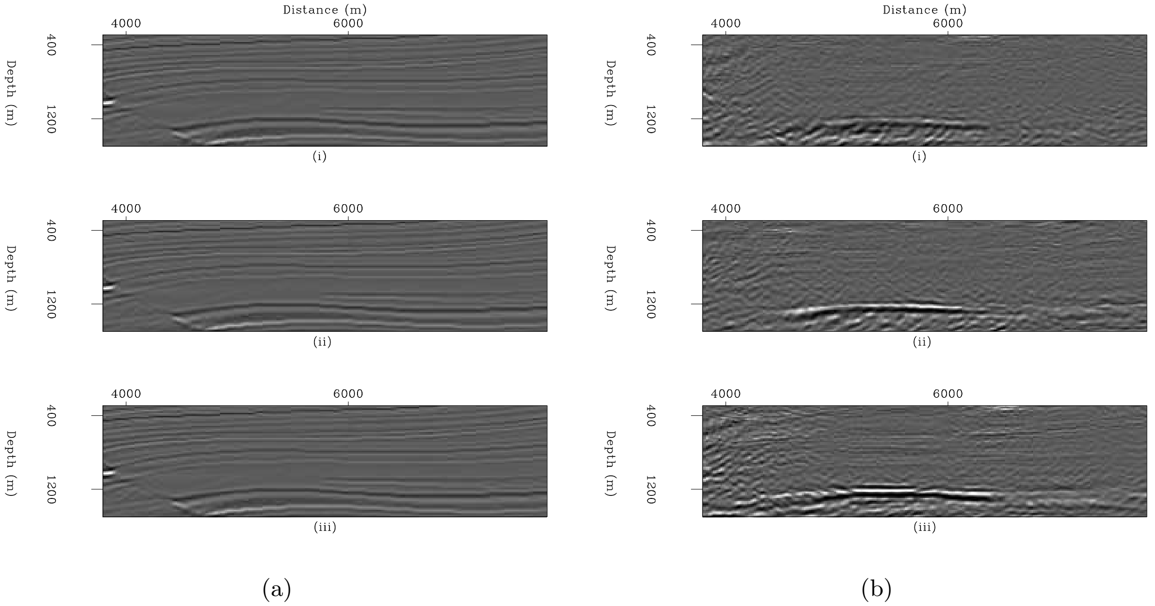


Figure 6: (a) Migrated (i) baseline and (ii)-(iv) monitor images. (b) Hessian diagonal corresponding to images in (a). (c) Cumulative time-lapse images from full baseline and monitor shot-receiver coverage [Figure 2(a)]. (d) Illumination-ratio for each monitor survey [Figures b(ii)-(iv)] relative to the baseline [Figure b(i)].[**CR**] gayeni2/. migs-1w-r,illums-hole-1w-r,migs-4d-1w2,illums-hole-r-1w-r



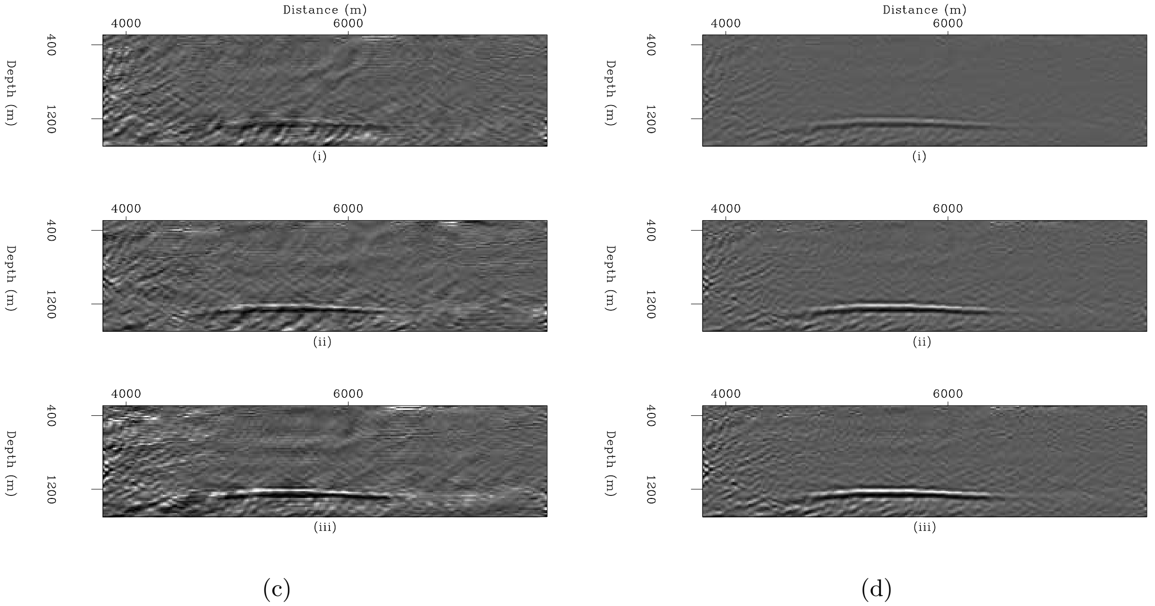


Figure 7: Cumulative time-lapse images at four production stages (with increasing production from top to bottom) obtained from (a) migration, (b) Hessian-diagonal illumination correction, (c) separate inversion, and (d) RJMI. Compare these results to those from the full data sets [Figure 6(c)].[**CR**] gayeni2/. migs-4d-1w-r,invd-4d-1w-r,invs-4d-1w-r,inv2-4d-1w-r

## CONCLUSIONS

We have demonstrated a target-oriented joint inversion method, based an iterative least-squares migration/inversion, for imaging incomplete time-lapse seismic data sets. By posing time-lapse imaging as a joint inversion problem, the RJMI method attenuates uncorrected artifacts caused by gaps in the monitor acquisition geometries. We considered an undershoot problem, where obstructions prevent perfect repetition of acquisition geometries for different surveys and a sparse time-lapse data problem, where a random fraction of the monitor data sets are recorded. In both numerical examples, we showed that joint inversion (within the RJMI framework) produces time-lapse images of the best quality relative to migration, normalization with the Hessian diagonal and separate inversion. We recognize that both the separate and joint inversion results can be improved with stronger spatial regularization, but it is arguable that such an approach will introduce too much unjustifiable bias into the inversion. Significant progress made in the field of compressive imaging (Candes and Romberg, 2007) provides a possible pathway for better image recovery from incomplete time-lapse seismic data.

## ACKNOWLEDGMENTS

We thank sponsors of the Stanford Exploration Project for their financial support.

## REFERENCES

Ajo-Franklin, J. B., J. Urban, and J. M. Harris, 2005, Temporal integration of seismic traveltime tomography: SEG Technical Program Expanded Abstracts, **24**, 2468–2471.

Arogunmati, A. and J. Harris, 2007, Data integration for spatiotemporal imaging: Proceedings, **26**, 2929–2933.

Ayeni, G. and B. Biondi, 2008, Joint wave-equation inversion of time-lapse seismic data: Stanford Exploration Project Report, **136**, 71–96.

Ayeni, G., Y. Tang, and B. Biondi, 2009, Joint preconditioned least-squares inversion of simultaneous source time-lapse seismic data sets: SEG Technical Program Expanded Abstracts, **28**, submitted.

Batzle, M. and Z. Wang, 1992, Seismic properties of pore fluids: Geophysics, **57**, 1396–1408.

Calvert, R., 2005, Insights and methods for 4D reservoir monitoring and characterization: SEG/EAGE DISC (Distinguished Instructor Lecture Course).

Calvert, R. and P. Wills, 2003, The case for 4D monitoring with sparse OBC: EAGE Expanded Abstracts, A15 – A15.

Candes, E. and J. Romberg, 2007, Sparsity and incoherence in compressive sampling: Inverse Problems, **23**, 969–985.

Candes, E. and M. Wakin, 2008, An introduction to compressive sampling: Signal Processing Magazine, IEEE, **25**, 21–30.

Clapp, M. L., 2005, Imaging under salt: illumination compensation by regularized inversion: PhD thesis, Stanford University.

Gray, S. H., 1997, True-amplitude seismic migration: A comparison of three approaches: Geophysics, **62**, 929–936.

Guitton, A., 2004, Amplitude and kinematic corrections of migrated images for nonunitary imaging operators: Geophysics, **69**, 1017–1024.

Hall, S. A., 2006, A methodology for 7D warping and deformation monitoring using time-lapse seismic data: Geophysics, **71**, O21–O31.

Lefeuvre, F., Y. Kerdraon, J. Peliganga, S. Medina, P. Charrier, R. L'Houtellier, and D. Dubucq, 2003, Improved reservoir understanding through rapid and effective 4D: Girassol field, Angola, West Africa: SEG Technical Program Expanded Abstracts, **22**, 1334–1337.

Plessix, R.-E. and W. Mulder, 2004, Frequency-domain finite-frequency amplitude-preserving migration: Geophysical Journal International, **157**, 975–985.

Rickett, J., 2003, Illumination-based normalization for wave-equation depth migration: Geophysics, **68**, 1371–1379.

Rickett, J. E. and D. E. Lumley, 2001, Cross-equalization data processing for time-lapse seismic reservoir monitoring: A case study from the Gulf of Mexico: Geophysics, **66**, 1015–1025.

Shin, C., S. Jang, and D.-J. Min, 2001, Improved amplitude preservation for prestack depth migration by inverse scattering theory: Geophysical Prospecting, **49**, 592–606.

Smit, F., M. Ligtendag, P. Wills, and R. Calvert, 2006, Toward affordable permanent seismic reservoir monitoring using the sparse OBC concept: The Leading Edge, **25**, 454–459.

Symes, W. W., 2008, Approximate linearized inversion by optimal scaling of prestack depth migration: Geophysics, **73**, R23–R35.

Valenciano, A., 2008, Imaging by Wave-Equation Inversion: PhD thesis, Stanford University.

Versteeg, R., 1994, The Marmousi experience: Velocity model determination on a synthetic complex data set: The Leading Edge, **13**, 927–936.

Whitcombe, D. N., J. M. Marsh, P. J. Clifford, M. Dyce, C. J. S. McKenzie, S. Campbell, A. J. Hill, R. S. Parr, C. Pearse, T. A. Ricketts, C. P. Slater, and O. L. Barkved, 2004, The systematic application of 4D in BP's North-West Europe operations — 5 years on: SEG Technical Program Expanded Abstracts, **23**, 2251–2254.

Zhang, Y., A. Ghodrati, and D. H. Brooks, 2005, An analytical comparison of three spatio-temporal regularization methods for dynamic linear inverse problems in a common statistical framework: Inverse Problems, **21**, 357–382.

Zou, Y., L. R. Bentley, L. R. Lines, and D. Coombe, 2006, Integration of seismic methods with reservoir simulation, Pikes Peak heavy-oil field, Saskatchewan: The Leading Edge, **25**, 764–781.

## APPENDIX A

## LINEAR LEAST-SQUARES MODELING/INVERSION

From the Born approximation of the linearized acoustic wave equation, the synthetic seismic data $d^s$ recorded by a receiver at $\mathbf{x_r}$ due to a shot at $\mathbf{x_s}$ is given by

$$d^s(\mathbf{x_s}, \mathbf{x_r}, \omega) = \omega^2 \sum_{\mathbf{x}} f_s(\omega) G(\mathbf{x_s}, \mathbf{x}, \omega) G(\mathbf{x}, \mathbf{x_r}, \omega) m(\mathbf{x}), \qquad \text{(A-1)}$$

where $\omega$ is frequency, $m(\mathbf{x})$ is *reflectivity* at image points $\mathbf{x}$, $f_s(\omega)$ is source waveform, and $G(\mathbf{x_s}, \mathbf{x}, \omega)$ and $G(\mathbf{x}, \mathbf{x_r}, \omega)$ are Green's functions from $\mathbf{x_s}$ to $\mathbf{x}$ and from $\mathbf{x}$ to $\mathbf{x_r}$ respectively.

Taking the true recorded data at $\mathbf{x_r}$ to be $d^t$, the quadratic cost function is given by

$$S(\mathbf{m}) = \|d^s(\mathbf{x_s}, \mathbf{x_r}, \omega) - d^t(\mathbf{x_s}, \mathbf{x_r}, \omega)\|_2^2. \qquad \text{(A-2)}$$

As shown by previous authors (Plessix and Mulder, 2004; Valenciano, 2008), the gradient $g(x)$ of this cost function (summed over all frequencies, sources and receivers) with respect to reflectivity is the real part of

$$g(\mathbf{x}) = \sum_{w} \omega^2 \sum_{\mathbf{x_s}} \sum_{\mathbf{x_r}} f_s(\omega) G(\mathbf{x_s}, \mathbf{x}, \omega) G(\mathbf{x}, \mathbf{x_r}, \omega) \left(d^s - d^t\right), \tag{A-3}$$

and the Hessian (second derivatives) is the real part of

$$H\left(\mathbf{x}, \mathbf{x}'\right) = \sum_{w} \omega^4 \sum_{\mathbf{x_s}} |f\,(s)|^2 G(\mathbf{x_s}, \mathbf{x}, \omega) \bar{G}(\mathbf{x_s}, \mathbf{x}', \omega) \sum_{\mathbf{x_r}} G(\mathbf{x}, \mathbf{x_r}, \omega) \bar{G}(\mathbf{x}', \mathbf{x_r}, \omega), \tag{A-4}$$

where $\mathbf{x}'$ denotes all image points and $\bar{G}$ is the complex conjugate of $G$. Plessix and Mulder (2004) and Valenciano (2008) discuss this derivation in detail.

## Target-oriented Hessian

The large computational cost of full Hessian (equation A-4) makes explicit computation impractical. Previous authors (Shin et al., 2001; Rickett, 2003; Guitton, 2004; Plessix and Mulder, 2004; Valenciano, 2008; Symes, 2008) have discussed possible approximations that reduce the computational cost or remove the need for explicit computation of the full Hessian.

Because reservoirs are limited in extent, the region of interest is usually smaller than the full image space, therefore, the Hessian can be explicitly computed for that region. For our problem, we follow the target-oriented approximation (Valenciano, 2008) to the Hessian, which for a target region $\mathbf{x_T}$ is

$$\begin{aligned} H\left(\mathbf{x_T}, \mathbf{x_{T+a_x}}\right) = \sum_{w} \omega^4 \sum_{\mathbf{x_s}} |f\,(s)|^2 G(\mathbf{x_s}, \mathbf{x_T}, \omega) \bar{G}(\mathbf{x_s}, \mathbf{x_{T+a_x}}, \omega) \\ \sum_{\mathbf{x_r}} G(\mathbf{x_T}, \mathbf{x_r}, \omega) \bar{G}(\mathbf{x_{T+a_x}}, \mathbf{x_r}, \omega), \end{aligned} \tag{A-5}$$

where $\mathbf{x_{T+a_x}}$ represents a *small* region around each point within $\mathbf{x_T}$. For any image point, $\mathbf{H}\left(\mathbf{x_T}, \mathbf{x_{T+a_x}}\right)$ represents a row of a sparse Hessian matrix $\mathbf{H}$ whose non-zero components are defined by $\mathbf{a_x}$. The term, $\mathbf{a_x}$, which can be estimated as a function of the decrease in amplitude of the Hessian diagonal, defines the *filter-size* around each image point. Valenciano (2008) discusses the target-oriented Hessian in detail and reviews the computational savings from this approximation.

*Stanford Exploration Project, Report SEP138, October 19, 2009*

# Near surface velocity estimation using early-arrival waveform inversion constrained by residual statics

*Xukai Shen*

## ABSTRACT

Early-arrival waveform inversions estimate near-surface velocity by matching the modeled and observed waveforms. Velocity estimated by such inversion usually has higher resolution if higher-frequency parts of early arrivals are used and data are surfficiently sampled in space. However, for land data, early arrivals usually contain more noise at higher frequencies, which makes higher-frequency data less reliable for waveform matching, also sampling is usually sparse due to cost considerations. Surface-consistent residual statics can correct the traveltime perturbations of deeper reflections caused by small near-surface velocity heterogeneities, assuming that rays travel vertically through the near-surface. Under this assumption, residual statics can provide information about small velocity heterogeneities. In this paper, I formulate the near-surface velocity estimation problem by constraining early-arrival waveform inversion with a modified version of receiver residual statics. The synthetic result shows that for inversion of early arrivals with relatively low frequency content, velocity estimation constrained by residual statics can provide a more detailed near-surface velocity field and more consistent reflector locations across migrated images from different shots.

## INTRODUCTION

Near-surface velocity can be important for imaging deeper reflectors. The conventional ways of estimating near-surface velocity are refraction statics (Hampson and Russell, 1984; Olson, 1984) and turning-ray tomography (White, 1989). These methods pick first breaks, and then iteratively estimate the near-surface velocity model by tracing rays through it and minimizing the difference between modeled first breaks and picked first breaks. Waveform inversion, a more sophisticated method, tries to match the waveform rather than the first-break traveltimes (Tarantola, 1984; Pratt and Hicks, 1998; Mora, 1987). By matching the waveform, more information in the data is used, resulting in higher resolution of the estimated velocity. When the higher-frequency parts of data are used, the estimated velocity usually has higher resolution (Sirgue and Pratt, 2004). Recently, people have applied this idea to estimate near-surface velocity by matching the refracted arrivals (C. Ravaut and Dell'Aversana, 2004) or early arrivals in general (J. Sheng and Schuster, 2006), while both papers use velocity estimated from refraction statics as the initial solution. Both these papers report higher resolution of estimated velocity after early-arrival waveform inversion and better migrated images using the estimated velocity. However, early arrivals in land data are usually very noisy at higher frequencies. Matching data in the presence of such noise is difficult and is likely to introduce errors in estimated velocity. Another important factor that affects the resolution of waveform inversion is receiver sampling. If receiver sampling is dense enough, then the wavefield perturbation caused by small scale velocity anomalies will be observed and can be used to invert for these small scale velocity features. Otherwise, these small perturbations are missing and result in lower resolution of estimated velocity. To obtain high-resolution velocity without using noisy high-frequency data and in the absence of densely sampled receiver data, I propose to use the receiver residual statics differences as a constraint in the early-arrival waveform inversion.

I define receiver residual-statics difference (RRSD) as the static shift between adjacent traces within the same shot. There are two ways of obtaining such measurements, one is to measure it from all shot gathers, and for each receiver location, such measured RRSDs are usually not exactly the same across different shots; the other way is to calculate it from near surface velocity using the surface-consistent concept. The surface-consistency concept is built on the assumption that rays reflected from deep reflectors travel vertically in the near surface. Conventional residual statics methods use the concept to derive surface-consistent receiver/source residual statics for each receiver/source location. The two kind of RRSDs defined here share some similarities with the conventional residual statics. The measured RRSDs from all shot gathers are not surface-consistent, however, they quantitatively show how reflection events in each shots are affected by near surface velocity anomalies. The modeled RRSDs from near surface velocities are derived from near-surface velocity, and thus are surface-consistent, but they are not so closely related to reflection events in recorded data. By forcing certain similarities between the modeled RRSDs and measured RRSDs, the vertically traveling ray assumption can be used to connect near-surface velocity and measured RRSDs analytically. Many authors (Rothman, 1985; Ronen and Claerbout, 1985) have noticed similarity between the results of conventional surface-consistent residual-statics estimations and the structures of near-surface velocity anomalies . In the case of RRSD, these similarities reflect the traveltime difference of vertical rays at adjacent receiver locations. By forcing similarities between the measured RRSDs and modeled RRSDs in the least square sense, discrete time sampling of modeled RRSDs from near surface velocity can be avoided. On the other hand, if conventional residual statics are used here, such traveltime differences between each pair of adjacent traces become discrete, which are usually not realistic.

The paper is organized as follows: I first set up the inverse problem mathematically with both waveform fitting goals and RRSD fitting goals. I then test the algorithm on a synthetic example. Last, I conclude with possible improvements and future directions.

## THEORY

The overall fitting goal can be written as

$$\left\{ \begin{array}{l} \mathbf{W_e F_{bp}} \left( \mathbf{D(m)} - \mathbf{d_{obs}} \right) \approx \mathbf{0} \\ \epsilon \left( \mathbf{T(m)} - \delta\tau_{\mathbf{obs}} \right) \approx \mathbf{0} \end{array} \right. \tag{1}$$

where $\mathbf{m}$ is the model, which consists of near-surface slowness (inverse of near surface velocity); $\mathbf{d_{obs}}$ are data, which consist of the recorded wavefield; $\mathbf{W_e}$ is a weighting function that windows out early arrivals; $\mathbf{F_{bp}}$ is the bandpass operator to selectively use the different frequency content of recorded early arrivals; $\mathbf{D}$ is the constant-density two-way acoustic wave-equation operator that generates synthetic early arrivals from source and near-surface velocity; and $\delta\tau_{\mathbf{obs}}$ is the measured RRSD from all shot and receiver locations. $\mathbf{T}$ is the operator that calculates RRSD, assuming vertically traveling rays in the near surface; and

$\epsilon$ is a number that balances the relative weighting of the two fitting goals, since the data in these equations are in different spaces and the first equation is the the major fitting goal. In the synthetic example shown here, I do not estimate the source wavelet, and source wavelet estimation will be added later. Next I will describe each fitting goal in more detail

Waveform inversion has been extensively studied, and there are many references on how to implement the algorithm (Mora, 1987; Tarantola, 1984; Pratt and Hicks, 1998). To match early arrivals, the only additional difference is the weighting of both recorded data and modeled data. The weighting acts as a window that mutes anything that is not an early arrival. In addition, to selectively use the different frequency content of recorded early arrivals, both recorded data and synthetic data will be band passed before applying the weighting function. Since the objective function of waveform inversion is more linear with respect to velocity at lower frequencies. The waveform inversion I use is a time-domain method, where all the wave propagation is done in the time domain, using an explicit finite-difference scheme with 6th order accuracy in space and 2nd order accuracy in time.

The second fitting goal in equation 1 is to minimize the misfit between the modeled RRSD and measured RRSD. Notice that for each receiver location, there is only one modeled RRSD, which is modeled from near-surface velocity assuming vertically traveling rays. On the other hand, there are many measured RRSD at each receiver location; the number of measurements equals the number of shots that were recorded at that receiver location. By having many measurements at each receiver location, the fitting goal is more robust in the presence of potentially biasing noise. The current algorithm minimizes the misfit in the least-square sense. I model the RRSD in two steps: first I calculate the traveltime of vertical rays through near-surface velocity; then I take the difference of all these modeled RRSDs along the receiver axis, which is the $x$-axis in the 2D case. Then the second fitting goal can be written more specifically as follows:

$$\epsilon\,(\mathbf{D_x I_z m} - \delta\tau_{\mathbf{obs}}) \approx \mathbf{0} \tag{2}$$

where $\mathbf{I_z}$ is the vertical integration operator, $\mathbf{m}$ is the slowness (inverse of velocity), $\mathbf{D_x}$ is the difference operator along the $x$ axis, and is dimensionless; the other symbols are the same as in equation 1.

To update the velocity, I use the nonlinear conjugate gradient method. I first calculate the gradient of equation 1, and then I calculate the step length using a method similar to that proposed by Mora (1981).

## SYNTHETIC EXAMPLE

I test the algorithm on a synthetic velocity example which is a modified version of the Amoco-Statics94 synthetic velocity model. To create the model, I first window out a small part of the whole velocity model, then retain some near-surface features and replace the deeper part of the velocity model with a constant velocity gradient plus a reflector (Figure 1a). An enlarged view of the near-surface part is shown in Figure 3a. I then compare the velocity-estimation algorithm with and without the RRSD constraint. I also compare the images produced by the two velocity-estimation methods.

The major feature of the synthetic near-surface velocity model is a weathered layer with rapidly varying thickness. Immediately below this weathered layer, there is a reflector that

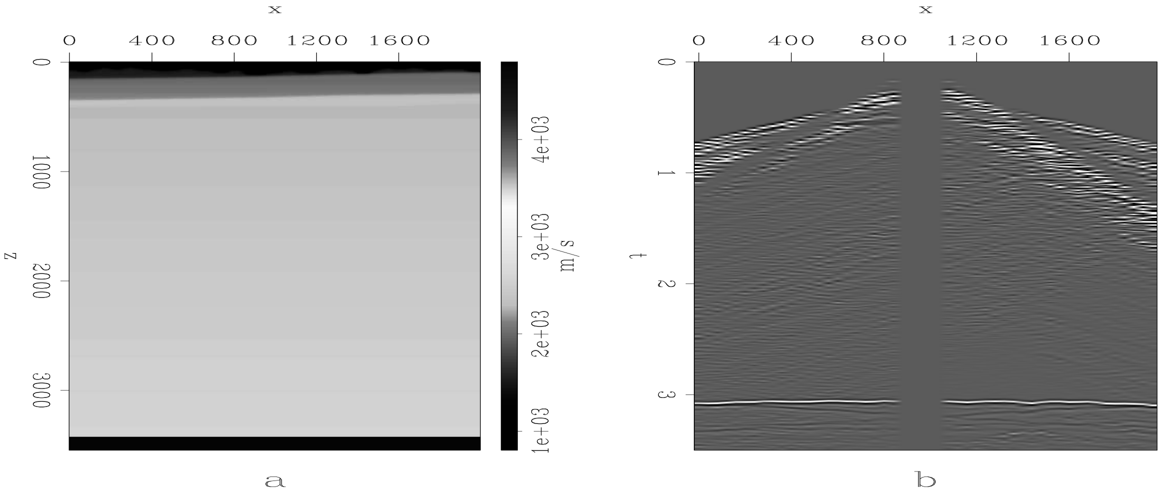


Figure 1: (a) The true velocity field with a 5 m spacing grid. (b) One typical synthetic shot record. [**CR**] xukai1/. simu

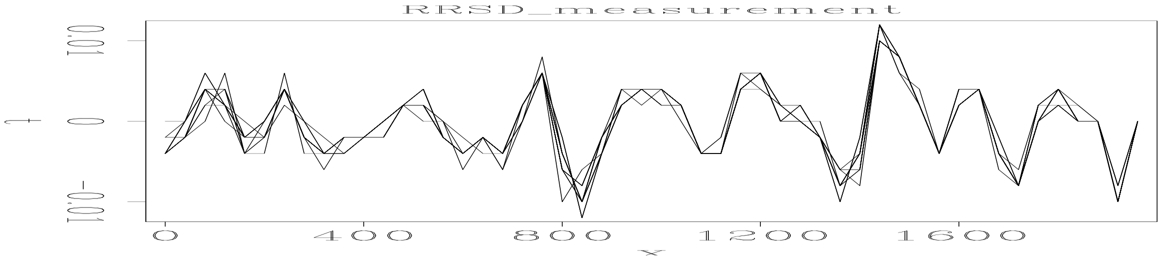


Figure 2: RRSD measured from all shot and receiver locations. [**CR**] xukai1/. RRSD

gently dips upwards towards the right. Also notice that the velocity of the weathered layer gradually decreases from left to right. The deep reflector is located at about 3.2 km depth. The size of the velocity model is 2000 m in $x$ and 3450 m in $z$ with the spacing of both $x$ and $z$ being 5 m. I generate 15 synthetic shots from $x = 120$ m to $x = 1860$ m with 120 m shot spacing, using the first derivative of the Gaussian as the source wavelet. The peak frequency of the source wavelet is 25 Hz. The receiver spacing is 40 m with nearest offset of 100 m, and the recording time sampling is 2 ms. A typical shot is shown in Figure 1b. For the measured RRSD, the time sampling is 2 ms. RRSD measurements from all 15 shots are shown in Figure 2. Although measurements at the same receiver location from different shots are not the same, they follow approximately the same trend. For near-surface velocity estimation, the $z$ and $x$ spacings are both 10 m. The starting velocity model is a smooth version of the true near-surface velocity model (Figure 3a). I smooth it in such a way that the bottom of the weathered layer loses all details, and the dipping reflector below it follows the same shape as the smoothed bottom of the weathered layer and no longer dips to the right.

### Influence of receiver sampling on velocity estimation result

As mentioned before, receiver spacing will affect the resolution of estimated velocity by early-arrival waveform inversion. Here I run the early-arrival waveform inversion without the RRSD constraints for two cases, one with receiver sampling of 10 m , the other with receiver sampling 40 m. The estimated near-surface velocity is shown in Figure 4. It can be seen that estimated velocity with receiver spacing of 10 m has higher resolution compared with the result using 40 m spacing receiver data.

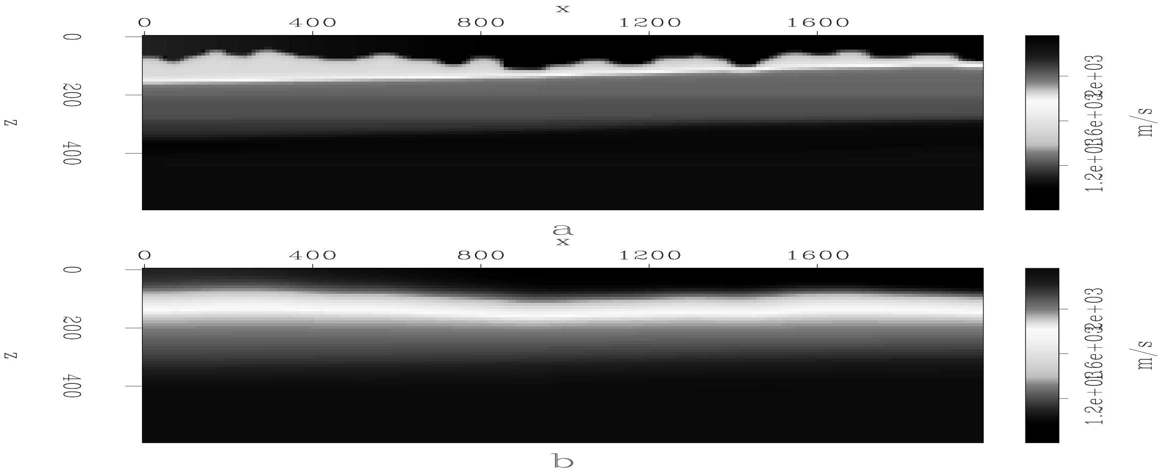


Figure 3: (a) The true near-surface velocity model with a 10 m spacing grid. (b) The starting velocity model for velocity estimation. [**CR**] xukai1/. velstart

### RRSD constrained velocity estimation result

To simulate more realistic settings, I use data recorded on 40 m receiver spacing in this synthetic example. From the starting model, I estimate the velocity by three variations of

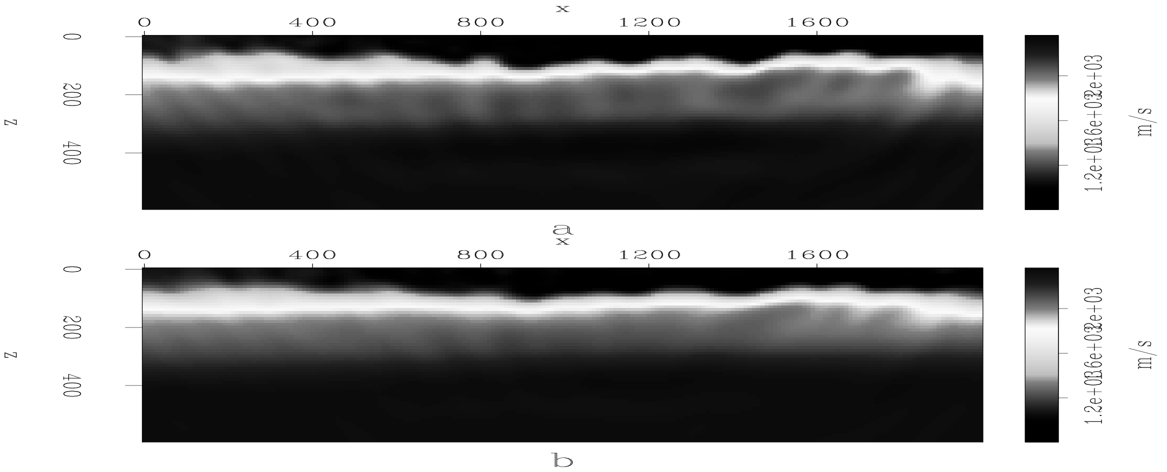


Figure 4: Early-arrival waveform inversion without RRSD using data recorded on different receiver spacing. (a) Estimated velocity with 10 m receiver spacing. (b) Estimated velocity with 40 m receiver spacing. [**CR**] xukai1/. velinvdifsamp

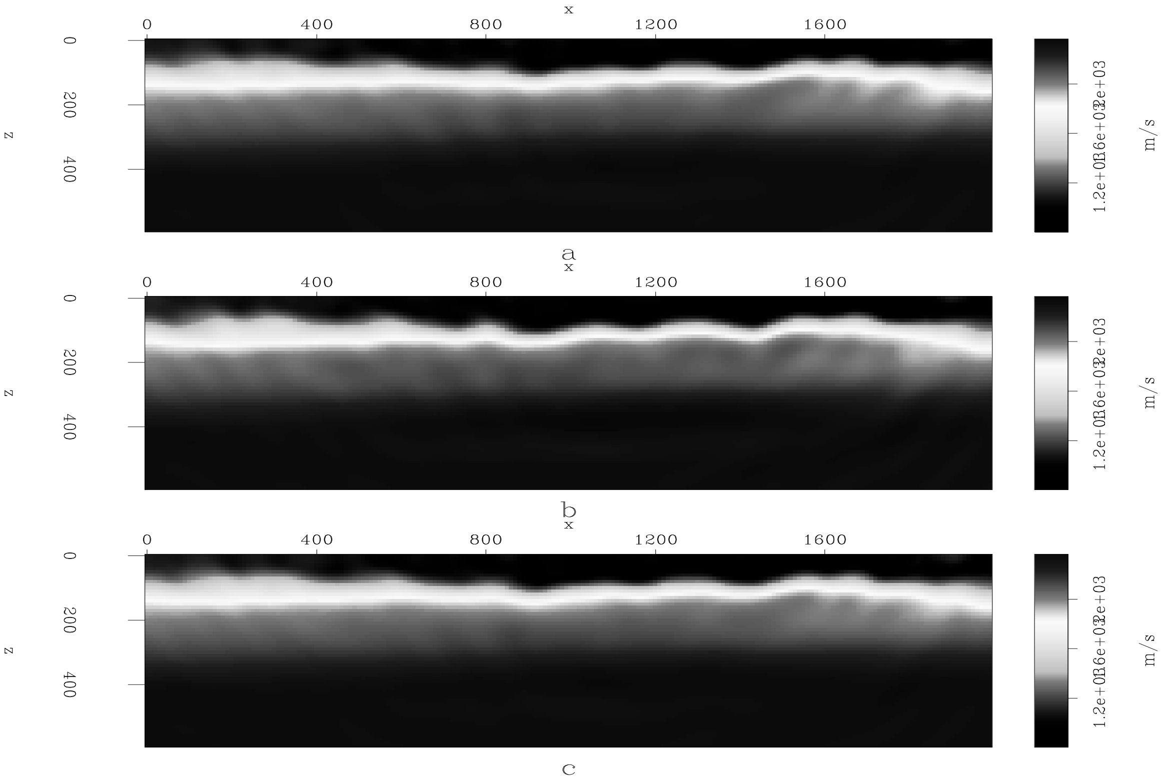


Figure 5: (a) Estimated velocity without RRSD constraints. (b) Estimated velocity with RRSD constraints. (c) Estimated velocity with starting model improved by RRSD constraints alone [**CR**] xukai1/. velinv

the alogrithm. In the first variation, I apply the algorithm without the RRSD constraints using early arrivals from 6 Hz to 10 Hz to obtain a velocity model that is closer to the true velocity model, then I use early arrivals from 9 Hz to 14 Hz to estimate the velocity also without RRSD constraints. The final estimated velocity for this case is shown in Figure 5a. For the second variation, the first step is the same as that of the first variation, in the second step, I use early arrivals from 9 Hz to 14 Hz to estimate the velocity with RRSD constraints. The final estimated velocity for this case is shown is Figure 5b. For the third variation, I first use the RRSD constraints alone to estimate a velocity field from the starting model, then I follow the same two step method in the first variation. The final result for this case is shown in 5c. It can be seen that with RRSD constraints, the estimated velocity has more sharply defined layer boundaries, and the reflector immediately below the bottom of the weathered layer is more correctly positioned, while using velocity estimated by the RRSD constraints alone for the following early-arrival waveform inversion results in an estimated velocity that is almost the same as using early-arrival waveform inversion on the original starting model. The starting residual early arrivals and the final residual early arrivals for the second pass of case one, case two and case three are shown in Figure 6, Figure 8 and Figure 7, respectively. Also, the RMS early-arrivals residual for the three variations is shown in Figure 9. Notice that both the early-arrival residual and its RMS are consistently smaller in the case of velocity estimation with RRSD constraints, while they are almost the same for the other two variations. Another way to say this is that the RRSD constraints help to speed the convergence of the velocity-estimation process, while the starting model derived by RRSD constrains alone does not result in much improvements in the final estimated velocity.

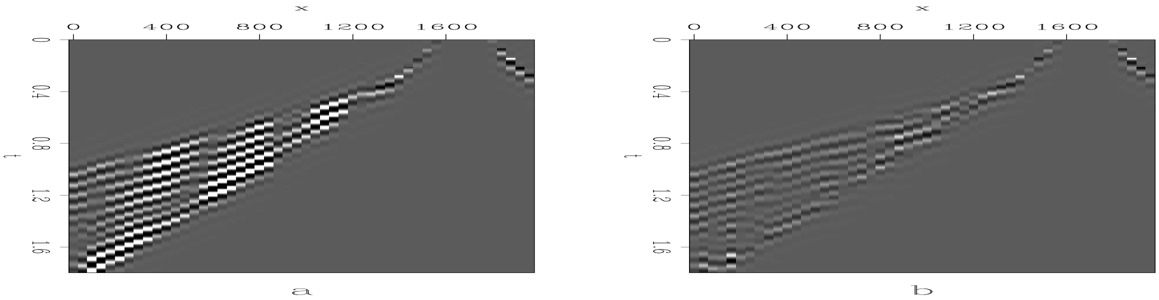


Figure 6: Early-arrival residuals for velocity estimation without RRSD; both panels are clipped to the same value. (a) Starting residuals. (b) Final residuals. [**CR**] xukai1/. resdwithout

Next, I compare the migrated images using these three velocities. All the images shown include only the part containing the deep reflector, since this is usually the area of interest. First I migrate each individual shot, and put all 15 images into a cube, to see if the reflector is consistently at the same depth (Figure 10). It can be seen that for migrated images using the velocity estimated without RRSD constraints, the reflector in different images is not consistently at the same depth. However, in the images using the velocity estimated *with* RRSD, the reflector depth is more consistent across different images. While the images using the estimated velocity with starting model improved by RRSD constraints alone is almost the same as 10b. When I stack all the images together (Figure 11), images using velocity estimated with RRSD constraints are slightly more continuous laterally since the reflector from images of different shots are more consistently positioned . Thus migrated

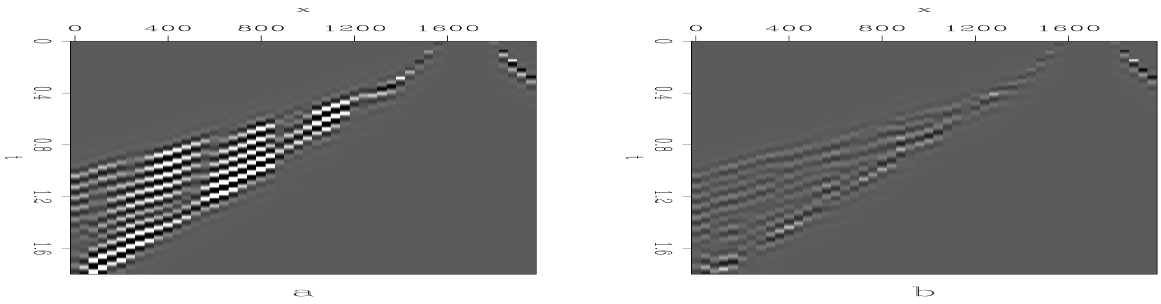


Figure 7: Early-arrival residuals for velocity estimation with RRSD; both panels are clipped to the same value. (a) Starting residuals. (b) Final residuals. [**CR**] xukai1/. resdwith

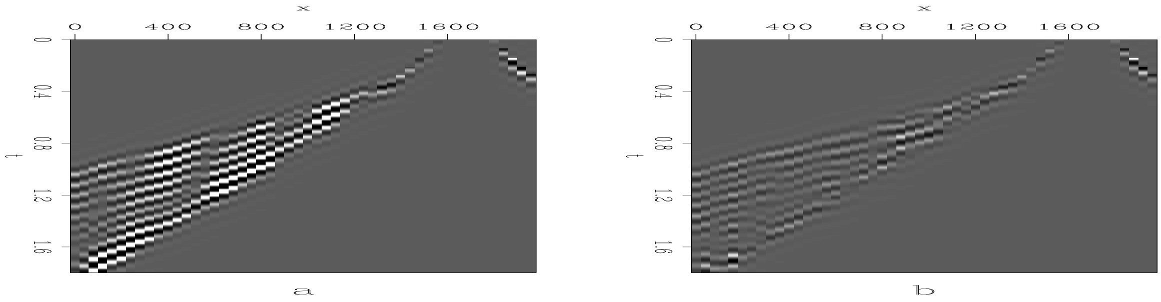


Figure 8: Early-arrival residuals for velocity estimation with starting model improved by RRSD constraints alone; both panels are clipped to the same value. (a) Starting residuals. (b) Final residuals. [**CR**] xukai1/. resdinit

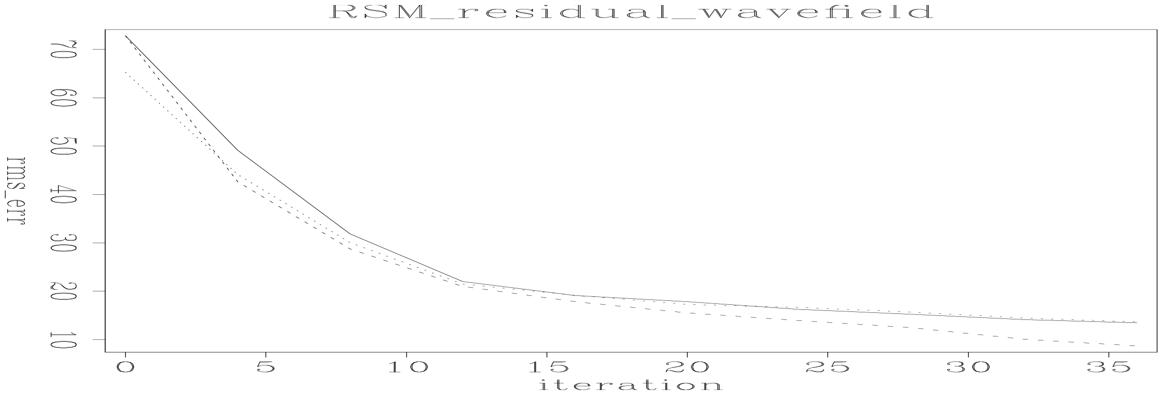


Figure 9: Change of RMS early-arrival residuals as a function of iteration. Solid curve is for estimation without RRSD, dashed curve is for estimation with RRSD, dotted curve is for estimation with starting model improved by RRSD constraints alone. [**CR**] xukai1/. RMSresd

images using estimated velocity from RRSD are more consistent in depth.

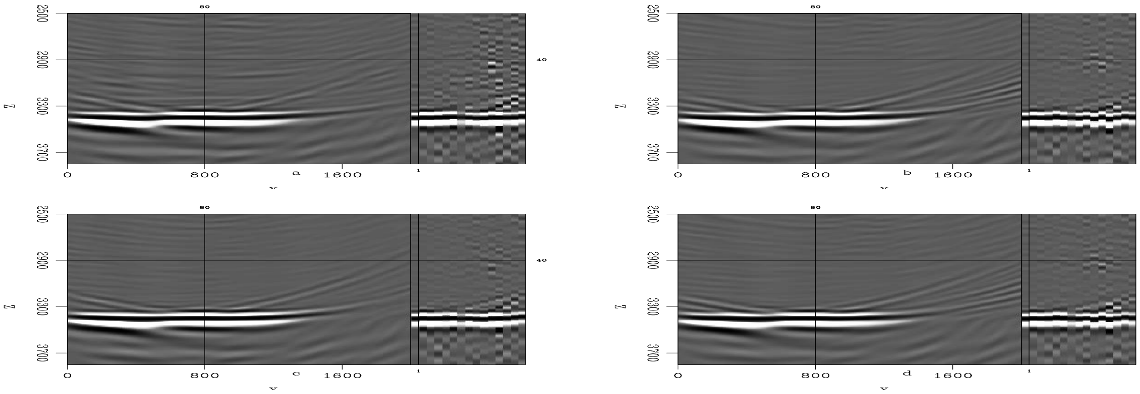


Figure 10: Migrated images of all 15 shots, showing only the deeper part with the reflector. Front panels are images of one shot. Right panels are images at the same $x$ location from different shots. Top panels are constant depth image from different shots. (a) Migrated with true near-surface velocity model. (b) Migrated with estimated velocity without RRSD constraints; notice that different shots do not put the reflector at the same depth. (c) Migrated with estimated velocity with RRSD constraints. Notice in this case the reflector depth is almost the same as that migrated with the true velocity. (d) Migrated with estimated velocity with starting model improved by RRSD constraints alone.[**CR**] xukai1/. imagcube

## CONCLUSIONS

In the synthetic example, near-surface velocity estimation with RRSD results in a velocity estimate with higher resolution and faster convergence. Also the migrated images have deep reflectors more consistently positioned at the same depth. The algorithm still needs to be tested on real land data before more general conclusions can be made. For real data, the noise will more strongly bias the RRSD measurement, in which case, using the L1 norm in the RRSD constraint might work better. Also, this algorithm works only in places where near-surface velocities are slower than the deeper velocities. In these places, near-surface velocity is slower compared with deeper velocities so the vertically traveling rays assumption is valid. Therefore, when near-surface velocity is higher (e.g. in permafrost), this method is unlikely to work. Then angle-dependent residual statics (Henley, 2009) can potentially better define the ray path of reflection events and be used to connect near-surface velocity with such residual statics.

## ACKNOWLEDGMENTS

I would like to thank Biondo Biondi, Shuki Ronen, Robert Clapp, Antoine Guitton and Jon Claerbout for a lot of helpful suggestions and discussions.

## REFERENCES

C. Ravaut, S. Operto, L. I. J. V. A. H. and P. Dell'Aversana, 2004, Multiscale imaging of complex structures from multifold wide-aperture seismic data by frequency-domain full-

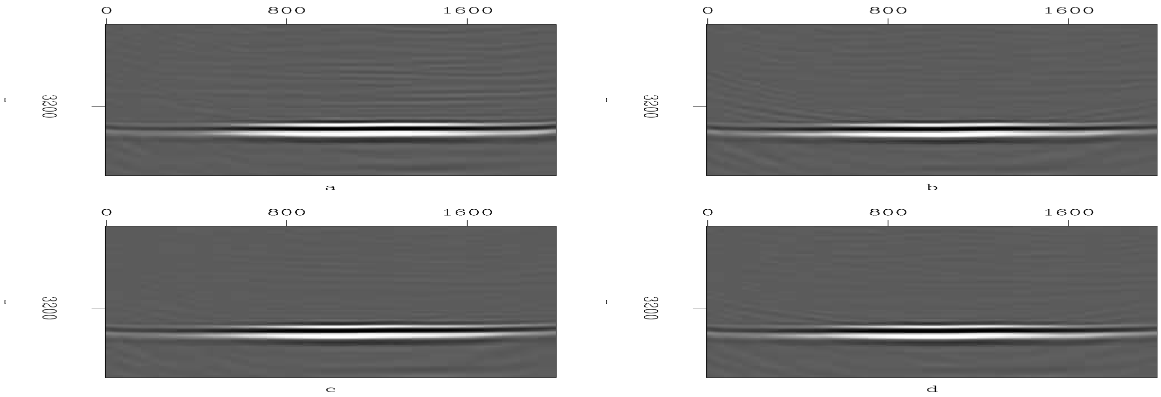


Figure 11: Stack of images from all shots, all are clipped to the same value. (a) Migrated with true velocity. (b) Migrated with estimated velocity without RRSD; notice that the reflector is of lower frequency content due to stacking of reflectors from different shots that are not at the exact same depth . (c) Migrated with estimated velocity with RRSD; notice that this image is almost identical to the image migrated with true velocity. (d) Migrated with estimated velocity using starting model improved by RRSD constraints alone.[**CR**] xukai1/. imagstk

waveform tomography: application to a thrust belt: Geophysical Journal International, **159**, 1032–1056.

Hampson, D. and B. Russell, 1984, First-break interpretation using generalized linear inversion: Journal of CSEG, **20**, 40–54.

Henley, D. C., 2009, Raypath interferometry: Statics in difficult places: The Leading Edge, **28**, 202–205.

J. Sheng, A. Leeds, M. B. and G. T. Schuster, 2006, Early arrival waveform tomography on near-surface refraction data: Geophysics, **71**, U47–U57.

Mora, P., 1987, Elastic wavefield inversion: SEP Ph.D Thesis.

Olson, K. B., 1984, A stable and flexible procedure for the inverse modelling of seismic first arrivals: Geophysical Prospecting, **37**, 455–465.

Pratt, R. G., S. C. and G. Hicks, 1998, Gauss-Newton and full Newton methods in frequency domain seismic waveform inversion: Geophysical Journal International, **133**, 341–362.

Ronen, J. and J. F. Claerbout, 1985, Surface-consistent residual statics estimation by stack-power maximization: SEP report 42.

Rothman, D. H., 1985, Automatic estimation of very large residual statics: SEP report 42.

Sirgue, L. and R. G. Pratt, 2004, Efficient waveform inversion and imaging: A strategy for selecting temporal frquencies: Geophysics, **69**, 231–248.

Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, **49**, 1259–1266.

White, D. J., 1989, Two-dimensional seismic refraction tomogaphy: Geophysical Journal International, **97**, 223–245.

# Seismic tests at Southern Ute Nation coal fire site

*Sjoerd de Ridder and Seth S. Haines*

## ABSTRACT

We conducted a near surface seismic test at the Southern Ute Nation coal fire site near Durango, CO. The goal was to characterize and image the coal fire and to help plan any future surveys. We collected data along two transects. Data from Line 1, which overlies unburned coal, shows useful frequency content above 100 Hz and a reflection that we interpret to originate at approximately 11 m depth. Data from Line 2, which crosses the burn front and many fissures, is of lower quality, with predominantly jumbled arrivals and some evidence of reflected energy at one or two shot points. It seems that neither refractions nor reflections image down to the coal layer; in part this is attributed to the presence of unexpected high-velocity layers overlying the coal. The consequence is that possible information about the coal is hidden behind the events from shallow layers. Based on these data, we suggest that further seismic work at the site is unlikely to successfully characterize the coal fire zone of interest.

## INTRODUCTION

In March 2009 personnel from Stanford University, the U.S. Geological Survey and the Southern Ute Department of Energy collected compressional (P) wave near surface seismic data along two transects at the site of a coal fire on Southern Ute Nation lands. The objective of this effort was to determine the utility of seismic methods for imaging the coal and ash layers of interest at the site, and to assist with planning any future data acquisition efforts.

The field site is generally open terrain with a gently dipping (∼10°) ground surface. The shallowest geology consists of sandstone (highly fractured and fissured in many places); the sandstone is overlain by a thin layer of soil, of about .5 m, throughout much of the site. Abundant well data show the coal layer to be about 8 m thick and to be dipping in the same direction, but a slightly higher angle (∼20°) than the ground surface. The top is approximately 5 m deep at the up-slope (up-dip) edge of the site (to the northwest) and up to 16 m deep in the down-slope part of the site. Open fissures with red-hot rock clearly visible 0.5 m below the ground surface are one clear indication of the shallow fire. Noxious gases coming from the vents are another indication.

Numerical simulations of seismic experiments at the site indicated that imaging the unburned coal could be possible with sufficiently high-frequency source energy if the impacts of fissures and layering above the coal were minimal (de Ridder and Haines, 2008). A short site visit enabled the collection of a simple data set. A basic data processing analysis was performed to assess data quality and identify arrivals. For line 1, located above unburned coal where no fissures have been mapped, data quality is good and reflections assumed to be from the sandstone layer that underlies the coal are visible. For line 2, crossing the burn front and numerous mapped fissures, data quality is reduced. The reflection might be visible for one shot point. Velocity estimation indicates that neither refractions nor reflections image as deep as the coal layer. Two main reasons for the differences between simulations and field test is the presence of high-velocities layers above the coal, and a shallow reflector.

## SEISMIC DATA

We collected P-wave seismic data along two transects, using conventional hammer-plate-seismic techniques. The recording arrays consisted of a Geometrics Geode with 24 live channels and 30 Hz vertical geophones at 3 m spacings. In order to create shot gathers with a greater number of traces at a narrower spacing, four shots were closely-spaced at each shot location. By placing these shots at a 0.75-m spacing, and then interleaving the resulting four 24-channel shot gathers, a 96 channel array was simulated, illustrated schematically in Figure 1. This approach is a variation on the "walk-away" testing that is commonly used for acquiring data at a new field site. The interleaving technique assumes that geology does not vary too rapidly along the transect. Any lateral geological variation would violate this assumption, and cause artifacts in the shot gathers.

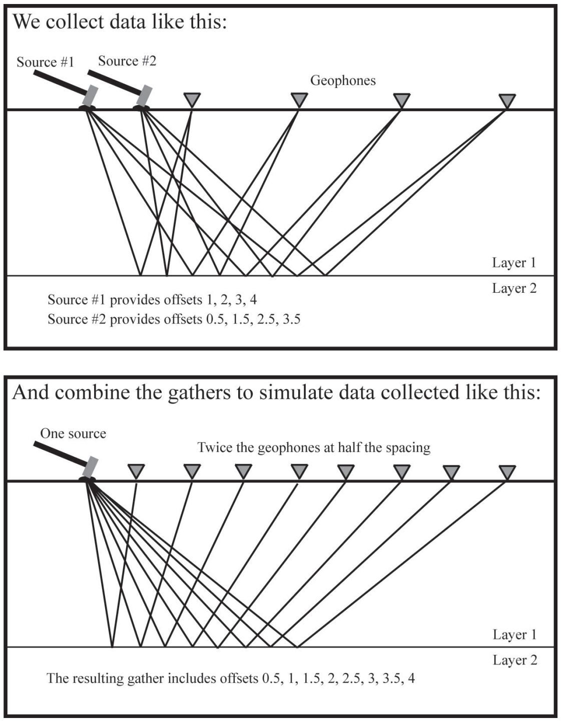

We collect data like this:
Source #1
Source #2
Geophones
Layer 1
Layer 2
Source #1 provides offsets 1, 2, 3, 4
Source #2 provides offsets 0.5, 1.5, 2.5, 3.5
And combine the gathers to simulate data collected like this:
One source
Twice the geophones at half the spacing
Layer 1
Layer 2
The resulting gather includes offsets 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4

Figure 1: Sketch of interleaving approach. With $Y$ geophones, at a spacing $X$, and using $N$ shots at a spacing of $X/N$, the interleave technique results in a shot gathers with $N\times Y$ traces and a spacing of $X/N$. Geology is assumed to be fairly constant in the lateral direction. [**NR**] sjoerd3/. reflection-interleave

Five or six shot locations were recorded along each line, rather than a shot between each pair of geophones. The shot locations are evenly distributed along the line, with one shot point off the end of each line to observe longer-offset arrivals. Each shot gather is the result of stacking approximately 5 hammer impacts, after manual data checking for trigger errors and other problems. It should be noted that for most data in these surveys, the noise reduction benefits of stacking are minimal; data quality for each individual hammer impact is good.

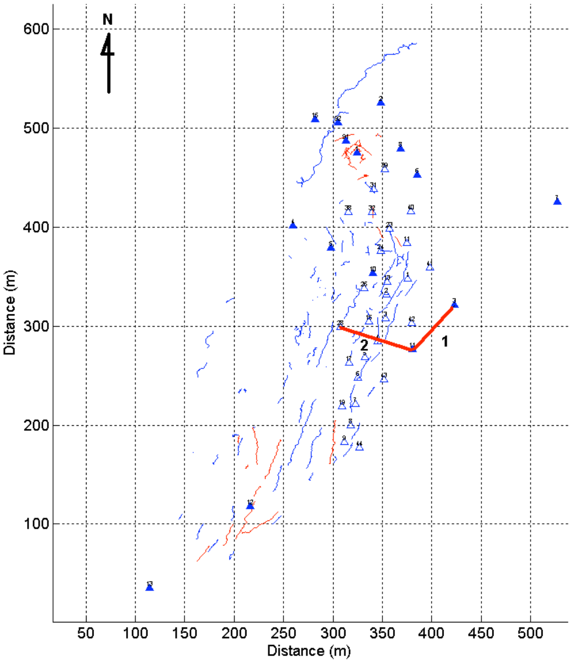


Figure 2: Site map with fissures (thin lines), wells (full triangles), thermocouples (open triangles) and seismic lines (thick lines). Courtesy of Taku Ide. [**NR**] sjoerd3/. seismic-lines

## Line 1: Unburned coal and no fissures

The first transect was recorded along a line running approximately NE-SW in an area thought to overlie purely unburned coal (based on boreholes at each end of the line). Being oriented roughly along the geologic strike, the topography along the line is generally flat and nearly horizontal. No fissures have been mapped in the area of this line. An averaged, normalized frequency spectrum of data recorded at Line 1 is shown in Figure 3. A frequency wave-number spectrum of the shot recorded at position 72 m along Line 1, at the southwestern end 10 m SW of well #11, is shown in Figure 4.

Figures 5(a)-5(f) and 6(a)-6(d) show shot gathers from shots at approximately 0 m, 18 m, 48 m, 72 m and 80 m, along Line 1. For each shot the figures contain a raw data section and a 80-500Hz bandpassed gain controlled section. In general, data quality is good, and strong energy is recorded to at least 100 Hz. In some shot gathers, a reflection can be observed. Energy propagates well to the farthest recorded offsets. For all of these line 1 data plots, NE is to the left. The interleaving technique for interpolation worked best for the shot gather in Figure 6(c). The shot in Figure 5(c) was recorded on a sandstone outcrop on the road, the data is ringy and the interleaving technique is less successful than for other shots along this line.

## Line 2: Across the burn front and many fissures

The second transect was recorded along a line running approximately NW-SE. The NW end of the line is within an area above burned coal and the SE end is in an area thought to be above unburned coal. The line crosses many mapped fissures, particularly toward the NW end of the line.

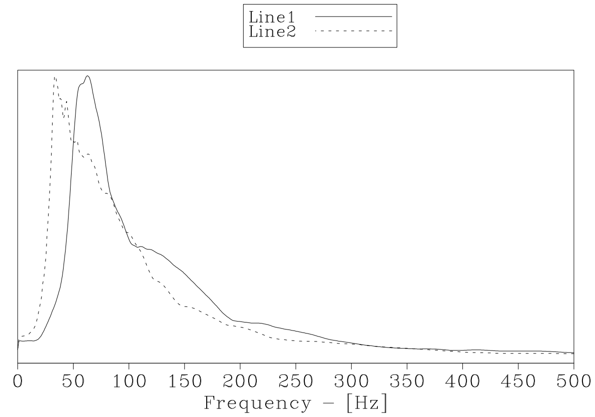


Figure 3: Averaged, normalized spectra of surveys at line 1 and line 2. [**ER**] sjoerd3/. Fspectra

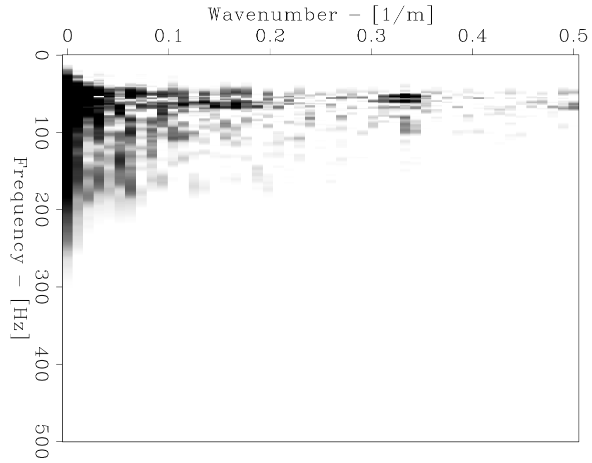
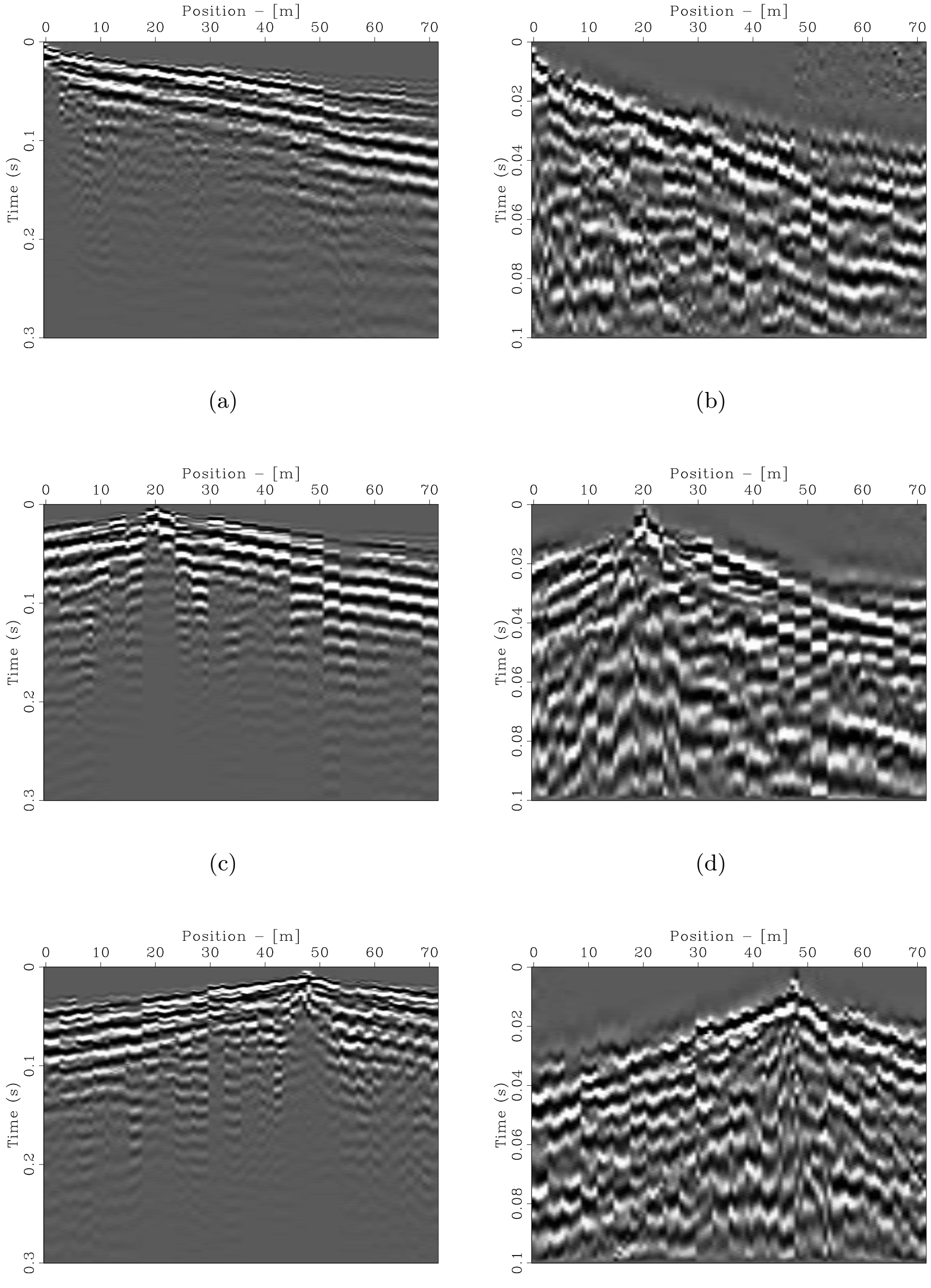


Figure 4: Frequency wavenumber spectrum of the shot recorded at Line 1, at 72 m (see also Figure 6(b) below). [**ER**] sjoerd3/. FKspectrum



(a)

(b)

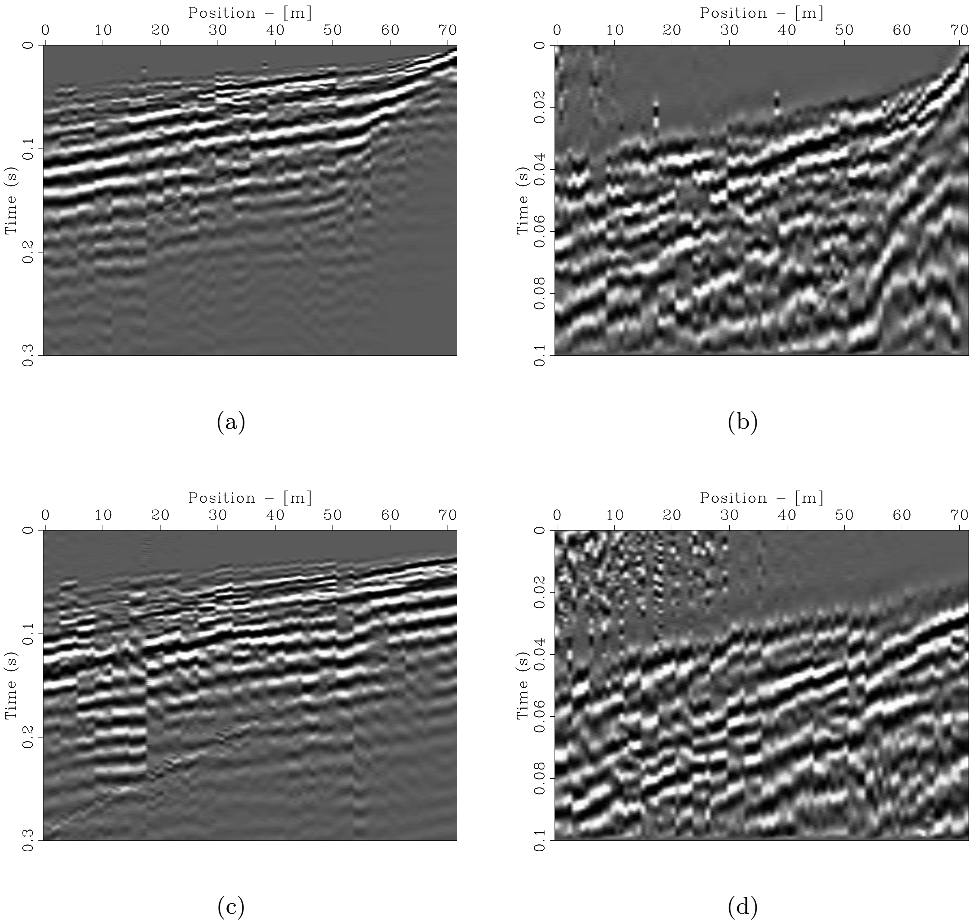
(c)

(d)

(e)

(f)

Figure 5: Line 1 shot gathers: a) Raw shot at 0 m, b) shot from (a) 80-500Hz bandpass and AGC and zoom; c) raw shot at 18 m, d) shot from (c) 80-500Hz bandpass and AGC and zoom; e) raw shot at 48 m, f) shot from (e) 80-500Hz bandpass and AGC and zoom. [**ER**] sjoerd3/. s1-inter-025,s2-inter-025,s1-inter-018,s2-inter-018,s1-inter-009,s2-inter-009



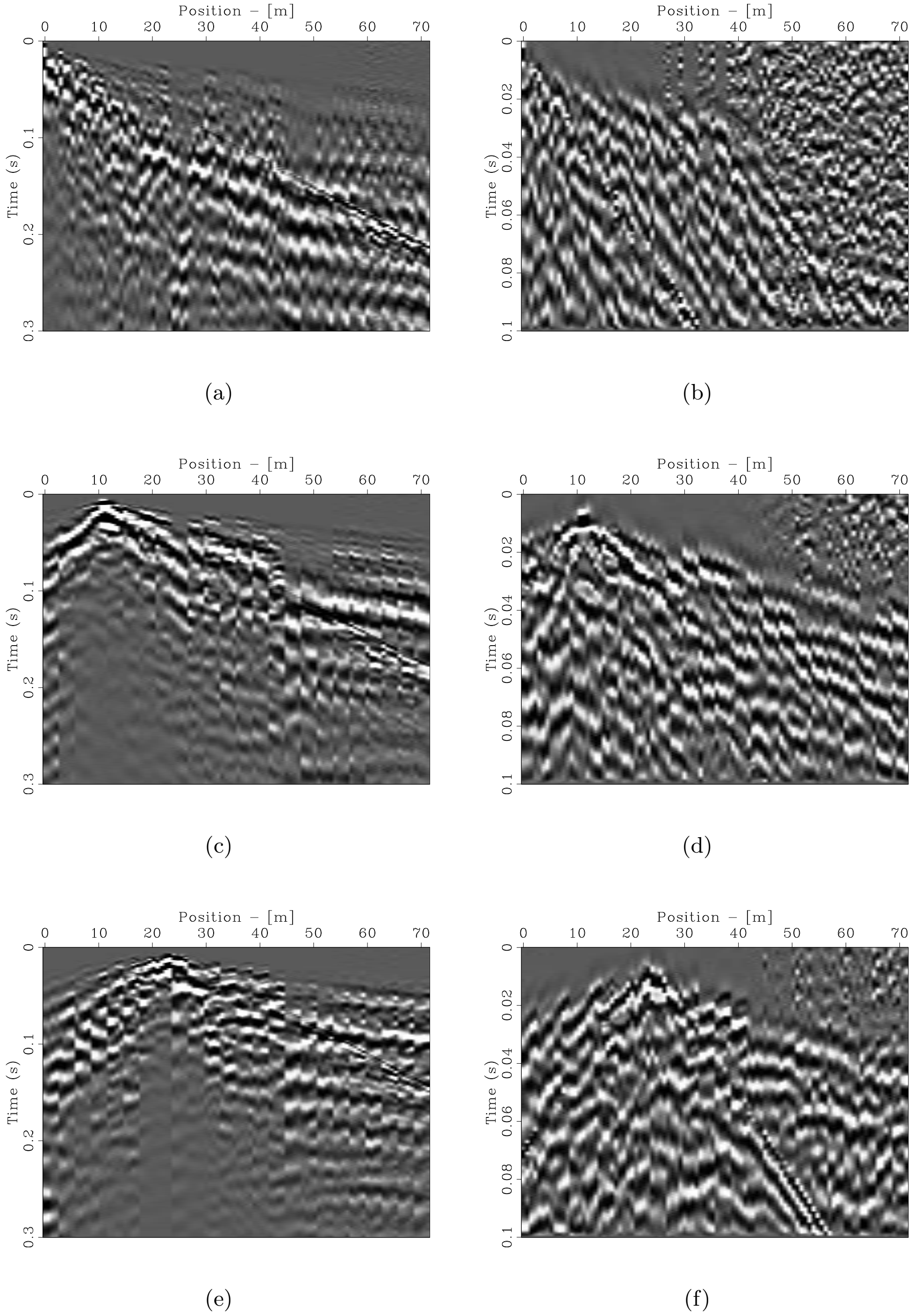
(a)

(b)

(c)

(d)

Figure 6: Line 1 shot gathers: a) Raw shot at 72 m, b) shot from (a) 80-500Hz bandpass and AGC and zoom; c) raw shot at 72 m, d) shot from (c) 80-500Hz bandpass and AGC and zoom. [**ER**] sjoerd3/. s1-inter-001,s2-inter-001,s1-inter-0m8,s2-inter-0m8



(a) (b)

(c) (d)

(e) (f)

Figure 7: Line 2 shot gathers: a) Raw shot at 0 m, b) shot from (a) 80-500Hz bandpass and AGC and zoom; c) raw shot at 12 m, d) shot from (c) 80-500Hz bandpass and AGC and zoom; e) raw shot at 24 m, f) shot from (e) 80-500Hz bandpass and AGC and zoom. [**ER**] sjoerd3/. s1-inter-225,s2-inter-225,s1-inter-221,s2-inter-221,s1-inter-217,s2-inter-217

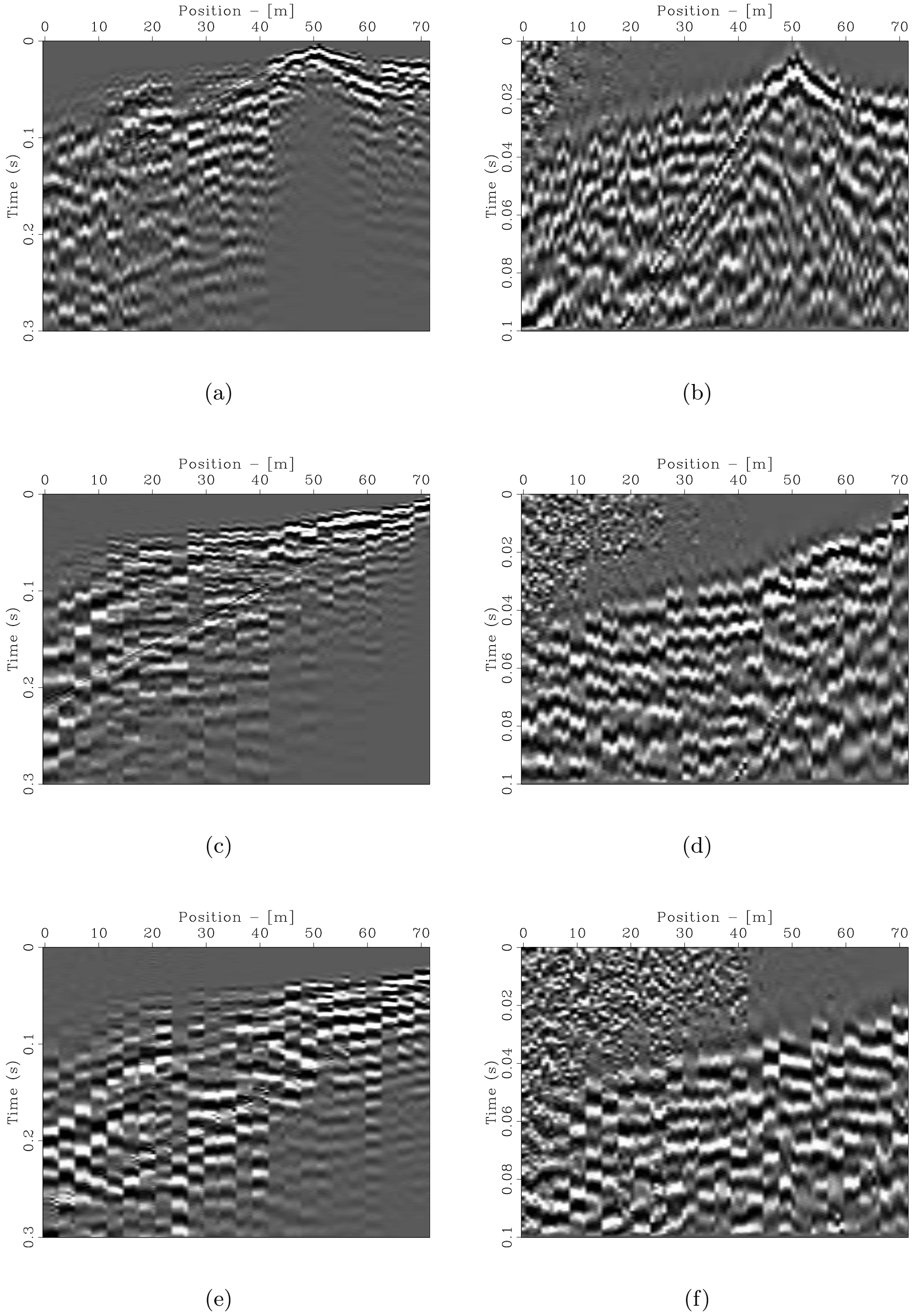


Figure 8: Line 2 shot gathers: a) Raw shot at 50 m, b) shot from (a) 80-500Hz bandpass and AGC and zoom; c) raw shot at 72 m, d) shot from (c) 80-500Hz bandpass and AGC and zoom; e) raw shot at 77 m, f) shot from (e) 80-500Hz bandpass and AGC and zoom. [**ER**] sjoerd3/. s1-inter-208,s2-inter-208,s1-inter-201,s2-inter-201,s1-inter-2m5,s2-inter-2m5

Figures 7(a)-8(b) and 8(a)-8(f) show shot gathers from shots at approximately 0 m, 12 m, 24 m, 50 m, 72 m and 77 m, along Line 2. An averaged and normalized frequency spectrum of data recorded at Line 2 is shown in Figure 3. The data of the shot gather in Figure 7(a) is rather jumbled due to nearby fissures, the interleaving technique is not so effective. Data quality is generally worse than for line 1, and is particularly poor for the NW end of the line. At the SE end of the line, the data more closely resemble those for line 1, and the previously mentioned reflection may be visible between position 45 m to 60 m in Figure 8(d). The interleaving technique falls apart for these data due to strongly heterogeneous wave propagation, larger topographical variations, and the many fissures. There is a high frequency event visible in almost all shot gathers; this is the air wave. Note how the interleaving technique works quite well for that event, as the event is neither jumbled nor aliased. For all of these line 2 data plots, NW is to the left.

## INTERPRETATION

In two of the better quality shot gathers, Figures 6(b) and 6(d), four different events can be distinguished. 2 refracted wave events are observed: a direct P wave event and a reflected wave event. These are annotated in Figures 9 and 10. The interpreted reflection is also visible at positions greater than about 55 m in Figure 6(d). At small offsets we see very slow and dispersive ground roll, annotated $G$.

We performed simple refraction analyses to estimate a velocity profile (Stein and Wysession, 2003). The direct P-wave and two refractions' slopes indicate three velocities: $v_1 = 480$ m/s, $v_2 = 1400$ m/s and $v_3 = 3000$ m/s. These are relatively consistent for both figures, although errors in the range of 30% are possible. The intersection times in Figure 9 indicate two layer thicknesses of $h_0 = 1.2$ m and $h_1 = 5.8$ m. The intersection times in Figure 9 indicate two layer thicknesses of $h_0 = 0.9$ m and $h_1 = 9.1$ m. These differences are due both to estimation error and lateral variation of the geology and topography. However, they suggest two positive velocity discontinuities at approximately 1 m and $8-10$ m. Comparing these depths to the log shown in Figure 11 (depth indicated in feet), suggests that they are not as deep as the coal and could be associated with the top of the thick sandstone bed. Perhaps the thin sandstone bed in the shale, or a positive velocity gradient in the shale, is the reflector. The intersection time of the reflection is approximately 0.02 s, using a estimated velocity of 1400 m/s. This would indicate a reflector at approximately 14 m depth.

The frequency wave-number spectrum in Figure 4 shows significant high wave-number noise. This is partly due to the interleaving technique. The energy associated with reflections, refractions and surface waves is all located and mixed together below wave-numbers of .15 $m^{-1}$ and frequencies below 25 Hz. This offers little opportunity to filter the refractions and surface waves from the reflection. Applying Normal Move Out (NMO) to flatten the reflector on the shot gather of Figure 6(b) was consistent with the velocity estimates.

The shot gather after NMO with three different velocities is shown in Figure 12. A NMO velocity of 1150 m/s seems optimal to flatten the flanks of the interpreted reflector. This NMO velocity was tested for consistency with the velocities coming from refraction. Various stacking velocities, $v_s$, and normal incidence travel times, $\tau$, were tested for their equivalent interval velocity, $v_i$, and layer thicknesses, $h$.

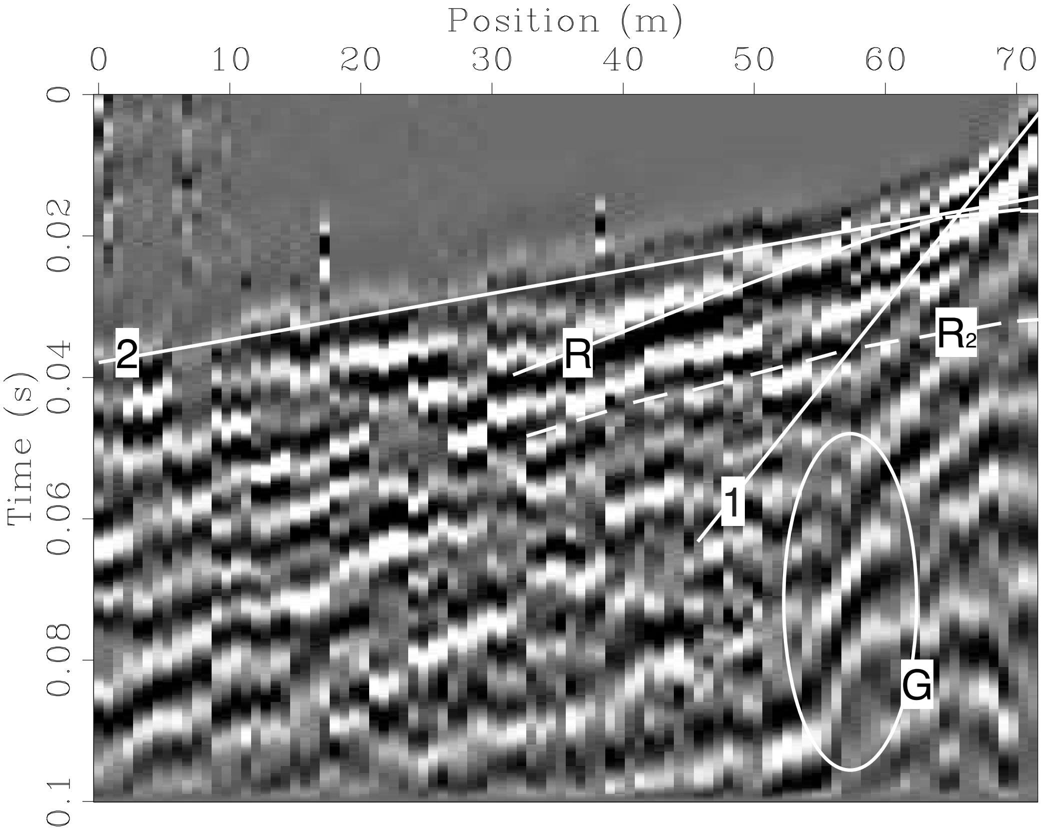


Figure 9: Line 1, shot at southwestern end, 10 m SW of well #11. A possible reflection, annotated $R$, appears between positions 30 to 55 m. A direct P-wave is annotated 1, and two refracted waves are annotated 2 and 3. A possibly hidden reflected event is annotated $R_2$. Dispersive ground roll $G$. [**NR**] sjoerd3/. inter-001-ann

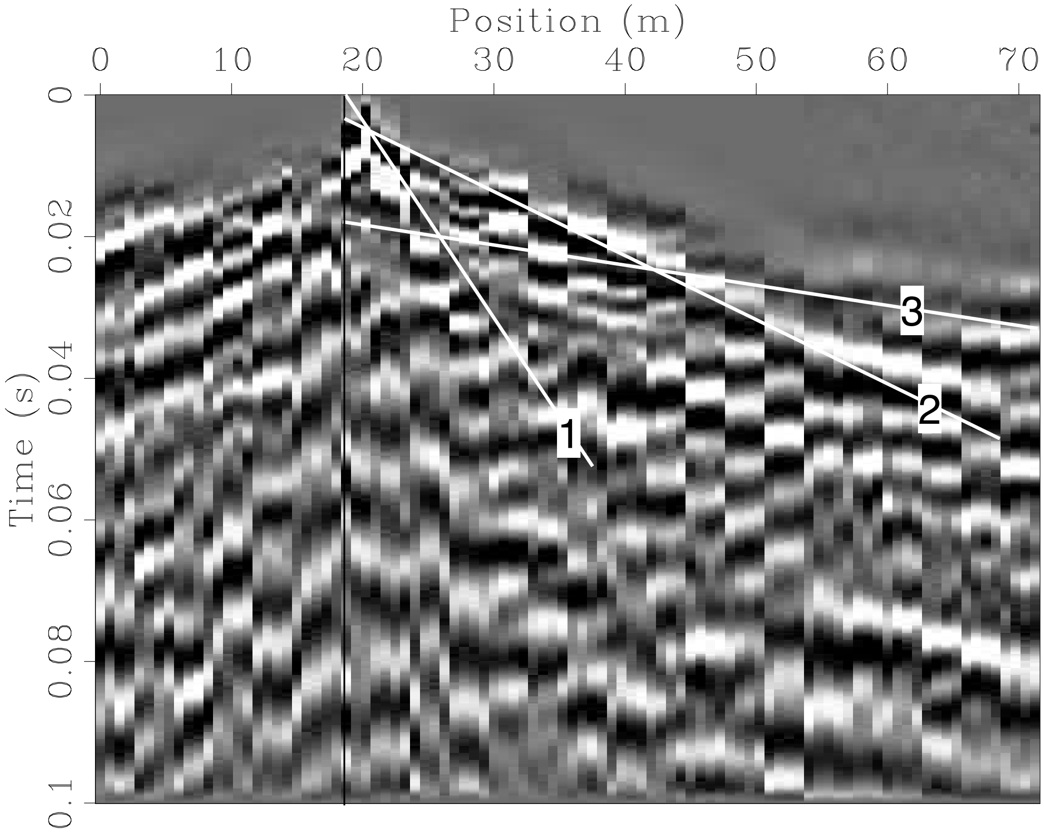


Figure 10: Line 1, shot on sandstone outcrop, in the road. A direct P-wave event annotated 1, two refracted waves annotated 2 and 3. [**NR**] sjoerd3/. inter-018-ann

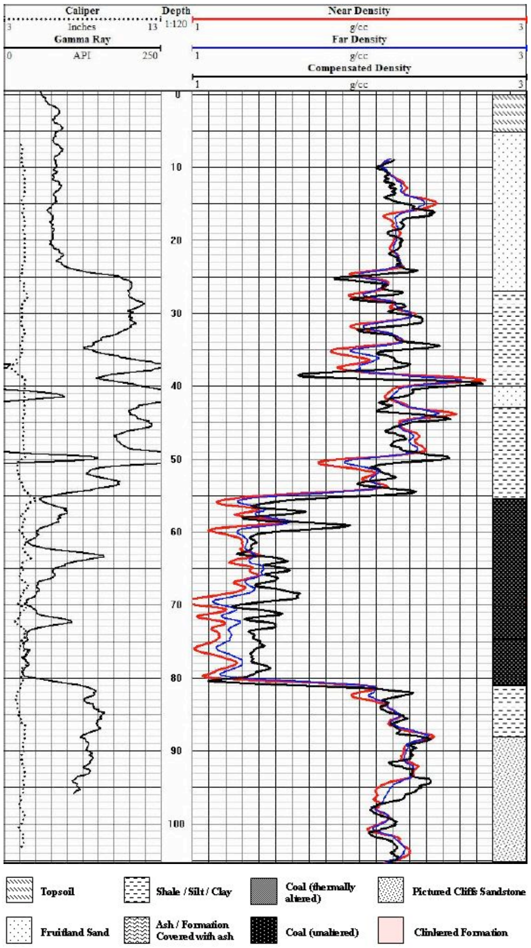


Figure 11: Well log at well #3. Lithology is indicated below, with depth indicated in feet. A 9 m thick coal layer is located at approximately 16 m depth. [**NR**] sjoerd3/. well3

$$\begin{array}{llll}
v_s = 480,\ 1150 & \tau = 0.0055,\ 0.02 & h = 1.32,\ 9.55446 & v = 480,\ 1317.86 \\
v_s = 450,\ 1150 & \tau = 0.005,\ 0.02 & h = 1.125,\ 9.76681 & v = 450,\ 1302.24 \\
v_s = 500,\ 1150 & \tau = 0.0055,\ 0.018 & h = 1.375,\ 8.3722 & v = 500,\ 1339.55 \\
v_s = 480,\ 1150 & \tau = 0.0075,\ 0.02 & h = 1.8,\ 8.78955 & v = 480,\ 1406.33
\end{array} \tag{1}$$

Most combinations are fairly sensible, but pushing the lower velocity to 1400 m/s also pushes the slow top layer to 2 m thick. This result is unrealistic considering our data and field observations. Regardless of exact numbers, it appears that the reflection originates at a depth of approximately 11 m. Revisiting Figure 9, note a weaker event, $R_2$, that is possibly a reflection hidden behind the interpreted reflection $R$. These estimates are rough, but error is unlikely to exceed 30 %.

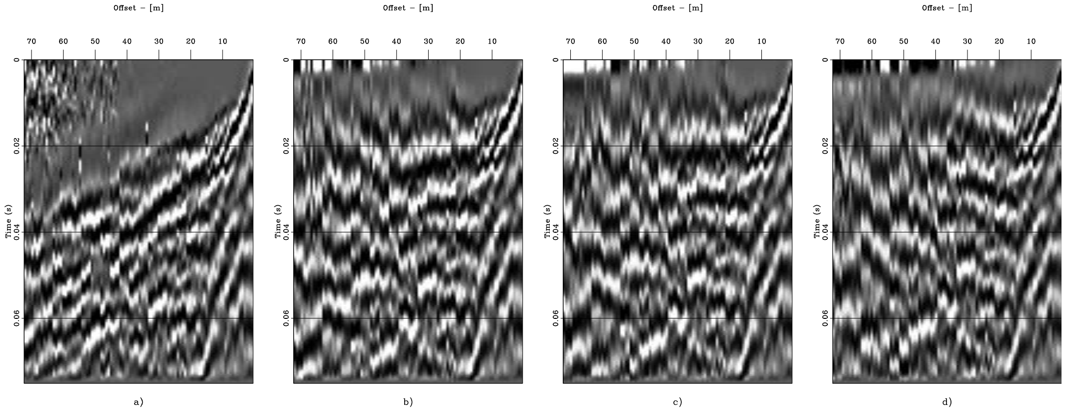


Figure 12: Line 1, shot at southwestern end, 10 m SW of well #11. a) No NMO, b) NMO with $v_s = 1300$, c) $v_s = 1150$, d) $v_s = 1000$. Note that 1150 m/s seems to be the correct NMO velocity. [**ER**] sjoerd3/. nmo-fig

## CONCLUSIONS

We conducted a seismic test at the Southern Ute Nation coal fire site using minimal equipment. Data quality is at the high end of what can be expected for sledgehammer-source data. The recorded frequency content is strong up to 100 Hz. The test shots were interpreted to contain several refracted events and a reflection event.

The refraction and reflection events are interpreted for a subsurface velocity profile. The velocities were higher than anticipated by a previous modeling effort (de Ridder and Haines, 2008). The fast layers that are interpreted to overlie the coal pose a difficulty to any seismic surveying because they are an impediment to deeper wave propagation. As we expected, the fissures present a major impediment to wave propagation and substantially degrade data quality. In addition, the test shots indicate highly dispersive ground roll and strong statics in the area.

Refraction and reflection analyses suggest a meter-thick layer of a little less than 500 m/s on top, and a layer of about 9-10 meters of a velocity of about 1300 − 1400 m/s overlying a lower layer with a velocity as high as 3000 m/s. The reflection event is originating at a depth of approximately 11 m, which is well above the coal layer. It might hide possible reflections from the coal layer.

There are no distinguishable events deep enough to adequately characterize the coal layer. The major difficulty is unexpected fast layers above the coal, as well as a relatively strong reflection event from a layer above the coal. It is conceivable that the depth estimates are inaccurate and the reflection event is from the top of the coal, but this is unlikely. Even if the reflection happens to be from the top of the coal, it is only clearly visible on only a few shot gathers. Thus we must conclude that further seismic work at the site is unlikely to be successful at imaging the targeted coal or ash layer.

## ACKNOWLEDGMENTS

We would like to acknowledge professor Lynn Orr from the Department of Energy Resources Engineering, Stanford University, for financial and aviation support of this project; Bill Flint, Kyle Siesser and Jonathan Begay from the Southern Ute Department of Energy for their financial support and expertise on the field site; Taku Ide and Michael Krause from the Department of Energy Resources Engineering, Stanford University, for expertise on the Southern Ute Nation coal fire site and field support; Nigel Crook from the Department of Geophysics, Stanford University, for his expertise on near surface geophysics; the Southern Ute Indian Tribe Growth Fund for financial support and Finally Jet West Geophysical Services for the log of Figure 11.

## REFERENCES

de Ridder, S. and S. S. Haines, 2008, Seismic investigation of natural coal fires: A pre-fieldwork synthetic feasibility study: Technical report, SEP-136.

Stein, S. and M. Wysession, 2003, An introduction to Seismology, Earthquakes, and Earth Structure: Blackwell.

# Source signature and static shifts estimations for multi-component ocean bottom data

*Mandy Wong and Shuki Ronen*

## ABSTRACT

We present an interpretive study to estimate the source signature and source statics of a field ocean bottom dataset. We use the down-going direct arrival to extract the source signature at different offsets. The down-going wavefield is obtained from a simple summation of the pressure (P) and the vertical partical velocity (Z) of the multi-component data. Such a summation is scaled by a factor that depends on offset and is estimated directly from the amplitude of the P and Z values in the $t-x$ domain. In addition, we compare two approaches to estimating the source-side static shifts. Our static shifts estimation give satisfactory result for an absolute offset up to ±5000 meters.

## INTRODUCTION

Many migration algorithms, such as shot-profile, plane wave, and reverse time migration, require the input of source wavelets as part of the calculation. Ocean bottom data have a distinct advantage in the estimation of the source wavelet, because an ocean bottom seismometer and hydrophone can pick up the source signature as direct arrivals through a relatively homogeneous medium. As shown in figure 1, the source wave path only needs to go through the water once to reach the receivers.

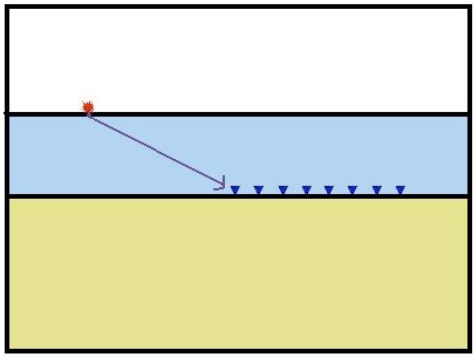


Figure 1: For ocean bottom data acquisition, the source wavelet is picked up by the down-going direct arrival data.[**NR**] mandy1/. DirectSource

Source signatures are offset-dependent because each shot is generated from an array of airguns and the obliquity of the source wave path directly affects the shape of the wavelet. A crude way to extract the source signature from a direct arrival is to time-window the pressure signal at the receiver. However, this technique is often inaccurate because the source bubble often overlaps with the recorded primary reflection. A better way to extract the source signature is to first separate the total data into up-going and down-going wavefields, and then perform time windowing only on the down-going component. We can do that because the direct arrival is strictly down-going, as illustrated in figure 1. For multi-component ocean bottom data, we can perform up-down separation using the pressure (P) and vertical velocity (Z) wavefields. This method is called PZ summation.

There is significant literature on the method of PZ summation to decompose data into up-going and down-going wavefields (Amundsen, 1993; Sonneland et al., 1986). Some methods involve separation in the $\tau - p$ or $\omega - k$ (Fourier) domain. However, transformation into other domains poses a problem when the data are aliased or sparsely sampled. In this study, we perform PZ summation in the physical (t-x) space to avoid this difficulty.

In the next section, we first present the field dataset and discuss some of the challenges. Futhermore, we estimate the source statics in a common receiver gather. Variations in shot deployment depth and the water column cause time anomalies that can be approximated as surface-consistent static time shifts. We compare two methods of static time shift estimation. One method uses the maximum pulse of the source while the other use cross-correlation. Finally, we discuss and show the result of the source signature extraction on the field dataset.

## OCEAN BOTTOM NODES DATASET

The P (pressure) and Z (vertical velocity) components of an ocean bottom common receiver gather are shown in Figures 2. Data are acquired using ocean bottom nodes. The typical water depth in the surveyed area is around 500m with a soft sediment layer of 150m. From the raw data, we can identify direct arrivals, refraction, and water reverberations. There is an outcrop at the sea bottom to the left of the ocean bottom node (OBN) receiver. The dataset is spatially aliased, with 50 meters spacing between shots.

To focus on the source signal, we apply hyperbolic moveout (HMO) at a velocity of 1480m/s and time-window the data near the direct arrival (Figure 3). Linear moveout performs time-shifting ($\Delta\tau$) on each trace. Time shifts are offset-dependent and are calculated as shown in equation 1:

$$\Delta\tau = \sqrt{\tau_o^2 + \frac{x^2}{v^2}} - \tau_o, \qquad (1)$$

where $x$ is the full offset, $v$ is the water velocity, and $\tau_o$ is the zero-offset arrival time. From Figure 3, we can see that the direct arrival is not perfectly flat. This non-flatness indicates shot statics in this common-reciever gather. Shot statics can be caused by variations in deployment depth and cross-line displacement. Another noticeable feature is that primary reflections follow almost immediately after the direct arrivals. This causes difficulty, as primary signal would overlap with the source bubble signal. Next, we present two methods to estimate the source statics by flattening the HMO output.

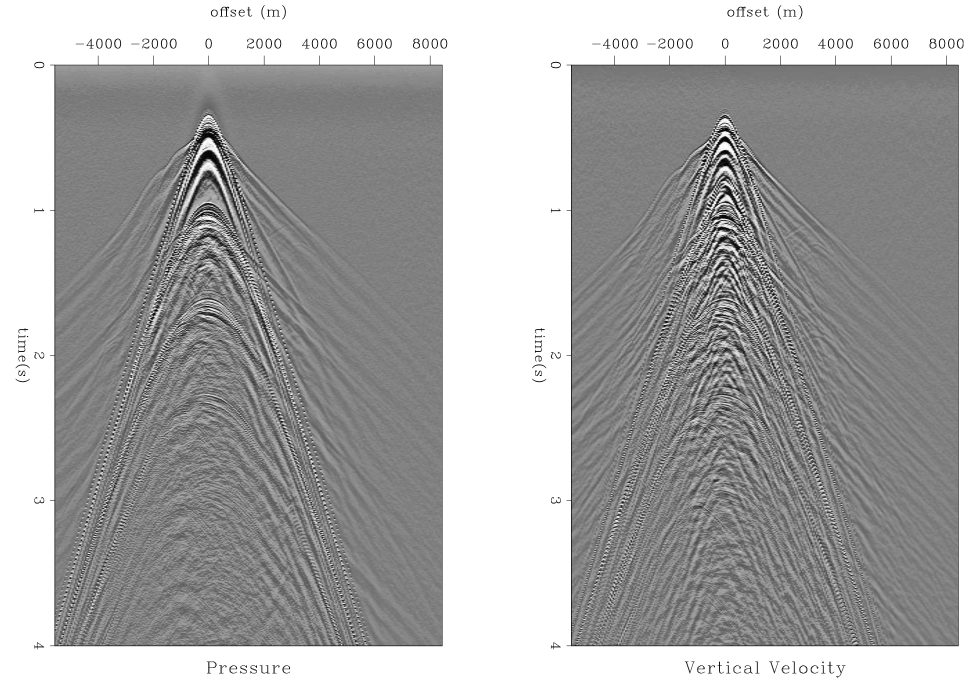


Figure 2: The left plot shows the pressure component and the right plot shows the vertical velocity component. From the raw data, we can identify direct arrivals, refraction, and water reverberations.[**ER**] mandy1/. PZdata

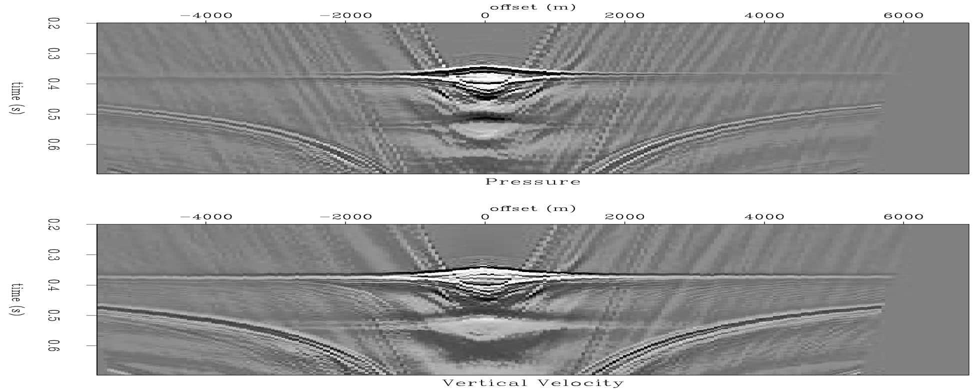


Figure 3: The top plot shows the pressure component and the bottom plot shows the vertical velocity component after hyperbolic moveout correction. The data are time-windowed around direct arrival.[**ER**] mandy1/. HMOraw

## STATIC SHIFTS CORRECTION

### Correction using maximum pulse alignment

One way to calculate the static shifts correction is by aligning the maximum pulse of the source signature. A typical air-gun signature, including the effect of the source ghost, consists of a large pulse and its bubbles, as shown in Figure 4. Since the amplitude of the initial maximum pulse is much higher than the amplitude of the bubble, we can estimate the static correction by picking and aligning the time of the maximum pulse. The middle panel of figure 5 shows the z-component of the resulting direct-arrival after static shifts correction using this method.

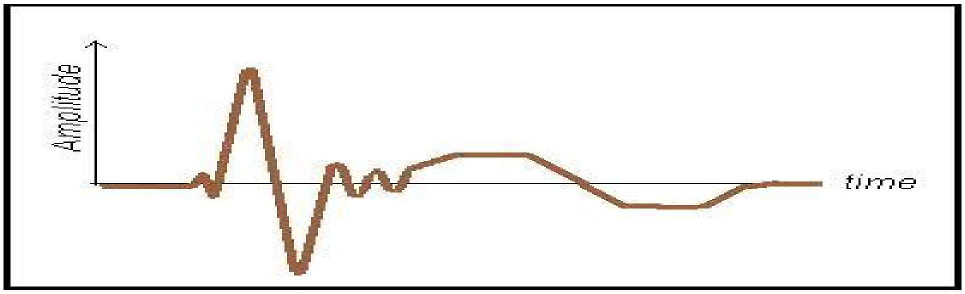


Figure 4: A typical air-gun signature consists of a large pulse and its bubbles. The negative signal comes from the source ghost. **[NR]** mandy1/. exshot

Searching for the maximum pulse can be tricky for traces at large offset, as the amplitude of the pulse attenuates with a longer travel distance. Satisfactory results can be obtained up to an absolute offset of ±5000$m$. We have restricted the search neighborhood to be near the LMO time in order to alleviate this problem. The top panel of figure 6 shows the static shift calculated at different offsets by aligning the maximum pulse.

### Correction using cross-correlation

Another way to estimate static shifts is to observe the cross-correlation of traces at different offsets with the zero-offset trace. In this way, the lag time that gives maximum cross-correlation would define the static shifts. The bottom panel of Figure 5 shows the resulting direct arrival after static shifts corrrection using this method.

To maximize alignment around the direct arrival, we perform cross-correlation only in the neighborhood of the HMO time. Comparing the result from using the maximum pulse method and the cross-correlation method, we can see that the former method performs better. The bottom panel of Figure 6 shows the static shift calculated at different offsets by using the cross-correlation method. From the figure, we see that the direct arrival are better lined up in the region of ±1000 meters for the maximum pulse method.

## UP-DOWN SEPARATION USING PZ SUMMATION

Traditionally, PZ summation is employed to extract the up-going portion of the wavefield with the goal of eliminating water reverberation (Rosales and Guitton, 2004). We use the

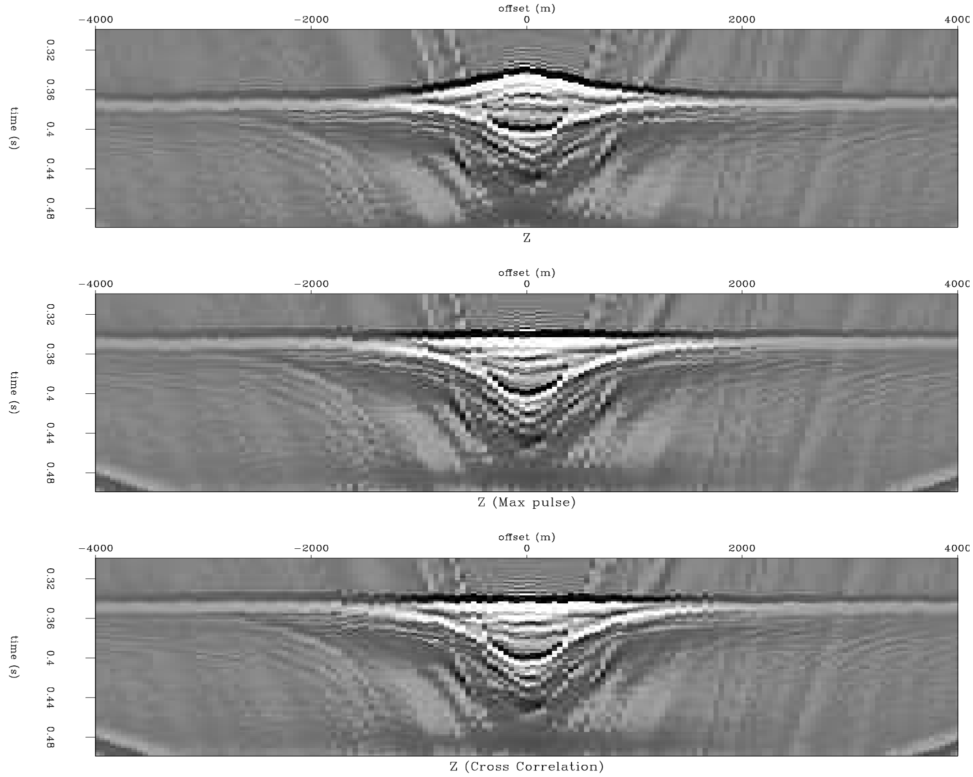


Figure 5: (Top) Vertical velocity before static shifts. (Middle) Vertical velocity after static shifts correction using maximum pulse alignment. (Bottom) Vertical velocity after static shifts correction using cross-correlation. **[ER]** mandy1/. Flat

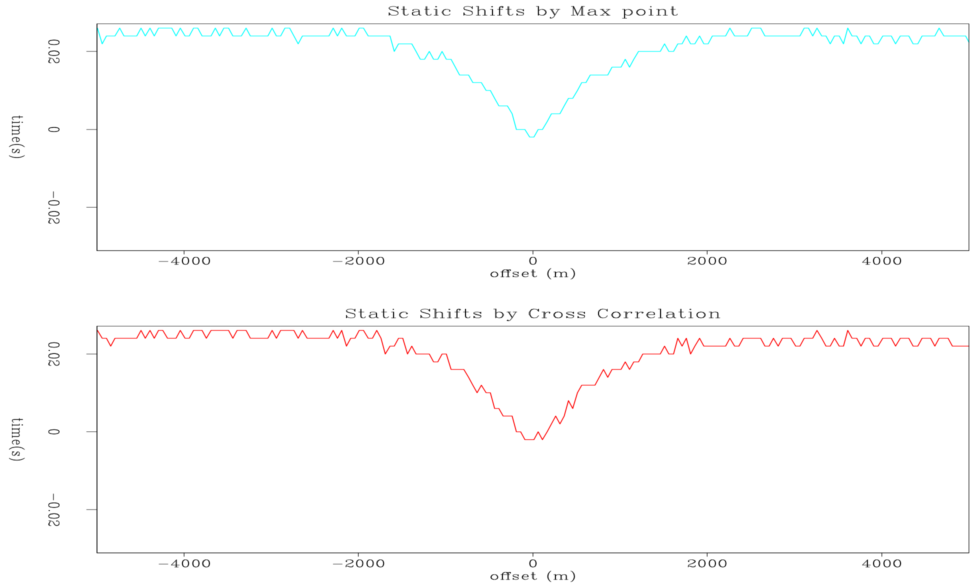


Figure 6: Static shifts estimated at different offsets (Top) using maximum pulse method and (bottom) cross-correlation method. **[ER]** mandy1/. SS

down-going wavefield to estimate the source wavelet at a different offset. For PZ summation, Barr and Sanders (1989) have derived a relation to model the up-going wavefield as shown in the following equation:

$$U(t,x) = \frac{1}{2}\left(P(t,x) + \frac{\rho v_p}{cos\gamma_p}\frac{k_t(1+k_r)}{(1-k_r)}Z(t,x)\right), \tag{2}$$

where $U(t,x)$ is the up-going wavefield, $P(t,x)$ is the pressure, $Z(t,x)$ is the vertical velocity, $\rho$ is the water density, $v_p$ is the P-wave water velocity, $\gamma_p$ is the P-wave refraction angle at the sea bottom for upgoing wavefield, and $k_r$, $k_t$ are the reflection coefficient and the refraction coefficient of the ocean bottom, respectively. One drawback of equation 2 is that it assumes that the reflection coefficent of the ocean surface is -1, which is not always true. We have used a more data-driven approach in which a scaling factor between P and Z is fitted from the amplitude of their direct arrival, as described by equations 3 and 4.

From equation 2, we can see that the scaling factor in front of $Z(t,x)$ is offset-dependent. In this study, instead of calculating the scaling factor from equation 2, we fit for it from the amplitude of the pressure and vertical velocity components, time-windowed around the direct arrival:

$$\begin{aligned} U(t,x) &= \frac{1}{2}\left(P(t,x) + \text{scale}(x)z(t,x)\right), \\ D(t,x) &= \frac{1}{2}\left(P(t,x) - \text{scale}(x)z(t,x)\right), \end{aligned} \tag{3}$$

$$\text{scale}(x) = \frac{\sum_{t\in\Omega_t} \mid P(t,x) \mid}{\sum_{t\in\Omega_t} \mid Z(t,x) \mid}, \tag{4}$$

where scale$(x)$ is the offset dependent scaling factor between pressure and vertical particle velocity, and $\Omega_t$ is the time-window near the direct arrival time. Figure 7 shows the scaling factor computed using equation 4. Figure 8 shows the resulting up-going and down-going signals after PZ summation. Notice that the up-going signal is much weaker than the down going signal.

## SOURCE SIGNATURE EXTRACTION

After obtaining the down-going wavefield, we can obtain an estimate of the source signature from the recorded amplitude of the direct arrival. Figure 9 shows several source wavelets at different offset values. For a near zero-offset wavelet, we can clearly identify the typical parts of a source wavelet, which include the source ghost and the source bubble. The amplitude of the large negative pulse is less than that of the large positive pulse. This fact indicates that the reflection coefficient of the water surface is not exactly -1.

In Figure 9, we see that the amplitude and shape of the source wavelet change drastically with the offset. Primary reflections overpower some of the source signals. The source bubble can hardly be identified.

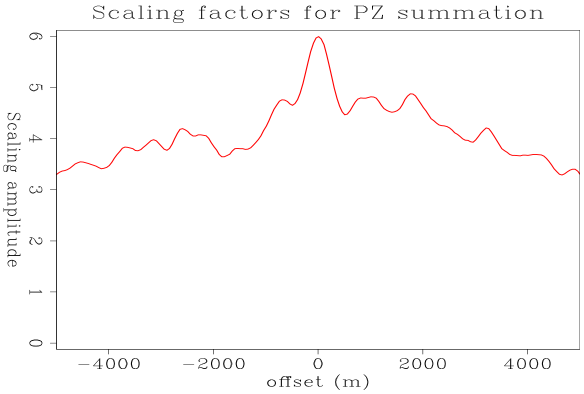


Figure 7: Scaling factor as a function of offset. It is estimated from the average amplitude of P over the average amplitude of Z [**ER**] mandy1/. scale

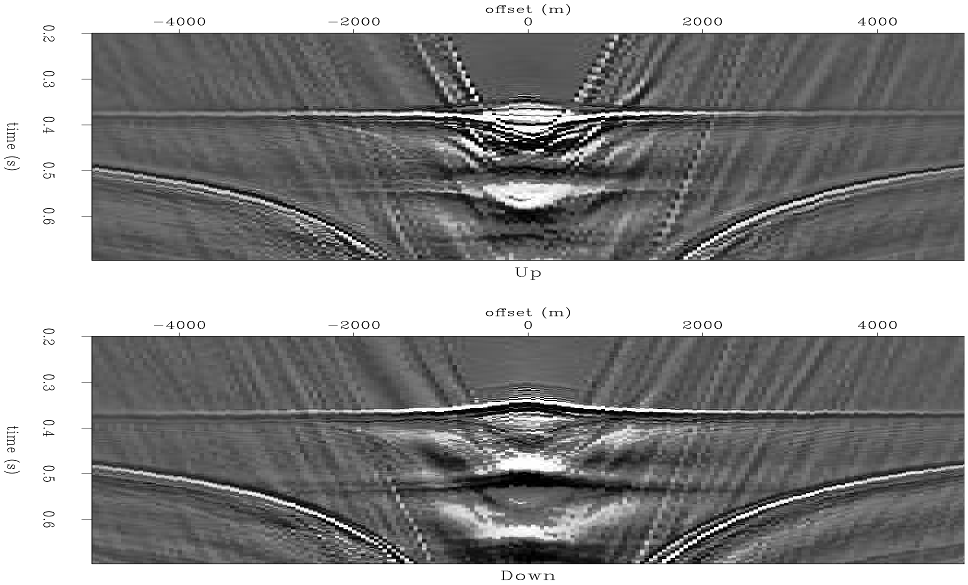


Figure 8: The top shows the resulting up-going wavefields and the bottom shows the down-going wavefields after PZ summation. [**ER**] mandy1/. PZ

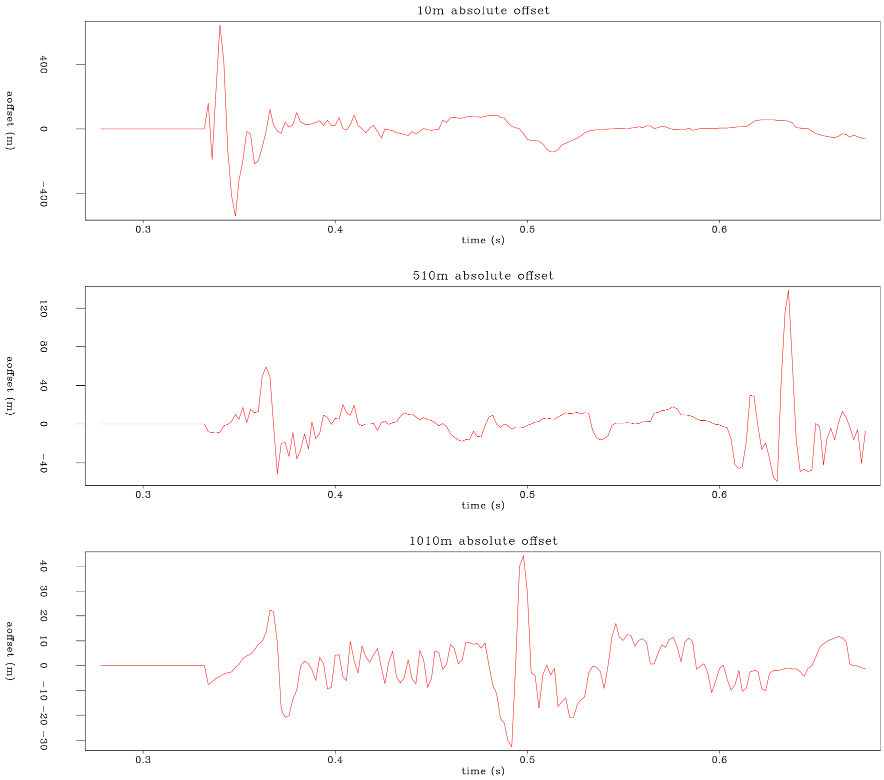


Figure 9: Source wavelets at different offset values. These wavelets are obtained from the amplitude of the down-going wavefields windowed near direct arrival time. **[ER]** mandy1/. slices

Next, we compare our result with another source wavelet that is independently estimated in the same survey. We can see the difference in the large negative pulse and the source bubbles.

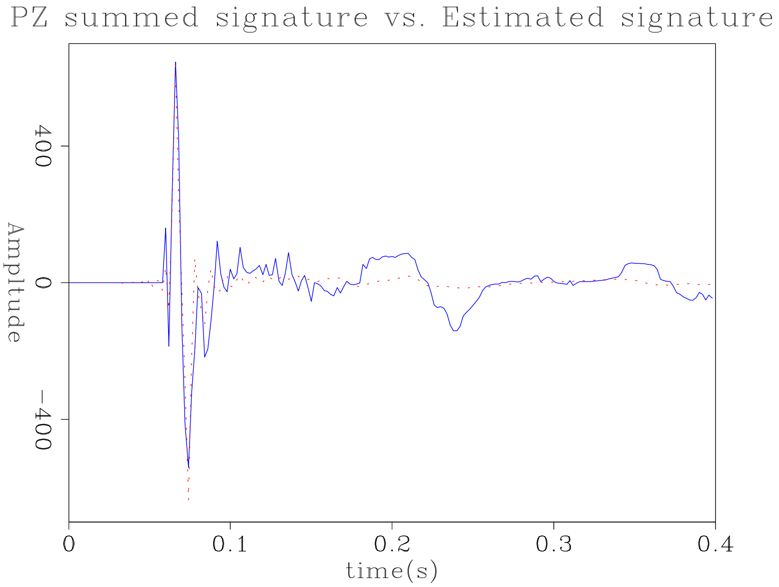


Figure 10: Comparison of our zero-offset source wavelet (solid line) with another wavelet independently estimated in the same survey (dashed line). Our wavelet indicates that the ocean surface reflection coefficent is not -1. Also, the amplitude of the source bubble is bigger. **[ER]** mandy1/. compare

## CONCLUSIONS

We performed an interpretive study of a field ocean bottom dataset to extract its source signature as a function of the offset and to estimate the source statics. Source signatures are obtained by capturing the direct arrival signal for the down-going wavefields. PZ summation is performed in the t-x domain with a scaling factor that is fitted from the data. Source statics are estimated by correcting the deviations from the linear moveout time. We have used two methods of performing that correction. Shifts estimations are consistent between the two approaches at small offsets. Our approaches become limited at larger offsets. As the amplitude of the signature becomes weaker with offset, obtaining the source signature and estimating the static shifts become difficult.

## ACKNOWLEDGMENTS

We thank SeaBird Explorations for the release of this field dataset. We also thank the sponsors of Stanford Exploration Project for funding this research.

## REFERENCES

Amundsen, L., 1993, Wavenumber-based filtering of marine point-source data: Geophysics, **58**, 1335–1348.

Barr, F. J. and J. L. Sanders, 1989, Attenuation of water-column reverations using pressure and velocity detectors in a water-bottom cable: SEG Expanded Abstracts, **59**, 653–656.

Rosales, D. A. and A. Guitton, 2004, Ocean-bottom hydrophone and geophone coupling: SEP-Report, **115**, 57–70.

Sonneland, L., L. E. Berg, P. Eidsvig, A. Haugen, B. Fotland, and J. Vestby, 1986, 2-d deghosting using vertical receiver arrays: SEG Expanded Abstracts, **5**, 516.

# Effective medium theory for elastic composites

*James G. Berryman*

## ABSTRACT

The theoretical foundation of a variant on effective medium theories for elastic constants of composites is presented and discussed. The connection between this approach and the methods of Zeller and Dederichs, Korringa, and Gubernatis and Krumhansl is elucidated. A review of the known relationships between the various effective medium theories and rigorous bounding methods for elastic constants is also provided.

## INTRODUCTION

In a series of papers [Berryman (1979, 1980a,b)], a variant on effective medium theories for elastic composites was developed by the author. In this paper, I will review the derivation of the effective medium formulas for the elastic constants of composites while elucidating the relationships between my results and the results from effective medium theories proposed by others. These results are then compared to known rigorous bounds on the effective elastic constants.

The general background for theories of elastic composites with special emphasis on earth sciences applications is provided in the review articles by Watt et al. (1976) and Berryman (1995). Another review of effective medium theories with an emphasis on connections to general applied physics applications is given by Elliott et al. (1974). Related work by Willis (1977, 1981) is also especially useful for some of the cases not considered here, including anisotropic media and polycrystalline composites.

## EFFECTIVE ELASTIC CONSTANTS

Mal and Knopoff (1967) derived an integral equation for the scattered displacement field from a single elastic scatterer. Let $\Omega_i$ symbolize the volume of the region occupied by a single inclusion $i$. Let the incident field be $\vec{u}^0(\vec{x})\exp(-i\omega t)$ and let $\vec{u}(\vec{x})\exp(-i\omega t)$ and $\vec{v}(\vec{x})\exp(-i\omega t)$ be the total field outside and inside the inclusion volume such that

$$\begin{array}{lll} \vec{u}(\vec{x}) = \vec{u}^0(\vec{x}) + \vec{u}^s(\vec{x}) & \text{for} & \vec{x} \notin \Omega_i, \\ \vec{v}(\vec{x}) = \vec{u}^0(\vec{x}) + \vec{v}^s(\vec{x}) & \text{for} & \vec{x} \in \Omega_i. \end{array} \tag{1}$$

The scattered fields are $\vec{u}^s$ and $\vec{v}^s$. Both $\vec{u}(\vec{x})$ and $\vec{u}^0(\vec{x})$ satisfy the same equation:

$$c^m_{\ell npq}\frac{\partial^2 u_p}{\partial x_n \partial x_q} + \rho_m \omega^2 u_\ell = 0 \tag{2}$$

outside the inclusion, while $\vec{v}(\vec{x})$ satisfies

$$c^i_{\ell npq}\frac{\partial^2 v_p}{\partial x_n \partial x_q} + \rho_i \omega^2 v_\ell = 0 \tag{3}$$

inside the inclusion. The indices $\ell$,$n$,$p$,$q$ take the values $1, 2, 3$ for the three spatial dimensions, and the Einstein summation convention applies in Equations (2) and (3), and also throughout this paper. The elastic tensor for the matrix and inclusion are respectively:

$$c^m_{\ell npq} = \lambda_m \delta_{\ell n}\delta_{pq} + \mu_m \left(\delta_{\ell p}\delta_{nq} + \delta_{np}\delta_{\ell q}\right), \tag{4}$$

$$c^i_{\ell npq} = \lambda_i \delta_{\ell n}\delta_{pq} + \mu_i \left(\delta_{\ell p}\delta_{nq} + \delta_{np}\delta_{\ell q}\right) \equiv c^m_{\ell npq} + \Delta c^i_{\ell npq}, \tag{5}$$

and $\rho_m$ and $\rho_i (\equiv \rho_m + \Delta\rho_i)$ are the respective densities.

Green's function for a point source in an infinite, isotropic, homogeneous elastic medium of the matrix material is given by

$$g_{pq}(\vec{x},\vec{\zeta}) = \frac{1}{4\pi\rho_m\omega^2}\left[s^2\frac{\exp{(isr)}}{r}\delta_{pq} - \frac{\partial^2}{\partial x_p \partial x_q}\left(\frac{\exp{(ikr)}}{r} - \frac{\exp{(isr)}}{r}\right)\right], \tag{6}$$

where $r = |\vec{x} - \vec{\zeta}|$, $k = \omega[\rho_m/(\lambda_m + 2\mu_m)]^{1/2}$, and $s = \omega[\rho_m/\mu_m]^{1/2}$ — with $k$ and $s$ being, respectively, the magnitudes of the wavevectors for compressional and shear waves in the matrix. Given the form of $g_{pq}$, Mal and Knopoff (1967) then derive an integral equation for $\vec{u}(\vec{x})$. Since the derivation follows standard lines of argument, I will not repeat it here. The result is

$$u_\ell(\vec{x}) = u^0_\ell(\vec{x}) + \int_{\Omega_i} d\vec{\zeta}\left[\Delta\rho_i\omega^2 v_n(\vec{\zeta}) - \Delta c^i_{njpq}\epsilon_{pq}\frac{\partial}{\partial\zeta_j}\right] g_{\ell n}(\vec{x},\vec{\zeta}). \tag{7}$$

Equation (7) is an exact integral equation for the displacement field in the region exterior to the scatterer in terms of the displacement and strain fields inside the inclusion volume $\Omega_i$.

To evaluate the integral (7), estimates of the interior displacement and strain fields are required. Considering the first Born approximation from quantum scattering theory suggests the estimates for wave speed and strain at $\vec{\zeta} \in \Omega_i$:

$$\vec{v}(\vec{\zeta}) \simeq \vec{u}^0(\vec{\zeta}), \tag{8}$$

and

$$\epsilon_{pq}(\vec{\zeta}) \simeq \epsilon^0_{pq}(\vec{\zeta}). \tag{9}$$

By Equations (8) and (9), I mean to approximate $\vec{v}$ and $\epsilon$ by the values $\vec{u}^0$ and $\epsilon^0$ would have achieved at position $\vec{\zeta}$ if the matrix contained no scatterers. For scatterers with small volumes, it follows from (7) that $\vec{u}^s(\vec{x})$ and its derivatives are small quantities for $\vec{x}$ outside of $\Omega_i$. Since the displacement is continuous across the boundary, it follows that Equation (8) will be a good approximation to $\vec{v}(\vec{\zeta})$. However, this argument fails for Equation (9), because the strains are not continuous across the boundary. Equation (9) should therefore be replaced by the formula:

$$\epsilon_{pq} = T_{pqrs}\epsilon^0_{rs}, \tag{10}$$

where $T$ is Wu's tensor [Wu (1966)], relating $\epsilon_{pq}$ for an arbitrary ellipsoidal inclusion to the uniform strain at infinity $\epsilon^0_{pq}$. Now, if the wavelength of the incident waves is large compared to the size of the ellipsoid (*i.e.*, $a/\bar{\lambda} << 1$, where $\bar{\lambda}$ is the wavelength), then the fields both near the ellipsoid and inside scatterer volume $\Omega_i$ will be essentially static and uniform [Eshelby (1957)]. Thus, to the lowest order of approximation, it is valid to make the substitutions (8) and (10). When the ellipsoid is centered at $\zeta_i$, it follows easily that

$$u^s_\ell(\vec{x}) = \Omega_i\left[\Delta\rho_i\omega^2 u^0_n(\vec{\zeta_i}) g_{\ell n}(\vec{x},\vec{\zeta_i}) - \left(\Delta\lambda^i T_{pprs}\delta_{nj} + 2\Delta u^i T_{njrs}\right)\epsilon^0_{rs} g_{\ell n,j}(\vec{x},\vec{\zeta_i})\right], \tag{11}$$

where the symmetry properties of $T$ have been used in simplifying the expression. A comma preceding a subscript indicates a derivative with respect to the as-labelled component.

Equation (11) gives the first order estimate of the scattered wave from an ellipsoidal inclusion whose principal axes are aligned with the coordinate axes. When the ellipsoid is oriented arbitrarily with respect to the coordinate axes, Equation (11) must be changed by replacing $T_{pqrs}$ everywhere with

$$U_{pqrs} = \ell_{p\alpha}\ell_{q\beta}\ell_{r\gamma}\ell_{s\delta}T_{\alpha\beta\gamma\delta}, \tag{12}$$

where $\ell_{\alpha\beta}$ are the appropriate direction cosines. For homogeneous, isotropic composites with randomly oriented ellipsoidal inclusions, the general form of the average tensor as given by Wu (1966) is

$$\bar{U}_{pqrs} = \frac{1}{3}(P-Q)\delta_{pq}\delta_{rs} + \frac{1}{2}Q(\delta_{pr}\delta_{qs} + \delta_{ps}\delta_{qr}), \tag{13}$$

where

$$P = \frac{1}{3}T_{ppqq} \qquad \text{and} \qquad Q = \frac{1}{5}\left(T_{pqpq} - T_{ppqq}\right). \tag{14}$$

Finally, suppose $N$ inclusions are contained in a small volume of radius $a$ centered at $\vec{\zeta}_0$. Assume that the effects of multiple scattering may be neglected at sufficiently low frequencies (*i.e.*, long wavelengths appropriate for seismology) to the lowest order. Then, to the same degree of approximation used in Equation (11) (*i.e.*, $a/\bar{\lambda} << 1$), the scattered wave has the form:

$$\langle u^s_\ell(\vec{x})\rangle^m \simeq \sum_{i=1}^{N}\Omega_i\left[\Delta\rho_i\omega^2 u^0_n(\vec{\zeta}_0)g_{\ell n}(\vec{x},\vec{\zeta}_0) - \left(\Delta\lambda^i\bar{U}^{mi}_{pprs}\delta_{nj} + 2\Delta\mu^i\bar{U}^{mi}_{njrs}\right)\epsilon^0_{rs}g_{\ell n,j}(\vec{x},\vec{\zeta}_0)\right], \tag{15}$$

where the superscripts $m$ and $i$ again refer to matrix and inclusion properties, respectively. Note especially that distinct superscripts $i$ must be used in Equation (15) to specify both the inclusion *material* itself, and also the *shape* of each distinct type of inclusion.

To apply this thought experiment to the analytical problem of estimating elastic constants, consider replacing the true composite sphere with a sphere composed of matrix material identical to the imbedding material and of ellipsoidal inclusions of the same materials as those in the true composite, and also in the same proportions. Then, if multiple scattering effects may be (and are) neglected, the theoretical expression which determines the elastic constants is

$$\langle u^s_\ell(\vec{x})\rangle^* = 0, \tag{16}$$

where the left hand side is given by Equation (15) with matrix-type $m = *$. Equation (16) states simply that the net (overall) scattering — due to many scatterers – in the self-consistently determined medium vanishes to lowest order.

If the volume fraction of the $i$-th component is defined by $f_i = \Omega_i/\sum_{j=1}^{N}\Omega_j$, then Equation (16) implies the following formulas:

$$\sum_{i=1}^{N} f_i(\rho_i - \rho^*) = 0, \tag{17}$$

$$\sum_{i=1}^{N} f_i(K_i - K^*)P^{*i} = 0, \tag{18}$$

and

$$\sum_{i=1}^{N} f_i(\mu_i - \mu^*)Q^{*i} = 0. \tag{19}$$

Equation (17) states that the effective density $\rho^*$ is just the volume average density (which is what one might reasonably expect, but nevertheless is not always true for effective medium theories). Equations (18) and (19) provide implicit formulas for $K^*$ and $\mu^*$. Such implicit formulas are typically solved numerically by iteration [Berryman (1980b)]. This step is usually necessary because the factors $P^{*i}$ and $Q^{*i}$ are themselves both typically functions of both the unknown quantities $K^*$ and $\mu^*$. Experience has shown that such iterative methods often converge in a stable fashion, and usually after a small number of iterations (typically 10 or less).

The derivation given and final results attained here are very similar to methods discussed by Elliott et al. (1974) and Gubernatis and Krumhansl (1975). I will therefore refer to the resulting effective medium method as the "coherent potential approximation" (or CPA), as is typically done in the physics literature, since the early work of Soven (1967). Equations (18) and (19) were also obtained independently by Korringa et al. (1979), while using an entirely different method. In the following sections, I will compare the results obtained from this effective medium theory to the known rigorous bounds on elastic constants and also to the results of other effective medium theories.

## RIGOROUS BOUNDS ON EFFECTIVE MODULI

In their review article, Watt et al. (1976) discuss various rigorous bounds on the effective moduli of composites. For example, the well-known Voigt (arthimetic) and Reuss (harmonic) averages are, respectively, rigorous [Hill (1952)] upper and lower bounds for both $K^*$ and $\mu^*$. Generally tighter bounds have also been given by Hashin and Shtrikman (1961, 1962, 1963).

Still tighter bounds have been obtained in principle by Beran and Molyneux (1966) for the bulk modulus and by McCoy (1970) for the shear modulus. However, the resulting formulas depend on three-point spatial correlation functions for the composite and are therefore considerably more difficult to evaluate than the expressions for the Hashin-Shtrikman [Hashin and Shtrikman (1961, 1962, 1963)] bounds, which depend only on the material constants and volume fractions. Miller (1969b,a) evaluated the bounds of Beran and Molyneux (1966) by treating an isotropic homogeneous distribution of statistically independent cells. Silnutzer (1972) used the same approach to simplify the bounds of McCoy (1970) for cell materials. Furthermore, Milton (1981) has shown that the bounds of Beran and Molyneux (1966) and McCoy (1970) can be simplified somewhat even if the composite is not a cell material. Nevertheless, the bounds which are most easily evaluated are still the Hashin-Shtrikman [Hashin and Shtrikman (1961, 1962, 1963)] (HS) bounds, the Beran-Molyneux-Miller (BMM) bounds, and the McCoy-Silnutzer (MS) bounds. I will compare these bounds to the estimates obtained from the coherent potential approximation (CPA), the specific effective medium theory being stressed here.

To aid in the following comparisons, it is convenient to introduce two functions:

$$\Lambda(x) = \left(\sum_{i=1}^{N} \frac{f_i}{K_i + 4x/3}\right)^{-1} - \frac{4}{3}x, \tag{20}$$

$$\Gamma(y) = \left(\sum_{i=1}^{N} \frac{f_i}{\mu_i + y}\right)^{-1} - y, \tag{21}$$

together with a third function that is needed in conjunction with $\Gamma$:

$$F(x,z) = \frac{x}{6}\left(\frac{9z + 8x}{z + 2x}\right). \tag{22}$$

It has been shown previously [Berryman (1980b, 1995)] that $\Lambda(x)$ and $\Gamma(y)$ are monotonically increasing functions of their real arguments. Similarly, I find that

$$\frac{\partial F}{\partial z} = \frac{5x^2}{3(z+2x)^2} \geq 0 \tag{23}$$

and

$$\frac{\partial F}{\partial x} = \frac{9z^2 + 16xz + 16x^2}{6(z+2x)^2} \geq 0 \quad \text{if} \quad x \geq 0, \quad \& \quad z \geq 0. \tag{24}$$

So, when both arguments of $F(x,z)$ are non-negative (which will soon be shown to be the case in these applications), it follows that $F$ is a monotonically increasing function of both arguments.

Now, if I define the minimum and maximum moduli among all the constitutents by

$$\begin{aligned} K_+ &= \max(K_1,\ldots,K_N), \qquad & K_- &= \min(K_1,\ldots,K_N), \\ \mu_+ &= \max(\mu_1,\ldots,\mu_N), \qquad & \mu_- &= \min(\mu_1,\ldots,\mu_N), \end{aligned} \tag{25}$$

then the Hashin-Shtrikman bounds are also given in general by

$$K^{\pm}_{HS} = \Lambda(\mu_\pm) \tag{26}$$

and

$$\mu^{\pm}_{HS} = \Gamma\left[F(\mu_\pm, K_\pm)\right]. \tag{27}$$

[Note that the only combinations considered on the right-hand side of (27) are those having both pluses or both minuses – no mixing of the subscripts.]

The Beran-Molyneux-Miller bounds and the McCoy-Silnutzer bounds are known for two-phase composites (*i.e.*, $N = 2$). These bounds can be written in succinct form using the notation of Milton (1981). By defining two geometric parameters $\zeta_1 = 1 - \zeta_2$ and $\eta_1 = 1 - \eta_2$, and two related averages [analogous to the volume fraction weighted average $\langle M\rangle = f_1M_1 + f_2M_2$] of any modulus $M$ by $\langle M\rangle_\zeta = \zeta_1M_1 + \zeta_2M_2$, and $\langle M\rangle_\eta = \eta_1M_1 + \eta_2M_2$, then the bounds can be written very concisely as:

$$K^{+}_{BMM} = \Lambda\left(\langle\mu\rangle_\zeta\right), \tag{28}$$

$$K^{-}_{BMM} = \Lambda\left(\langle 1/\mu\rangle_\zeta^{-1}\right), \tag{29}$$

$$\mu^+_{MS} = \Gamma(\Theta/6), \tag{30}$$

and

$$\mu^-_{MS} = \Gamma(\Xi^{-1}/6), \tag{31}$$

where

$$\Theta = \left[10\left<\mu\right>^2\left<K\right>_\zeta + 5\left<\mu\right>\left<2K+3\mu\right>\left<\mu\right>_\zeta + \left<3K+\mu\right>^2\left<\mu\right>_\eta\right]/\left<K+2\mu\right>^2 \tag{32}$$

and

$$\Xi = \left[10\left<K\right>^2\left<\frac{1}{K}\right>_\zeta + 5\left<\mu\right>\left<2K+3\mu\right>\left<\frac{1}{\mu}\right>_\zeta + \left<3K+\mu\right>^2\left<\frac{1}{\mu}\right>_\eta\right]/\left<9K+8\mu\right>^2. \tag{33}$$

For symmetric cell materials, it is known that $\zeta_1 = \eta_1 = f_1$ for spherical cells, $\zeta_1 = \eta_1 = f_2$ for disks, while $\zeta_1 = (3f_1 + f_2)/4$, and $\eta_1 = (5f_1 + f_2)/6$ for needles.

It is particularly simple to compare these bounds with the results of effective medium theory when the inclusions are assumed to be spherical in shape. Then, the estimates of the moduli are given by the self-consistent formulas (which are mutually interdependent):

$$K^* = \Lambda(\mu^*) \tag{34}$$

and

$$\mu^* = \Gamma\left[F\left(\mu^*, K^*\right)\right]. \tag{35}$$

Furthermore, the bounds (28)–(30) simplify in this case and are given by

$$K^+_{BMM} = \Lambda\left(\left<\mu\right>\right), \tag{36}$$

$$K^+_{BMM} = \Lambda\left(\left<1/\mu\right>^{-1}\right), \tag{37}$$

and

$$\mu^+_{MS} = \Gamma\left[F\left(<\mu>, <K>\right)\right]. \tag{38}$$

From the monotonicity properties of the functions (20)–(22), from elementary arguments relating the estimates to the Voigt and Reuss averages, and also from the fact that all the arguments of these functions depend on quantities composed of elastic constants averaged using positive measures such as volume fractions and the related quantities for various cell-material shapes, I find for the bulk modulus that

$$\Lambda(\mu_-) \le \Lambda\left(\left<1/\mu\right>^{-1}\right) \le \Lambda(\mu^*) = K^* \le \Lambda\left(\left<\mu\right>\right) \le \Lambda(\mu_+), \tag{39}$$

or equivalently that

$$K^-_{HS} \le K^-_{BMM} \le K^* \le K^+_{BMM} \le K^+_{HS}. \tag{40}$$

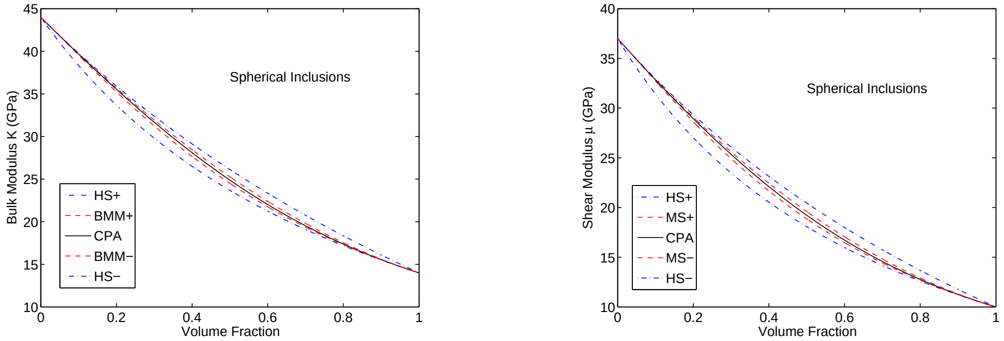
Similarly, by making use of $\Gamma(y)$ from (21), it follows for the shear modulus that

$$\mu^-_{HS} \le \mu^* \le \mu^+_{MS} \le \mu^+_{HS}. \tag{41}$$

The detailed argument leading to Equation (39) is a little involved: First, I must show that $K^*$, $\mu^*$ are bounded by the Hashin-Shtrikman bounds [Berryman (1980a)]. Then, since the Hashin-Shtrikman bounds are themselves bounded by the Voigt and Reuss bounds, Equation (39) follows from

$$\Lambda(\langle 1/\mu\rangle^{-1}) \le \Lambda(\mu_{HS}^{-}) \le \Lambda(\mu^{*}) \le \Lambda(\mu_{HS}^{+}) \le \Lambda(\langle\mu\rangle). \tag{42}$$

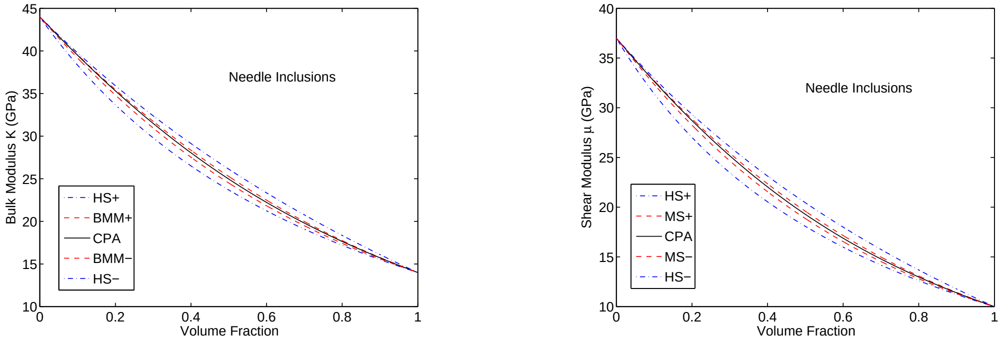
The arguments just given are valid *only* for the case of spherical inclusions. The author knows of no general argument relating the effective medium results to the rigorous bounds for arbitrary inclusion shapes. However, as will be observed in the following Figures, numerical examples illustrate the effective medium estimates always lying between the bounds.



(a) (b)

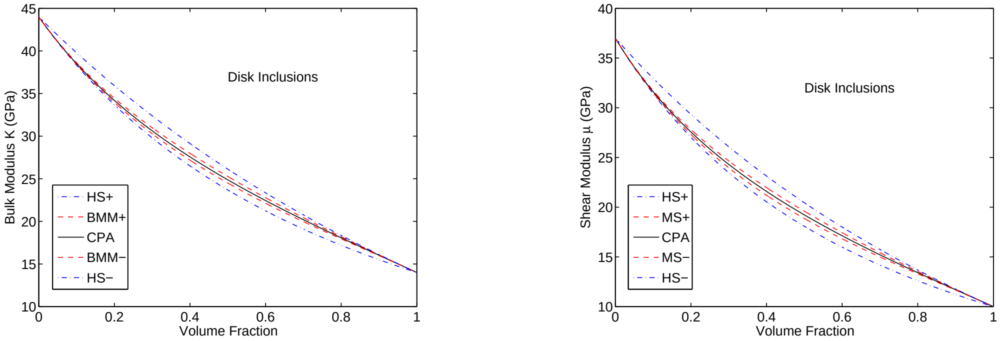
Figure 1: Estimates of the effective bulk (a) and shear (b) moduli of elastic composites with constituents $K_1 = 44.0$ GPa, $\mu_1 = 37.0$ GPa, $K_2 = 14.0$ GPa, and $\mu_2 = 10.0$ GPa as the volume fraction of type-2 increases. The curves are respectively the CPA (or coherent potential approximation: a self-consistent estimator) — which is the black solid line, the Beran-Molyneux-Miller bounds for the bulk modulus and the McCoy-Silnutzer bounds for the shear modulus — which are the red dashed lines, and the Hashin-Shtrikman bounds — which are the blue dot-dashed lines. Inclusions are treated as having spherical shape. **NR** jim1/. K-SPH,G-SPH

Typical results are presented in Figures 1–3. The values of the constituents' moduli were chosen to be: $K_1 = 44.0$ GPa, $\mu_1 = 37.0$ GPa, $K_2 = 14.0$ GPa, and $\mu_2 = 10.0$ GPa. The values of $K_2$ and $\mu_2$ were chosen as a compromise between two extremes: (a) If $K_2$ and $\mu_2$ are too close to $K_1$ and $\mu_1$, then the bounds are too close together to be distinguishable on the plots. (b) If $K_2$ and $\mu_2$ are both chosen to be zero, the iteration to the effective medium



(a) (b)

Figure 2: Estimates of the effective bulk (a) and shear (b) moduli of elastic composites with constituents $K_1 = 44.0$ GPa, $\mu_1 = 37.0$ GPa, $K_2 = 14.0$ GPa, and $\mu_2 = 10.0$ GPa as the volume fraction of type-2 increases. The curves are respectively the CPA (or coherent potential approximation: a self-consistent estimator) — which is the black solid line, the Beran-Molyneux-Miller bounds for the bulk modulus and the McCoy-Silnutzer bounds for the shear modulus — which are the red dashed lines, and the Hashin-Shtrikman bounds — which are the blue dot-dashed lines. Inclusions are treated here as having needle-like shape. **NR** $\boxed{\text{jim1/. K-NDL,G-NDL}}$



(a) (b)

Figure 3: Estimates of the effective bulk (a) and shear (b) moduli of elastic composites with constituents $K_1 = 44.0$ GPa, $\mu_1 = 37.0$ GPa, $K_2 = 14.0$ GPa, and $\mu_2 = 10.0$ GPa as the volume fraction of type-2 increases. The curves are respectively the CPA (or coherent potential approximation: a self-consistent estimator) — which is the black solid line, the Beran-Molyneux-Miller bounds for the bulk modulus and the McCoy-Silnutzer bounds for the shear modulus — which are the red dashed lines, and the Hashin-Shtrikman bounds — which are the blue dot-dashed lines. Inclusions are treated here as having disk-like shape. **NR** jim1/. K-DSK,G-DSK

theory results does not converge for the case of disk-like inclusions [Berryman (1980b)], although all the other cases converge without difficulties. I find in all cases considered that the effective medium theory results lie between the rigorous bounds, as stated above.

## OTHER EFFECTIVE MEDIUM THEORIES

A great variety of effective medium theories exist for studies of the elastic properties of composites. Of these theories, the scattering theory presented by Zeller and Dederichs (1973), Korringa (1973), and Gubernatis and Krumhansl (1975) have the most in common with the scattering-theory approach presented here. However, the present approach appears to be unique among the self-consistent scattering-theory variety, being dynamic (*i.e.*, frequency dependent), while all the others are based on static or quasi-static derivations. This difference becomes a very useful advantage if we want to generalize the approach to finite (nonzero) frequencies, as is required for viscoelastic media. The bounding arguments presented here do not carry over directly to the frequency dependent case, but they actually can be generalized — as shown by Gibianksy and Milton (1993), Milton and Berryman (1997), and Gibiansky et al. (1999).

Another class of effective medium theories studied by Hill (1965), Budiansky (1965), Wu (1966), Walpole (1969), and Boucher (1974) does *not* yield the same results as the present one, except for the case of spherical inclusions. It has been shown elsewhere [Berryman (1980b)] how the derivation of the approach of Hill, Budiansky, and others can be kinds of symmetrized to yield the symmetrical results as presented here that I prefer. Since the CPA class of effective medium theories gives results equivalent to the Hashin-Shtrikman [Hashin and Shtrikman (1961, 1962, 1963)] bounds when the inclusions are disk-shaped, I conclude that these results are preferred – since they do satisfy these bounding constraints, while the alternatives do not. The numerical results show general satisfaction of the bounds.

To elucidate somewhat further the relationship between the static and dynamic derivations of the effective medium results, I will outline the static derivation next. The integral equations for the static strain field are given by

$$\epsilon_{ij}(\vec{x}) = \epsilon^0_{ij}(\vec{x}) + \int d^3x' G_{ijkl}(\vec{x},\vec{x}')\Delta c_{klmn}(\vec{x}')\epsilon_{mn}(\vec{x}'), \tag{43}$$

where Green's function is

$$G_{ijkl}(\vec{x},\vec{x}') = \frac{1}{2}\left(g^0_{ik,jl} + g^0_{jk,il}\right), \tag{44}$$

with the Kelvin solution given by

$$g_{pq}(\vec{x},\vec{x}') = \frac{1}{4\pi\mu_m}\left[\frac{\delta_{pq}}{r} - \frac{1}{4(1-\nu_m)}\frac{\partial^2 r}{\partial x_p \partial x_q}\right], \tag{45}$$

where $r = |\vec{x} - \vec{x}'|$ and $\mu_m$ and $\nu_m$ are, respectively, the shear modulus and Poisson's ratio of the matrix material. Equation (43) may be rewritten formally as

$$\epsilon = \epsilon^0 + G\Delta c\epsilon, \tag{46}$$

where $G$ is now an integral operator defined by

$$Gf = \int d^3x' G(\vec{x},\vec{x}')f(\vec{x}'). \tag{47}$$

Iterating Equation (46), I obtain the well-known Born series

$$\epsilon = \epsilon^0 + G\Delta c\epsilon^0 + G\Delta cG\Delta c\epsilon^0 + \ldots, \tag{48}$$

and then summing the Born series formally yields

$$\epsilon = (I + Gt)\,\epsilon^0 = (I - G\Delta c)^{-1}\,\epsilon^0, \tag{49}$$

where the so-called $t$-matrix is defined by

$$t = \Delta c\,(I - G\Delta c)^{-1} = \Delta c\,(I + Gt)\,. \tag{50}$$

Taking the ensemble average of Equation (49), I have

$$\langle\epsilon\rangle = (I + G\,\langle t\rangle)\,\epsilon^0 = \left\langle (I - G\Delta c)^{-1}\right\rangle \epsilon^0. \tag{51}$$

For a single scatterer, Equation (49) is equivalent to Equation (10). Therefore, it is worth noting that Wu's (1966) tensor $T$ is formally related to the $t$-matrix by

$$T = I + Gt = (I - G\Delta c)^{-1}\,. \tag{52}$$

Equation (51) is now in a convenient form for use in determining the effective elastic tensor $c^*$ of a composite defined by

$$\langle\sigma\rangle = \langle c\,\epsilon\rangle \equiv c^*\,\langle\epsilon\rangle\,, \tag{53}$$

where the averages in Equation (53) are again ensemble averages over possible composites having similar physical and statistical properties. Using the standard definition $c = c^m + \Delta c$, I find that

$$\langle c\,\epsilon\rangle = c^m\,\langle\epsilon\rangle + \langle\Delta c\,\epsilon\rangle = c^m\,\langle\epsilon\rangle + \langle t\rangle\,\epsilon^0. \tag{54}$$

From Equation (54), it follows easily that the effective elastic tensor is given by

$$c^* = c^m + \langle t\rangle\,(I + G\,\langle t\rangle)^{-1}\,. \tag{55}$$

The choice of matrix elastic tensor $c^m$ is still completely free since the decomposition $c = c^m + \Delta c$ is not unique. Thus, I am free to choose, for example, $c^m = c^*$ (*i.e.*, the matrix material has now exactly the properties of the equivalent composite material), which implies:

$$\langle t\rangle \equiv 0. \tag{56}$$

Equation (56) is an implicit formula determining the effective elastic tensor $c^*$, and says that the effective scattering $t$-matrix averages to zero.

In principle, Equation (56) provides an exact solution for the effective moduli. However, the total $t$-matrix itself is generally too difficult to calculate. It turns out to be more reasonable and more effective [Velicky et al. (1968)] to rearrange the terms of the total $t$-matrix into a series of terms with repeated scattering from individual scatterers ($t_i$). Then, by setting the ensemble average of the individual $t$ matrices to zero

$$\langle t_i\rangle = \sum_{i=1}^{N} f_i\Delta c_i\,(I - G\Delta c_i)^{-1} = 0, \tag{57}$$

and neglecting terms corresponding to fluctuations in the scattered wave [Velicky et al. (1968)], a tractable approximation for the estimate of the elastic moduli is obtained.

When the constituents and the composite as a whole are all relatively homogeneous and isotropic, the tensor Equation (57) reduces to two coupled equations:

$$\sum_{i=1}^{N} f_i(K_i - K^*)P^{*i} = 0, \tag{58}$$

and

$$\sum_{i=1}^{N} f_i(\mu_i - \mu^*)Q^{*i} = 0, \tag{59}$$

where Equations (13), (14), and (52) were used to simplify Equation (57). Note that Equations (58) and (59) are identical to Equations (18) and (19), thereby establishing the equivalence of these two approaches in the isotropic case.

## SUMMARY AND CONCLUSIONS

I conclude that my preferred choice of effective medium theory (the CPA) satisfies all the known constraints on a viable theory: (a) it gives correct values and slopes for both large and small volume fractions of inclusions; (b) numerical evidence indicates that the results always satisfy the Hashin-Shtrikman bounds, the Beran-Molyneux-Miller bounds, and the McCoy-Silnutzer bounds; (c) the theory is also known [Berryman (1980b)] to reproduce Hill's exact result [Hill (1963)] for composites with uniform shear modulus — which fact is a fairly simple exercise to check, so that the reader might find it instructive to carry this through.

The single-scatterer theory is designed to minimize multiple scattering effects while yielding formulas that are relatively easy to use. Nevertheless, the theory is not exact, and some potentially significant effects have been neglected. The neglected terms become more important for propagation of higher frequency elastic waves. But it is important to note that bounding methods and formulas are also much harder to implement rigorously for the frequency dependent (viscoelastic) case. This fact is surely one reason that the theory is seldom applied at significantly higher frequencies than typical seismic frequencies, or in regions of very much higher viscosity, and wave dissipation and dispersion. So, it is expected that, for small ranges of frequency — and especially those that are pertinent to exploration seismology — will naturally be included in the range of useful applications since the seismic band is fairly narrow. Then the viscoelastic effects can typically be treated without great additional difficulty. Some future efforts should nevertheless be directed towards extending this effective medium theory to scattering from clusters of inclusions at finite frequency — thereby including within the expanded theory more of the important scattering effects discussed (but then specifically neglected, and therefore not treated in any detail) here.

## ACKNOWLEDGMENTS

I thank P. A. Berge, G. M. Mavko, G. W. Milton, and R. W. Zimmerman for helpful conversations.

# REFERENCES

Beran, M. J. and J. Molyneux, 1966, Use of classical variational principles to determine bounds for the effective bulk modulus in heterogeneous media: Quarterly Journal of Applied Mathematics, **24**, 107–118.

Berryman, J. G., 1979, Theory of elastic properties of composite materials: Applied Physics Letters, **35**, 856–858.

———, 1980a, Long-wavelength propagation in composite elastic media. I. Spherical inclusions: Journal of the Acoustical Society of America, **68**, 1809–1819.

———, 1980b, Long-wavelength propagation in composite elastic media. II. Ellipsoidal inclusions: Journal of the Acoustical Society of America, **68**, 1820–1831.

———, 1995, Mixture theories for rock properties, *in* Ahrens, T. J., ed., Rock Physics and Phase Relations, 205–228, American Geophysical Union.

Boucher, S., 1974, On the effective moduli of isotropic two-phase elastic composites: Journal of Composite Materials, **8**, 82–89.

Budiansky, B., 1965, On the elastic moduli of some heterogeneous materials: Journal of the Mechanics and Physics of Solids, **13**, 223–227.

Elliott, R. J., J. A. Krumhansl, and P. L. Leath, 1974, The theory and properties of randomly disordered crystals and related physical systems: Reviews of Modern Physics, **46**, 465–543.

Eshelby, J. D., 1957, The determination of the elastic field of an ellipsoidal inclusion, and related problems: Proceedings of the Royal Society of London A, **241**, 376–396.

Gibianksy, L. V. and G. W. Milton, 1993, On effective viscoelastic moduli of two-phase media. I. Rigorous bounds on the complex bulk modulus in three dimensions: Proceedings of the Royal Society of London A, **241**, 376–396.

Gibiansky, L. V., G. W. Milton, and J. G. Berryman, 1999, On effective viscoelastic moduli of two-phase media. III. Rigorous bounds on the complex shear modulus in two dimensions: Proceedings of the Royal Society of London A, **455**, 2117–2149.

Gubernatis, J. E. and J. A. Krumhansl, 1975, Macroscopic engineering properties of polycrystalline materials: Elastic properties: Journal of Applied Physics, **46**, 1875–1883.

Hashin, Z. and S. Shtrikman, 1961, Note on a variational approach to the theory of composite elastic materials: Journal of the Franklin Institute, **271**, 336–341.

———, 1962, On some variational principles in anisotropic and nonhomogeneous elasticity: Journal of the Mechanics and Physics of Solids, **10**, 335–342.

———, 1963, A variational approach to the theory of the elastic behaviour of multiphase materials: Journal of the Mechanics and Physics of Solids, **11**, 127–140.

Hill, R., 1952, The elastic behaviour of a crystalline aggregate: Proceedings of the Physical Society of London A, **65**, 349–354.

———, 1963, Elastic properties of reinforced solids: Some theoretical principles: Journal of the Mechanics and Physics of Solids, **11**, 357–372.

———, 1965, A self-consistent mechanics of composite materials: Journal of the Mechanics and Physics of Solids, **13**, 213–222.

Korringa, J., 1973, Theory of elastic constants of heterogeneous media: Journal of Mathematical Physics, **14**, 509–513.

Korringa, J., R. J. S. Brown, D. D. Thompson, and R. J. Runge, 1979, Self-consistent imbedding and the ellipsoidal model for porous rocks: Journal of Geophysical Research, **84**, 5591–5598.

Mal, A. K. and L. Knopoff, 1967, Elastic wave velocities in two-component systems: Journal of the Institute of Mathematics and Its Applications, **3**, 376–387.

McCoy, J. J., 1970, On the displacement field in an elastic medium with random variation of material properties, *in* Eringen, A. C., ed., Recent Advances in Engineering Science, volume **5**, 235–254, Gordon and Breach.

Miller, M. N., 1969a, Bounds for effective bulk modulus of heterogeneous materials: Journal of Mathematical Physics, **10**, 2005–2013.

———, 1969b, Bounds for effective electrical, thermal, and magnetic properties of heterogeneous materials: Journal of Mathematical Physics, **10**, 1988–2004.

Milton, G. W., 1981, Bounds on the electromagnetic, elastic, and other properties of two-component composites: Physical Review Letters, **46**, 542–545.

Milton, G. W. and J. G. Berryman, 1997, On effective viscoelastic moduli of two-phase media. II. Rigorous bounds on the complex shear modulus in three dimensions: Proceedings of the Royal Society of London A, **453**, 1849–1880.

Silnutzer, N. R., 1972, Effective constants of statistically homogeneous materials: PhD thesis, University of Pennsylvania. (283pp).

Soven, P., 1967, Coherent-potential model of substitutional disordered alloys: Physical Review, **156**, 809–813.

Velicky, B., S. Kirkpatrick, and H. Ehrenreich, 1968, Single-site approximations in the electronic theory of simple binary alloys: Physical Review, **175**, 747–766.

Walpole, L. J., 1969, On the overall elastic moduli of composite materials: Journal of the Mechanics and Physics of Solids, **17**, 235–251.

Watt, J. P., G. F. Davies, and R. J. O'Connell, 1976, Elastic properties of composite materials: Reviews of Geophysics and Space Physics, **14**, 541–563.

Willis, J. R., 1977, Bounds and self-consistent estimates for the overall properties of anisotropic composites: Journal of the Mechanics and Physics of Solids, **25**, 185–202.

———, 1981, Variational and related methods for the overall properties of composites, *in* Advances in Applied Mechanics, 1–78, Academic Press.

Wu, T. T., 1966, The effect of inclusion shape on the elastic moduli of a two-phase material: International Journal of Solids and Structures, **2**, 1–8.

Zeller, R. and P. H. Dederichs, 1973, Elastic constants of polycrystals: Physica Status Solidi, **B55**, 831–842.

# Inversion of up and down going signal for ocean bottom data

*Mandy Wong, Biondo L. Biondi, and Shuki Ronen*

## ABSTRACT

We formulate an inversion problem using the up- and down-going signals of ocean bottom data to imaging primaries and multiples. The method involves separating pressure ($P$) and vertical particle velocity ($V_z$) data into up- and down-going components. Afterward, the up- and down-going data can be used for inversion with appropiate modeling operators. To a first-order accuracy, we use mirror imaging to define the up- and down-going modeling operator. A complete modeling scheme can be defined by the composition of the over-under modeling operator and the up-down decomposition operator. This scheme effectively models all primaries, water reveberations and multiples.

## INTRODUCTION

Traditionally, multiples in seismic surveying are considered as noise to be removed because most migration algorithims do not account for multiples. Recently, there have been efforts to use multiples as signals. For example, Berkhout and Verschuur (2003) and Guitton (2002) image the multiples with shot-profile migration while Shan (2003) transforms multiples into pseudo-primaries by cross-correlation in the souce-receiver domain.

One motivation to image with multiples is that multiples can provide subsurface information not found in primaries. The angular and spatial ranges covered by multiples are different than that of primaries (Figure 1). Ronen et al. (2005); Guimaraes et al. (1998) propose a mirror imaging technique that takes advantage of this property for ocean bottom and vertical cable acquisition geometry, respectively. By using receiver ghosts as signals, mirror imaging provides a much wider aperture in the image space, given the same set of data. While mirror imaging correctly images multiples by using down-going receiver ghost at the ocean bottom, the primary signal is imaged separately. On the other hand, Brown (2004); Brown and Guitton (2005) proposed joint imaging between primaries and multiples by using least-square inversion. One advantage of joint inversion is that both the primary and multiple signals are used.

For the case of ocean bottom data, signals can be separated into up- and down-going parts. Traditioanlly, up- and down-going signals are used for de-ghosting (Canales and Bell, 1996). Muijs et al. (2007) use this property to formulate prestack depth migration of primary and surface-related multiple using downward continuation. In this study, we carry on the work of Brown (2004) and discuss the theory of joint imaging of multiples and primaries using up- and down-going signals of ocean bottom data.

We will first discuss the geometry of ocean bottom data acquisition. Next, we will consider the techniques available to separate data into up-going and down-going wavefields. We will discuss the theory of joint imaging of multiples and primaries using up- and down-going

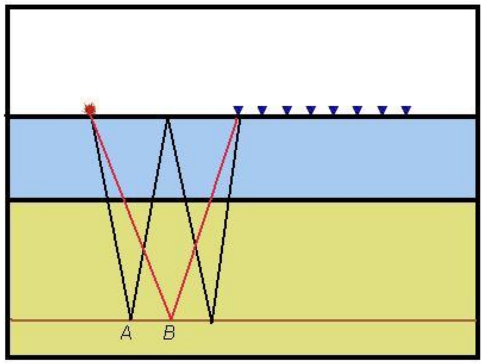


Figure 1: The angular and spatial ranges covered by multiples is different than that of primary. Near the edge of the receiver array, A can only be illuminated by multiple as primary can only illuminate up to a spatial extend of B.[**NR**] mandy2/. Fig1

ocean bottom data. Such an inversion scheme requires a good modeling operator for up- and down-going signals. To a first order accuracy and flat water bottom, we use mirror imaging to define the two modeling operators. A more accurate modeling scheme requires an over-under modeling and an up-down decomposition operator. This scheme effectively images all primaries, water reveberations, and multiples. It also reduces crosstalk leakage between up- and down-going signals and hence reduces incorrectly placed reflectors. The testing of this thoery would be the focus of my research for the next quarter.

## SEISMIC ACQUISITION OF OCEAN BOTTOM DATA

Although conventional streamers acquisition is well developed in terms of technology and processing techniques, it has significant limitations. For example, in obstructed oil fields, working with streamers could be difficult. In addition, streamers are more prone to drift and to be affected by weather conditon, which may compromise repeatability in time-lapse reservoir monitoring (4D). These limitations bring out a growing demand for ocean bottom seismometers (OBS). OBS data acquisition is an alternate approach in which seismometers are placed at the ocean bottom and shots are fired at the ocean surface.

OBS data acquisition can be done with either ocean bottom cables (OBC) or with nodes (OBN). Because the OBS method uses geophones and hydrophones, it can measure both compressional and shear waves. This capability permits separating up- and down-going waves at the seabed and therefore provides good opportunities for imaging with multiples.

Multi-component streamers (Pharez et al., 2008) also permit separation of up-going and down-going waves, but because the data are recorded near the sea surface, imaging with multiples is much more limited than with OBS in deep water.

To understand the events represented by the up-going an down-going signals, consider Figure 2. For ocean bottom data acquisition, down-going events include direct arrival, receiver ghosts, and higher-order pegleg multiples. On the other hand, up-going events include primaries, and pegleg multiples. Since the kinematics of these events are quite distinct for up-going and down-going signals, an inverse problem can be formulated to fit the up and down signals jointly. Before formulating the inverse problem, we will dicuss methods for separating receiver signals into their up- and down-going parts.

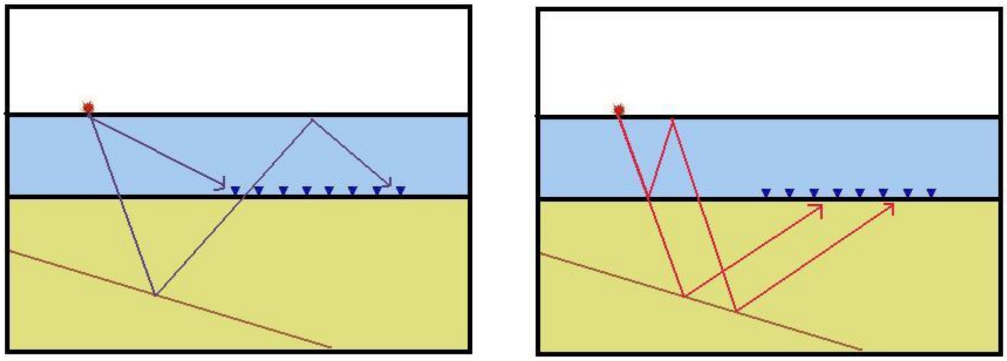


Figure 2: Left: The down-going signal consists of events such as direct arrival, receiver ghosts, and higher-order pegleg multiples. Right: The up-going signal consists of events such as primaries and other pegleg multiples. [**NR**] mandy2/. Fig2

## SEPARATION OF UP- AND DOWN-GOING WAVEFIELD

There are two well developed techniques to separate recorded signal into up- and down-going waves. For multi-component data, such separation can be done with the pressure and particle velocity recordings. Another way is to have two sets of hydrophones with one set on top of the other (an over-under arrangement). Although the over-under arrangement is used near the ocean's surface, it is an easy way to obtain up- and down-going wavefields for synthetic data examples.

### Separation using pressure and particle velocity recordings

The basic idea of up-down separation using pressure and vertical particle velocity is quite simple. Hydrophones measure compressional waves ($P$) regardless of their direction. Ocean bottom seismometers measure vertical particle velocity ($V_z$) that depends on the direction of the waves measured. Figure 3 illustrates the measurement of a positive pulse coming from above and from below.

Since the polarity of the $P$ and $V_z$ signal is the same for up-going waves and opposite for down-going waves, one can decompose the $P$ and $V_z$ measurements into up-going ($U$) and down-going ($D$) pressure components:

$$\begin{aligned} P(z_r) &= \left[U(z_r) + D(z_r)\right], \\ V_z(z_r) &= \left[U(z_r) - D(z_r)\right]/I, \end{aligned} \tag{1}$$

where $z_r$ is the receiver depth and $I$ is an impedance factor that scales vertical velocity value to pressure value. The impedance can be offset, frequency, wavenumber, or density

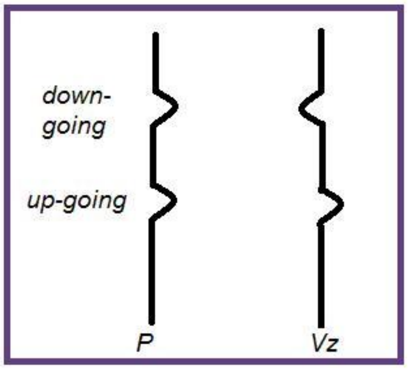


Figure 3: This Figure illustrates pressure **P** and verticle particle velocity $V_z$ measurement of a positive pulse coming from above and from below. Down-going events have opposite polarity while up-going events have the same polarity. [**NR**] mandy2/. pzfig

dependent depending on the method used. One way to perform PZ summation is in the Fourier $(\omega - k_x - k_y)$domain as

$$\begin{aligned} U(z_r) &= \frac{1}{2}\left[P(z_r) - \frac{\rho\omega}{k_z} V_z(z_r)\right], \\ D(z_r) &= \frac{1}{2}\left[P(z_r) + \frac{\rho\omega}{k_z} V_z(z_r)\right], \end{aligned} \tag{2}$$

where $\omega$ is frequency in time. $k_z = \sqrt{\frac{w^2}{v^2} - k_x^2 - k_y^2}$ is the vertical wavenumber calculated from horizontal wave numbers $k_x$ and $k_y$. For a complete derivation of equation 2, please refer to Amundsen (1993).

## Separation using over-under recordings

The derivation for decomposing over-under pressure waves into up-going and down-going signals is best done in the Fourier domain. Denote $S_1(\omega, k_x)$ and $S_2(\omega, k_x)$ to be the Fourier transformed measurement of compressional waves at depth $z_1$ (over) and $z_2$ (under). Theoretically, $S_1(\omega, k_x)$ can be viewed as a sum of the up-going $U_1(\omega, k_x)$ and down-going $D_1(\omega, k_x)$ components. Likewise for $S_2(\omega, k_x)$:

$$\begin{aligned} S_1(\omega, k_x) &= U_1(\omega, k_x) + D_1(\omega, k_x), \\ S_2(\omega, k_x) &= U_2(\omega, k_x) + D_2(\omega, k_x). \end{aligned} \tag{3}$$

Down-going waves visit the over array ($D_1$) before visiting the under array ($D_2$). Therefore, $D_1$, when shifted forward in time, would match the signal $D_2$. Similarily, up-going waves visit the under array first. Therefore, $U_2$, when shifted forward in time would match the signal $U_1$. This relationship is equivalent to a phase-shift in the Fourier domain:

$$\begin{aligned} e^{ik_z\Delta z}D_1 &= D_2, \\ U_1 &= e^{ik_z\Delta z}U_2, \end{aligned} \tag{4}$$

where $\Delta z = z_2 - z_1$ and $k_z$ is the usual dispersion relation. Finally, substituting equation 4 into 3 yields the formula for the up-going and down-going waves at the receivers:

$$\begin{aligned} U_2 &= \frac{S_2 - e^{ik_z\Delta z}S_1}{1 - e^{2ik_z\Delta z}}, \\ D_2 &= \frac{e^{ik_z\Delta z}S_1 - e^{2ik_z\Delta z}S_2}{1 - e^{2ik_z\Delta z}}. \end{aligned}$$

Data acquisition using over-under arrangement is often used to elimate receiver ghosts and water reverberation. For a thorough review of this method, please see Sonneland et al. (1986). Although over-under arrays are rarely placed at the sea floor in real seismic surveying, this technique allows easy generation of up- and down-going data at the sea bottom in synthetic examples using the simpler acoustic wave equation.

## THE INVERSE PROBLEM FOR IMAGING MULTIPLES USING UP- AND DOWN-GOING DATA

The inverse problem for imaging multiples using up- and down-going data can be fomulated as follow. We first break down the recorded data as the superposition of up- $\mathbf{d}_\uparrow$ and down-going $\mathbf{d}_\downarrow$ signals at the receivers. This can be done by using PZ data to give up-going and down-going data as discussed in the PZ summation section above.

$$\begin{bmatrix} \mathbf{d}_\uparrow \\ \mathbf{d}_\downarrow \end{bmatrix} = \mathbf{S}_{pz} \begin{bmatrix} \mathbf{d}_p \\ \mathbf{d}_z \end{bmatrix}$$

The above construction assumes that the vertical particle velocity $d_z$ contains mostly pressure ($\mathbf{P}$) waves. A pre-processing step can be included into $\mathbf{S}_{pz}$ to separate the $\mathbf{P}-$ and converted $\mathbf{S}-$wave arrivals (Helbig and Mesdag, 1982; Dankbaar, 1985). Next, we denote the modelling operator for up-going signals at the receivers as $\mathbf{L}_\uparrow$. Similarily, denote the modeling operator for down-going signals at the receivers as $\mathbf{L}_\downarrow$. The two modeling operators provide the up- and down-going modeled data, denoted as $\mathbf{d}_\uparrow^{mod}$ and $\mathbf{d}_\downarrow^{mod}$;

$$\begin{aligned} \mathbf{d}_\uparrow^{mod} &= \mathbf{L}_\uparrow\mathbf{m}, \\ \mathbf{d}_\downarrow^{mod} &= \mathbf{L}_\downarrow\mathbf{m}. \end{aligned} \tag{5}$$

The inverse problem is defined as minimizing the $L_2$ norm of the two data residuals $\mathbf{r}_\uparrow$ and $\mathbf{r}_\downarrow$ , with respect to a single model $\mathbf{m}$. The data residuals are defined as the difference between the recorded data and the modelled data,

$$\begin{aligned}\mathbf{r}_{\uparrow} &= \mathbf{d}_{\uparrow} - \mathbf{d}_{\uparrow}^{mod} = \mathbf{d}_{\uparrow} - \mathbf{L}_{\uparrow}\mathbf{m}, \\ \mathbf{r}_{\downarrow} &= \mathbf{d}_{\downarrow} - \mathbf{d}_{\downarrow}^{mod} = \mathbf{d}_{\downarrow} - \mathbf{L}_{\downarrow}\mathbf{m},\end{aligned} \tag{6}$$

$$min\left(\|\mathbf{L}_{\uparrow}\mathbf{m} - \mathbf{d}_{\uparrow}\|_2^2 + \|\mathbf{L}_{\downarrow}\mathbf{m} - \mathbf{d}_{\downarrow}\|_2^2\right), \tag{7}$$

where $\|.\|_2$ represents the $L_2$ norm. In matrix form, the fitting goal can be written as

$$0 \approx \begin{bmatrix} \mathbf{L}_{\uparrow} \\ \mathbf{L}_{\downarrow} \end{bmatrix} \mathbf{m} - \begin{bmatrix} \mathbf{d}_{\uparrow} \\ \mathbf{d}_{\downarrow} \end{bmatrix}.$$

With the conjugate gradient method, the model update $\Delta\mathbf{m}$ at each iteration has contributions from both the up-going and down-going parts of the inversion;

$$\Delta\mathbf{m} = \mathbf{L}_{\uparrow}'\mathbf{r}_{\uparrow} + \mathbf{L}_{\downarrow}'\mathbf{r}_{\downarrow}. \tag{8}$$

where $\mathbf{r}_{\uparrow} = \mathbf{L}_{\uparrow}\mathbf{m} - \mathbf{d}_{\uparrow}$ and $\mathbf{r}_{\downarrow} = \mathbf{L}_{\downarrow}\mathbf{m} - \mathbf{d}_{\downarrow}$ is the up- and down-going part of the residual, respectively.

The justification for this inverse problem is to reduce the wrong placement of image point with the use of both $\mathbf{d}_{\uparrow}$ and $\mathbf{d}_{\downarrow}$ signals. Traditional migration scheme only uses $d_{\uparrow}$ to determine the image since all primary signal can be found in $\mathbf{d}_{\uparrow}$. However, migration of primaries can give incorrect image point as well. In Figure 4, a primary event with a given travel time can indicate a correct image point at A and an incorrect image point at B. If we include the previously ignored information $\mathbf{d}_{\downarrow}$ into a joint inversion, some wrongly placed image point can be refuted. Joint imaging allow us to use both primaries and multiples to estimate the image. This can be a distinct advantage because multiples and primaries illuminate different parts of the sub-surface. For ocean bottom data with sparse receiver spacing, multiples illuminate more than primaries.

The quality of the inverse problem would depend on the implemetation of the modeling operator. The next section will discuss how to approximate the modeling operator.

## MODELLING OPERATORS

To implement for a modeling operator that maps the image only into up-going signal $d_{\uparrow}$ or only into down-going signal $d_{\downarrow}$ at the receiver, we can use wave equation extrapolation, choosing either an one-way or a two-way wave equation. We can define the modeling operator with different levels of accuracy. As a first-order of accuracy, we can use the mirror imaging operator to model the down-going wave as described in the next section.

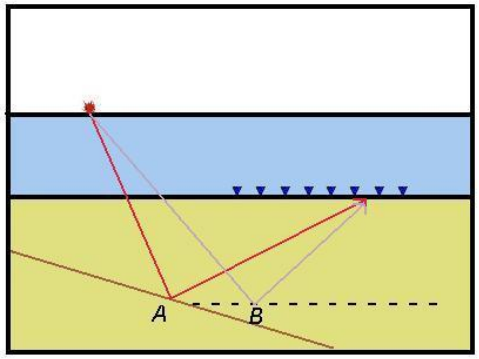


Figure 4: An up-going signal with a given travel time can indicate a correctly placed reflector at A and an incorrectly placed reflector at B. Reflector A will be supported by down-going data while reflector B will be refuted. [**NR**] mandy2/. whyupdown

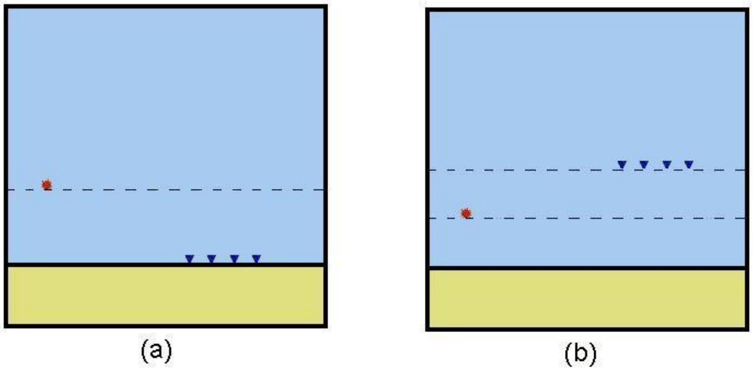


Figure 5: (a) shows the set-up of a first-order modelling operator for the up-going signal. The air-water interface is removed and filled with a half space of water. (b) shows the set-up of a first-order modeling operator for down-going signal. The air-water interface is removed and the receivers is elevated to twice the water depth. [**NR**] mandy2/. Lmod

## Mirror Imaging modeling opeators

The first-order implementation of $\mathbf{L}_\uparrow$, denoted $\mathbf{L}_1^\uparrow$ performs wave equation modeling in a model space without the sea surface as shown in Figure 5 (a). Our up-going modeling operator is now approximated as

$$\mathbf{L}_\uparrow \approx \mathbf{L}_1^\uparrow. \tag{9}$$

When the sea surface is removed, waves always return to the receiver going upward. There are obvious limitations to $\mathbf{L}_1^\uparrow$. It can only model primaries and internal multiples. Higher order water reverberations are excluded. Figure 6 shows some events that are captured and excluded by $\mathbf{L}_1^\uparrow$.

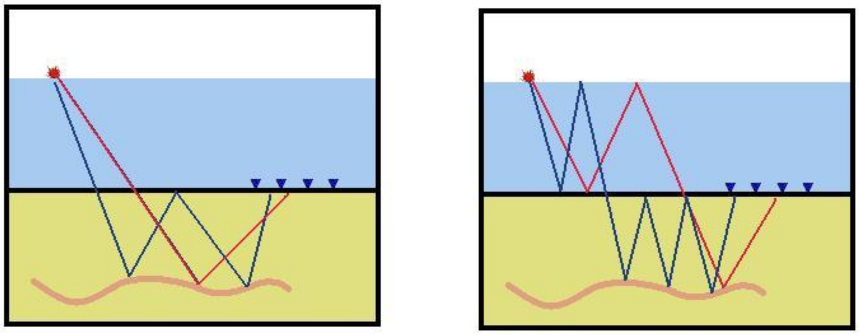


Figure 6: Left: The first-order up-going operator captures all the primaries and internal multiples. Right: It does not capture higher-order water reverberations and any pegleg multiples with a bounce at the sea surface. [**NR**] mandy2/. upmod

On the other hand, we can use the mirror imaging modeling operator to get a first-order estimate of down-going signals, denoted $\mathbf{L}_1^\downarrow$. This operator performs wave equation modeling by raising the receivers to twice the water depth level as shown in Figure 5 (b). Our down-going modeling operator is now approximated as

$$\mathbf{L}_\downarrow \approx \mathbf{L}_1^\downarrow. \tag{10}$$

The limitation of $\mathbf{L}_1^\downarrow$ is that it can only image direct arrival and pegleg multiples having only one bounce from the sea surface. Higher-order water reverberations and pegleg multiples with two or more bounces from the sea surface are excluded. Figure 7 shows some events that are captured by $\mathbf{L}_1^\downarrow$. To do mirror imaging, we had to assume that the sea bottom is flat.

## Complete modeling

To model beyond primaries and first order receiver pegleg multiples, one can use the separation operator that maps over-under data to up-going and down-going data,

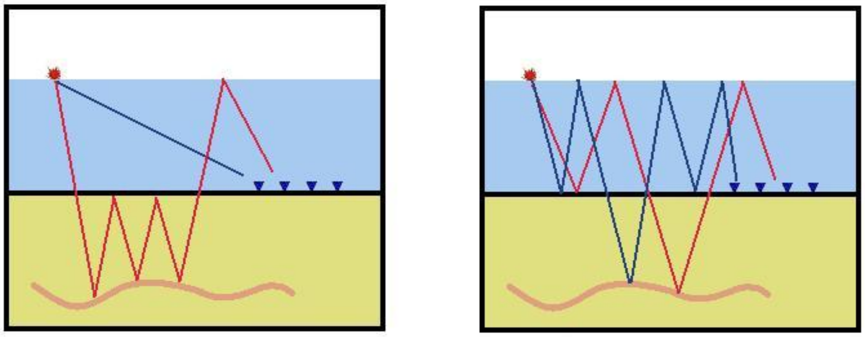


Figure 7: Left: The first-order down-going operator captures the direct arrival and pegleg multiples with only one reflection from the sea surface. Right: It does not capture higher-order water reverberation and any pegleg multiples with two or more bounces at the sea surface. [**NR**] mandy2/. downmod

$$\begin{bmatrix} \mathbf{L}_{\uparrow} \\ \mathbf{L}_{\downarrow} \end{bmatrix} = \mathbf{S}_{ou}\mathbf{A},$$

where **A** represents the wave equation forward modelling operator that generates over and under signals. $\mathbf{S}_{ou}$ is a separation operator that extracts the up- and down-going signals from over-under data. In matrix form, the inversion scheme has the following fitting goal:

$$0 \approx \begin{bmatrix} \mathbf{L}_{\uparrow} \\ \mathbf{L}_{\downarrow} \end{bmatrix} \mathbf{m} - \begin{bmatrix} \mathbf{d}_{\uparrow} \\ \mathbf{d}_{\downarrow} \end{bmatrix} = \mathbf{S}_{ou}\mathbf{A}\mathbf{m} - \mathbf{S}_{pz} \begin{bmatrix} \mathbf{d}_{p} \\ \mathbf{d}_{z} \end{bmatrix},$$

where $\mathbf{S}_{pz}$ is a separation operator that extracts the up- and down-going signals from PZ data. Note that $\mathbf{d}_{\uparrow}$ and $\mathbf{d}_{\downarrow}$ can be viewed as processed data from the orignal recorded $\mathbf{d}_p$ and $\mathbf{d}_Z$. The advantage of this joint modeling is that we are now imaging all multiples event that return to the ocean bottom receivers going upward or downward.

For our complete modeling operator, $\mathbf{L}_{\uparrow}$ and $\mathbf{L}_{\downarrow}$, there is an alternate way to interpret the inverson problem we have set-up from above. Consider an equivalent fitting goal below,

$$0 \approx \mathbf{S}_{pz}^{-1}\mathbf{S}_{ou}\mathbf{A}\mathbf{m} - \begin{bmatrix} \mathbf{d}_{p} \\ \mathbf{d}_{z} \end{bmatrix}.$$

The above fitting goal converts our model into over-under data. Afterward, over-under data are separated into up-going and down-going data. Finally, the up-going and down-going data are converted into PZ data using $\mathbf{S}_{pz}^{-1}$. Therefore, this inversion scheme can be interpreted as fitting both P and Z data using only acoustic equation.

## SUMMARY

We have discussed the theory of joint imaging of multiples and primaries using up- and down-going ocean bottom data. To a first order accuracy, we can use mirror imaging to define $\mathbf{L}_{\uparrow}$ and $\mathbf{L}_{\downarrow}$. A complete modeling scheme first models over-under data and then decompose them into up- and down-going data. This scheme effective uses water reveberations and multiples as signal instead of noise. It also reduces crosstalk leakage between up- and down-goign signals and hence reduces incorrectly placed reflectors. The testing of this thoery would be the focus of my research for the next quarter.

## REFERENCES

Amundsen, L., 1993, Wavenumber-based filtering of marine point-source data: Geophysics, **58**, 1497–150.

Berkhout, A. J. and D. J. Verschuur, 2003, Transformation of multiples into primary reflections: 69th Annual International Meeting, 1497–1500.

Brown, M. P., 2004, Least-squares joint imaging of multiples and primaries: PhD thesis, Stanford University.

Brown, M. P. and A. Guitton, 2005, Least-squares joint imaging of multiples and primaries: Geophysics, **70**, S79–S89.

Canales, L. L. and M. L. Bell, 1996, Ghost attenuation using dual sensor cable data: SEG Expanded Abstract.

Dankbaar, J. W. M., 1985, Separation of p- and s- waves: Geophysical Prospecting, **33**, 970–986.

Guimaraes, M. G., K. K. Sekharan, D. Sukup, and P. Krail, 1998, 3-d prestack depth migration of vertical cable data over seg/eage physical model: SEG Expanded Abstract.

Guitton, A., 2002, Shot-profile migration of multiple reflections: 72nd Annual International Meeting, 1296–1299.

Helbig, K. and C. S. Mesdag, 1982, The potential of shear-wave observations: Geophysical Prospecting, **30**, 413–431.

Muijs, R., J. O. A. Robertsson, and K. Holliger, 2007, Prestack depth migration of primary and surface-related multiple reflections: Part i - imaging: Geophysics, **72**, S59–S69.

Pharez, S., N. Hendrick, and R. Tenghamn, 2008, First look at seismic data from a towed dual-sensor streamer: The Leading Edge, **27**, 904–907.

Ronen, S., L. Comeaux, and X. G. Miao, 2005, Imaging downgoing waves from ocean bottom stations: SEG Expanded Abstracts, **24**, 963.

Shan, G., 2003, Source-receiver migration of multiple reflections: SEP-Report, **113**, 75–83.

Sonneland, L., L. E. Berg, P. Eidsvig, A. Haugen, B. Fotland, and J. Vestby, 1986, 2-d deghosting using vertical receiver arrays: SEG Expanded Abstracts, **5**, 516.

# Performance of RTM with ODCIGs computation fully offloaded to GPU

*Abdullah Al Theyab and Robert G. Clapp*

## ABSTRACT

Nvidia's graphics processing units (GPU) powered with Compute Unified Device Architecture (CUDA), the supporting API, have allowed a significant speedup to finite difference time domain (FDTD) seismic modeling and, consequently, to reverse time migration (RTM). To utilize the power of GPUs for velocity analysis, we implemented kernels for generating offset-domain common image gathers (ODCIGs). With 4GB of memory, a single Tesla 10 series GPU can perform the 2D RTM with generation of ODCIGs. Computing the ODCIGs takes the majority of the algorithm execution time because of the large volume of output. We examine the performance of a general 2D RTM algorithm with ODCIGs computation fully offloaded to a single GPU device. We optimized the imaging kernel utilizing the available shared memory on the GPU to double the throughput of the kernel.

## INTRODUCTION

Reverse time migration (RTM) is a full wave equation imaging technique that constructs an image that best represents the subsurface structure. Seismic data are migrated using an estimate of the wave propagation velocity in the subsurface. Estimates of the velocity field can be inaccurate at the first imaging attempt, and subsurface offset gathers can provide a measure of the errors in velocity estimation (Biondi and Symes, 2004). In addition, they can give amplitude versus offset (AVO) information if amplitudes are handled properly. Algorithms for RTM and generation of ODCIGs are known to be computationally expensive and sometimes unaffordable. Therefore, an efficient implementation of RTM with ODCIGs generation algorithm is vital for minimizing the time required for creating a complete image.

Reverse time migration falls into the computational class of convolution with a stencil. The stencil computation workload can be divided among many processing units in an embarrassingly parallel fashion. However, the main performance limitation on modern computer architectures is the memory latency. Cache-aware algorithms minimize data traffic by taking advantage of spatial and temporal locality and/or the data prefetch capabilities of modern CPUs. Another way of hiding memory latency is to have more threads than cores to execute some threads while other threads are waiting for memory access. The performance gain given by this technique is not significant on CPUs because thread switching is expensive. This is not the case on GPUs, which can run a massive number of threads concurrently to hide memory latency. This capability makes GPUs very attractive hosts for stencil computational problems.

Micikevicius (2009) has shown an order of magnitude increase in performance for the GPU's FDTD kernel as compared to the kernel's performance on multi-core CPUs. The reported performance numbers are optimized for 3D by using equal grid spacing in all dimensions and without using the surface and absorbing boundary conditions.

Bus connection speed can be a performance bottleneck for mixed CPU-GPU processing. Generating subsurface offset gathers on the host using wavefields computed on the GPU will slow down the algorithm significantly. Fortunately, this can be avoided by fully offloading RTM and ODCIGs generation to GPUs.

In this report, we review briefly the theory of RTM for an acoustic medium. We detail the GPU implementation of RTM with generation of ODCIGs. We then analyze the performance of GPU kernels on a single device. Finally, we optimize the imaging kernel by using shared memory to double the throughput. The developed kernels were run on Tesla 10 series GPUs.

## GOVERNING EQUATIONS

Wave propagation in an acoustic medium is described by

$$\left(\nabla^2 - \frac{1}{v^2(\mathbf{x})}\frac{\partial^2}{\partial t^2}\right) P(\mathbf{x},t) = -f(\mathbf{x},t), \tag{1}$$

where $P$ is the pressure at a point $\mathbf{x}$ in the medium at time $t$, $v(\mathbf{x})$ is the wave propagation velocity field, and $f(\mathbf{x},t)$ is the source term. A seismic experiment is conducted in the field by exciting a seismic source $f(\mathbf{x_s},t)$ and recording the response at many receiver stations $(r_1, r_2, ...)$, with each receiver $r_i$ positioned at $\mathbf{x}_{r_i}$. The data from one experiment are collected into a *shot gather* $D_s(r_i,t)$. The experiment is repeated many times with different source locations to make a collection of shot gathers.

To build a subsurface image, each shot is migrated independently by simulating two wavefields using equation 1 with a zero source term. The first wavefield is the *forward* propagated wavefield $P_f(\mathbf{x},t)$, which is simulated using the boundary condition

$$P_f(\mathbf{x},t) = \delta(\mathbf{x}-\mathbf{x}_s)\int_0^t f_s(t')dt' \tag{2}$$

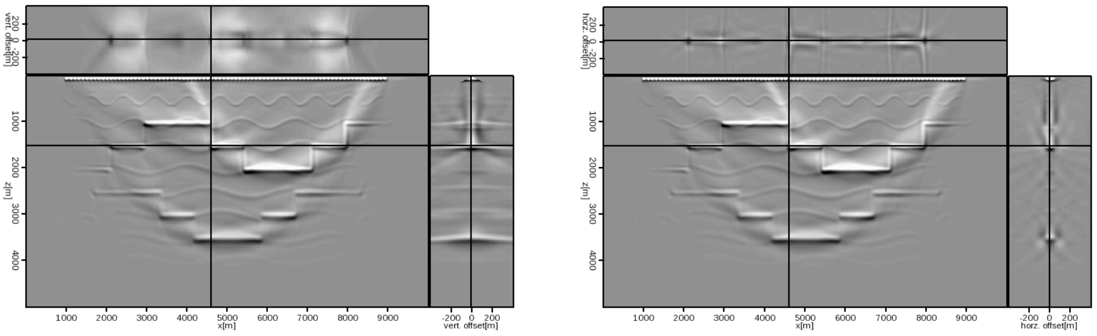
and a zero boundary condition above the Earth's surface (Zhang and Sun, 1993). Here, $f_s(t')$ is the source signature. The second wavefield is the *backward* propagated wavefield, computed using the upper boundary condition

$$P_b(\mathbf{x},t) = \sum_j \delta(\mathbf{x}-\mathbf{x}_{r_j})D_s(r_j, t_{max}-t). \tag{3}$$

The final image is computed using

$$I(\mathbf{x},\mathbf{h}) = \sum_s \sum_t P_f(\mathbf{x}+\mathbf{h}, t_{max}-t; s)P_b(\mathbf{x}-\mathbf{h}, t; s) \tag{4}$$

where $\mathbf{h}$ is the subsurface offset (Biondi and Symes, 2004). In practice, $\mathbf{h}$ is sampled in the horizontal direction, producing horizontal ODCIGs $\mathbf{I}(\mathbf{x}, h_x)$, and in the vertical direction to produce the vertical ODCIGs $\mathbf{I}(\mathbf{x}, h_z)$. For a 2D imaging problem, the horizontal and vertical ODCIGs are two 3D image volumes. Figures 1(a) and 1(b) show respectively slices through the vertical and horizontal subsurface offset cubes for a 2D synthetic dataset migrated using the correct velocity model.



(a) (b)

Figure 1: **(a)** Vertical ODCIGs. **(b)** Horizontal ODCIGs. The energy is focused at the zero-offset because the data were migrated using the correct velocity. [**CR**] altheyab1/. zgath,xgath

## CUDA PROGRAMMING MODEL

Compute Unified Device Architecture (CUDA) is the GPU supporting API extension to the C programming language. A GPU has many multiprocessors, each with its own set of stream processors and shared memory. A GPU also has a global DRAM memory, usually with a size of several gigabytes. This memory is uncached and memory latency is hidden by executing a massive number of threads concurrently. Threads are grouped together in thread blocks. Threads within a thread block can share data by using shared memory and can synchronize by using barriers. However, data sharing and synchronizing are not possible between thread blocks. Calling a kernel from the host code will launch the thread blocks in an unspecified order.

The number of processing units limits the maximum number of threads per thread block. Moreover, the fixed size of the fast shared memory and the number of registers limit the number of the thread blocks that run concurrently on a single multi-processor. For stencil computation, 2D thread blocks are mapped to the data grid, which means that each thread block handles a tile of grid points. The optimal thread block size is 16x16, which is determined by the device instruction set (NVIDIA, 2008) and the available shared memory.

Many considerations have to be taken into account when optimizing CUDA kernels. Coalesced global memory access can radically reduce memory access instructions. Shared memory usage can also significantly reduce global memory access. However, memory bank conflicts can hinder the performance gain from using the shared memory. Therefore kernels should be designed to avoid or reduce simultaneous access to the same memory banks by the threads in a thread block.

The kernels used for our implementation were run on a Tesla 10-series GPU (Tesla S1070). This GPU contains 30 multiprocessors, each with 8 streaming processors and 16 KB of shared memory. The memory bandwidth is 104 GB/s, and the global memory is 4GB in size.

## PERFORMANCE METRICS

The total cost of the 2D RTM with ODCIGs generation algorithm for a grid with the size $N_x \times N_z$ and $N_t$ time steps is

$$C_{\text{total}} = (C_{\text{forward}-\text{FDTD}} + C_{\text{backward}-\text{FDTD}}) N_x N_z N_t N_{shots} + C_{\text{Imaging}} N_I N_x N_z N_h N_{shots}, \quad (5)$$

where $C_{\text{forward}-\text{FDTD}}$, $C_{\text{backward}-\text{FDTD}}$, and $C_{\text{Imaging}}$ are the costs for forward wave propagation, backward wave propagation, and imaging, respectively. $N_I$ is the number of imaging steps, and $N_h$ is the number of points in the subsurface offset axis. FDTD cost includes the cost of setting the surface boundary condition. Because of the asynchronous execution of backward propagating kernels and imaging kernels, it is difficult to isolate their costs. We estimate that $C_{\text{forward}-\text{FDTD}} = C_{\text{backward}-\text{FDTD}}$, where $C_{\text{forward}-\text{FDTD}}$ can be calculated by timing the forward propagation part of the algorithm. Therefore the throughput metrics for wave propagation and imaging kernels are

$$\begin{aligned} C_{FD}^{-1} &= N_x N_z N_t / (\text{FDTD exec. time}), \\ C_{\text{Imaging}}^{-1} &= N_I N_x N_z N_h / (\text{Total exec. time} - 2 \times \text{FDTD exec. time}). \end{aligned}$$

Both are expressed in millions of output points per second (Mpts/sec).

## WAVE PROPAGATION KERNELS

The wave equation is solved by explicit finite differencing that is second order in time and eighth order in space, as expressed in

$$P^{\mathrm{t}+1} = v^2 \Delta t^2 \left( f^{\mathrm{t}} + \nabla^2 P^{\mathrm{t}} \right) + 2P^{\mathrm{t}} - P^{\mathrm{t}-1}. \quad (6)$$

We use an approach similar to the one implemented by Micikevicius (2009), but we generalize it to accommodate variable grid spacing for each dimension. We also have separate kernels for the source function injection. The waves incident on the grid boundaries are attenuated by adding an absorption term to the wave equation. This term is active around the neighborhood of the side and bottom boundaries. These inclusions to the algorithm increase the number of floating point operations and the number of memory accesses, which in turn reduce the throughput of FDTD by a few hundreds of Mpts/sec as compared to the reported performance measures by Micikevicius (2009).

The problem grid is divided into a grid of 16x16 blocks as illustrated by Figure 2. Each grid block is assigned to a thread block of the same size; i.e., each thread performs the finite differencing on a single point on the grid. Since data sharing is allowed between threads within a thread block, shared memory is used to keep local copies of $P^{\mathrm{t}}$ that are needed for computing the spatial derivatives. Each thread within a thread block loads the corresponding point on the grid of $P^{\mathrm{t}}$ into shared memory. Because the derivative stencil requires eight neighboring points in each spatial dimension, some of the threads are assigned to load the halos, i.e., the surrounding points of the thread block.

## IMAGING KERNEL

The imaging equation (4) is used in practice to generate two ODCIG cubes with the two x- and z-spatial dimensions and a third dimension along $h_x$ ($h_z$) for horizontal (vertical)

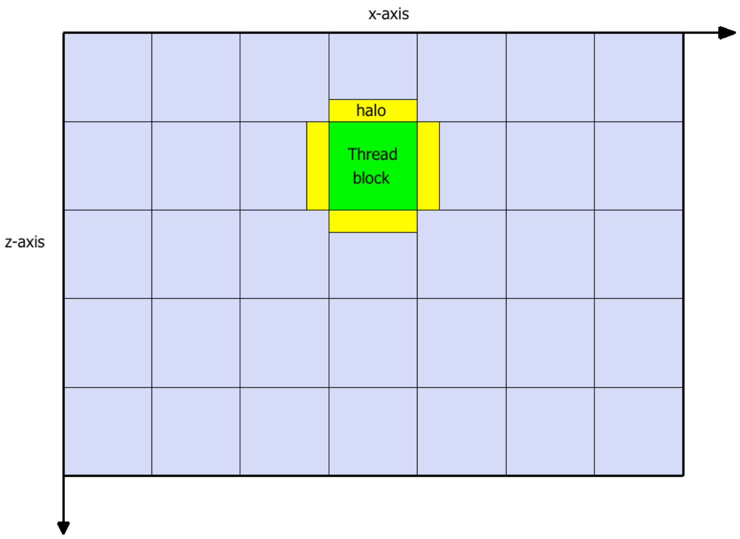


Figure 2: Thread blocks of the wave propagation kernel are mapped to areal blocks of the domain. A thread block has to read the assigned grid points (green area) and halos (yellow) from neighboring points to shared memory. [**NR**] altheyab1/. fdblocking-crop

subsurface offsets. The subsurface offset can have both positive and negative values. The following code snippet shows a simple kernel that computes $\mathbf{I}(\mathbf{x}, h_z)$:

```
__global__ void img_hz_kernel(float *p1 , float *p2){
  int i=blockIdx.x*blockDim.x+threadIdx.x;            /* z image location */
  int j=blockIdx.y*blockDim.y+threadIdx.y;            /* x image location */
  for(int h=hmin; h<hmax; h++)
    img_zh[iloc(i,j,h)]+=p1[loc(i-h,j)]*p2[loc(i+h,j)];/* imaging condition*/
}
```

Grid blocking of image locations $(z, x)$ is similar to the one shown for wave propagation. The iteration occurs along the offset axis, $h$, which means that the thread block *extent* is the whole offset axis. This is a naive implementation of the imaging condition because of the redundant reads from global memory. The global memory access pattern of the naive imaging kernel is illustrated in Figure 3(a), where, for each point in the 3D output space, two values from $P_f$ and $P_b$ are needed. The imaging kernel can be improved so that redundant reads from within a thread block are eliminated.

For the sake of simplicity, we consider a 1D thread block of dimension $n$ that has an offset extent of $n$ points; i.e., each thread block will compute a tile of size $n^2$ from the output space $(z, h_z)$. For the naive imaging kernel, the number of global memory accesses is $4n^2$; i.e., to update a point in the output volume (offset gathers), two reads from the wavefields, and one read and one write for updating the offset gathers are needed. Figure 3(b) shows areas that will be accessed from the wavefields. The width for each area is twice the width of the thread block $n$. The common areas can be copied to the shared memory, so that each thread will copy two values from the shared area. After that, the thread block advances along the offset axis and uses the data from shared memory, as shown using the following kernel:

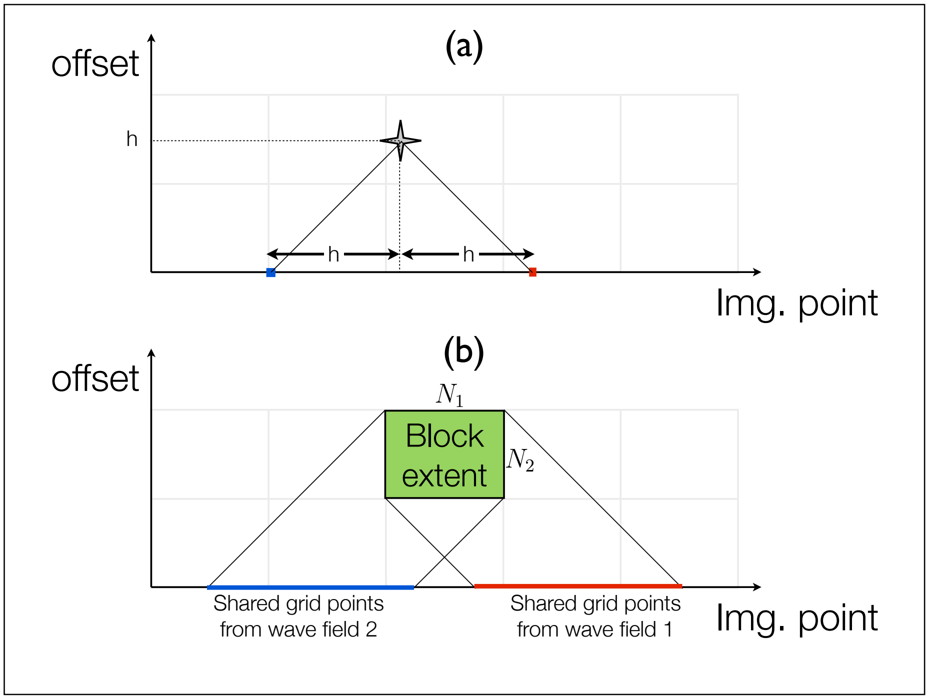

(a)
offset
h
h
h
Img. point
(b)
offset
$N_1$
Block extent
$N_2$
Shared grid points from wave field 2
Shared grid points from wave field 1
Img. point

Figure 3: The memory access pattern of imaging: (a) To compute the image value ( at the star), two points from $P_f$ (square) and $P_b$ (circle) are needed. (b) This diagram shows the thread block advancement along the offset axis and data sharing from within the block. All data access from with the thread block will fall in the shared areas from $P_f$ (dashed) and $P_b$ (solid). **[NR]** altheyab1/. imagingstartegy-crop

```
__global__ void img_hz_improved_kernel(int h0 /* first offset */,
                                       float *p1, float *p2){
  __shared__ float s_p1[BLOCK_SIZE][2*BLOCK_SIZE];
  __shared__ float s_p2[BLOCK_SIZE][2*BLOCK_SIZE];

  int i=hmax_gpu+blockIdx.x*blockDim.x+threadIdx.x;   /* z-location */
  int j=blockIdx.y*blockDim.y+threadIdx.y;            /* x-location */

  /* offset extent of the thread block */
  int hmin=h0-hmax_gpu;                               /* minimum offset */
  int hmax=hmin+BLOCK_SIZE;                           /* maximum offset */

  /* copy to shared memory */
  s_p1[threadIdx.y][threadIdx.x]=p1[loc(i-hmax, j)];
  s_p2[threadIdx.y][threadIdx.x]=p2[loc(i+hmin, j)];
  s_p1[threadIdx.y][BLOCK_SIZE+threadIdx.x]=p1[loc(i-hmin, j)];
  s_p2[threadIdx.y][BLOCK_SIZE+threadIdx.x]=p2[loc(i+hmax, j)];

   /* synchronize threads */
  __syncthreads();

  /* prepare addressing variables */
  int base=loc(i,j)+h0*n1*n2;
  int stride=n1*n2;

  /* internal loop along the offset axis */
  for(int h=0; h<BLOCK_SIZE; h++){
    zimg[base+h*stride]+=
        s_p1[threadIdx.y][BLOK_SIZE+threadIdx.x-h]
       *s_p2[threadIdx.y][threadIdx.x+h];
  }
}
```

The number of global memory accesses for the improved algorithm is $2n^2 + 4n$. For a thread block of width 16, the number of global memory accesses is reduced from 1024 (naive) to 576 (improved).

## KERNELS' PERFORMANCE

The throughput of the implemented kernels are shown in Figure 4. For comparison, thread blocks for imaging kernels were 16x16 for both kernels, and the improved kernel has an offset extent of 16. The throughput of the improved kernel is double the throughput of the naive implementation. Modifying the block sizes yielded a negligible improvements in performance. This optimized kernel is applicable to 3D imaging. However, 4GB of GPU memory is not sufficient to host the 4D output volume.

Figure 5 shows the execution time for RTM without ODCIGs generation, RTM with ODCIGs generation using the naive imaging kernel, and RTM with ODCIGs generation using the improved imaging kernel. Computing ODCIGs is very costly because of the large volume that is updated at every imaging step. The floating-point operation count is very small for the imaging kernel. This indicates that most of the execution time for ODCIGs generation is spent accessing and writing to the device global memory. The memory access time is so dominant that constructing angle-domain common image gathers (ADCIGs) at every time step could be considered.

## SUMMARY AND FUTURE DIRECTIONS

In this paper, we have shown the implementations of the naive imaging kernel. We also demonstrated a strategy for improving the imaging kernel that doubles the throughput of the naive implementation. This implementation is applicable to 3D imaging problems, although the available memory for current GPUs is not sufficient to hold the generated image volumes. We are considering generating horizon-based ODCIGs for 3D wave propagation. Due to the

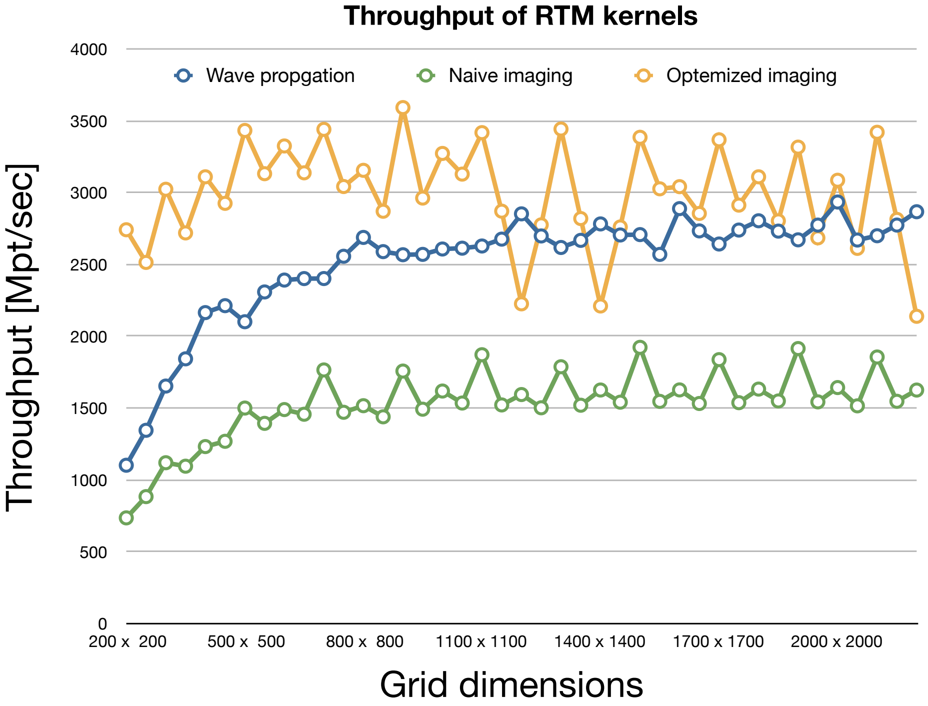


Figure 4: Throughput of the implemented kernels. Improving the imaging kernel using shared memory doubles the throughput of the imaging kernel. **[NR]** altheyab1/. throughput-crop

large output volume, generating ODCIGs is costly as compared to RTM without ODCIGs generation. To reduce the cost, we are considering data compression and subsampling the ODCIGs, which will reduce the output volume that is updated at every imaging step. The cost of generating ADCIGs will be minimal compared to generating ODCIGs. Therefore, we are considering generating ADCIGs on the GPU.

## ACKNOWLEDGMENTS

We would like to thank Nvidia for providing the Tesla S1070 GPUs for SEP. We also thank Paulius Micikevicius for sharing his CUDA implementation of FDTD.

## REFERENCES

Biondi, B. and W. W. Symes, 2004, Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging: Geophysics, **69**, 1283–1298.

Micikevicius, P., 2009, 3d finite difference computation on gpus using cuda: GPGPU-2: Proceedings of 2nd Workshop on General Purpose Processing on Graphics Processing Units, 79–84, ACM.

NVIDIA, 2008, Nvidia cuda programming guide 2.0.

Zhang, Y. and J. Sun, 1993, Practical issues in reverse time migration: true amplitude gathers, noise removal and harmonic source encoding: first break, **27**, 53–60.

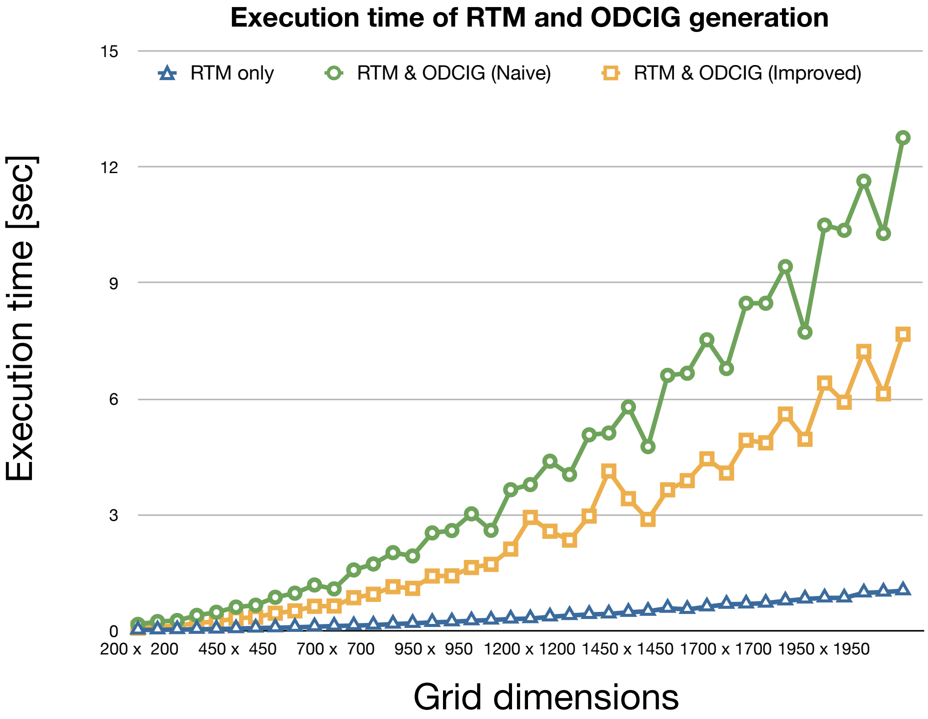


Figure 5: Execution times for migrating a single shot with 1000 time steps and 100 imaging steps. There are 81 points along the z- and x-offset axes for the generated ODCIGs. Because of the large volume of ODCIGs (2 × 81× grid size) that is updated at every imaging step, most of the computing time is spent on generating the gathers. **[NR]** altheyab1/. cost-crop

# Seismic imaging using GPGPU accelerated reverse time migration

*Nader Moussa*

## ABSTRACT

In this report, I outline the implementation and preliminary benchmarking of a parallelized program to perform reverse time migration (RTM) seismic imaging using the Nvidia CUDA platform for scientific computing, accelerated by a general purpose graphics processing unit (GPGPU). This novel software architecture allows access to the massively parallel computational capabilities of a high performance GPU system, which is used instead of a conventional computer architecture because of its high throughput of numeric capabilities.

The key aspects of this research concern the hardware setup for an optimized GPGPU computer system, and investigations into coarse-grained, algorithm-level parallelism. I also perform some analysis at the level of the numerical solver for the Finite-Difference Time Domain (FDTD) wave propagation kernel. This paper demonstrates that the GPGPU platform is very effective at accelerating RTM, and this will lead to more advanced processing for better imaging results.

## INTRODUCTION

Reverse time migration (RTM) is often used for seismic imaging, as it has preferable numerical and physical properties compared to competing algorithms, and thus generates better images (Zhang and Sun, 2009). These benefits come at a high computational cost, so research effort is required to make RTM a more economically competitive method for seismic imaging. This is the motivation for GPGPU parallelism of RTM.

The processing flow for imaging a seismic survey can be parallelized in many tiers. This multi-tiered parallelism has been noted in earlier computer architecture research for seismic imaging (Bording, 1996). This hierarchical parallelism is particularly prominent in RTM, and it provides opportunities for significant performance increase throughout the algorithm. A modern GPGPU platform, such as the Nvidia S1070, is uniquely capable of mirroring this tiered algorithm structure, because its architecture is similarly structured with both coarse-grain and fine-grain parallel capabilities.

At the highest level of abstraction, a data set can be divided into spatially separate regions of independent data (shot profiles). This is a Single Program, Multiple Data (SPMD) approach, and due to low data dependency, interprocess communication is generally not needed. This can directly map to a hardware multi-GPU implementation.

At each data subset, the migration can be further parallelized at a finer granularity. There are three potential stages for parallelism in the RTM algorithm, but there is a severe data dependency limitation. The imaging condition requires the computed results of both the forward wavefield, $p_F$, and the reverse wavefield, $p_R$, for each time step. Unfortunately, because the two wavefields are computed in opposite time directions, performing the imaging condition usually requires computing the complete wavefield $p_F$, writing it to disk, and reading its precomputed values for image condition correlation as soon as that time step is available from the reverse time wavefield. This data dependency is a major obstacle to parallelism at this stage, and constrains performance.

At the finest level of parallelism, the individual wavefield propagation steps can reduce the compational load by taking advantage of vectorization, floating-point math optimizations, and numerical reorganization. The imaging condition can also benefit from parallelization, because it is essentially a large 2D or 3D correlation. This is easily vectorizable, and is especially suitable for a GPU, which was originally designed as a large vector-computer.

Clearly, the GPGPU platform provides multi-tiered parallelism capability that matches the RTM structural design. The encouraging preliminary results seem to confirm that the GPGPU platform is well suited to RTM optimization, and suggest that further optimization can continue to yield dramatic execution time improvements. This will allow more advanced processing with correspondingly better subsurface image results.

## CUDA PROGRAMMING METHODOLOGY

Nvidia's novel technology, "Compute Unified Device Architecture" (CUDA) is a software interface and compiler technology for general purpose GPU programming (Nvidia, 2008). The CUDA technology includes a software interface, a utility toolkit, and a compiler suite designed to allow hardware access to the massive parallel capabilities of the modern GPU, without requiring the programmer to construct logical operations as graphical instructions. The latest release of CUDA, version 2.1, exposes certain features only available in the Tesla T10 GPU series. Below, all specifications are given based on the capabilities of the T10 GPU using CUDA 2.1 software. For easy reference, Table 1 summarizes the terminology and acronyms that apply to the software and hardware tiers. An acronym-guide is also provided in Table 2 in the Appendix.

CUDA programs have two parts: "host" code, which will run on the main computer's CPU(s); and "device" code, which is compiled and linked with the Nvidia driver to run on the GPU device. Most device code is a "kernel," the basic functional design block for parallelized device code. Kernels are prepared and dispatched by host code. When the kernel is dispatched, the host code specifies parallelism parameters, and the kernel is assigned to independent threads which are mapped to device hardware for parallel execution.

The coarsest kernel parallelism is the "block," which contains several copies of threads running the same code. Each block structure maps to a hardware multiprocessor. Blocks subdivide a large problem into manageable units which will execute independently. It should be noted that inter-block synchronization and communication is difficult without using expensive global memory space or coarse barriers. Inside each block there are up to 512 threads, organized into sub-groups or "warps": these groups of up to 32 threads. At this level of parallelism, shared memory and thread synchronization is very cheap, and specific hardware instructions exist for thread synchronization. As of CUDA 1.3, available on the Tesla T10, synchronization "voting" can be used to enable single-cycle inter-thread

| **Software Model** | | **Hardware Model** | |
|---|---|---|---|
| Element | Maximum | Physical Unit | # |
| **Thread** | 512 threads per block<br>Arranged in 3D block not exceeding<br>$512 \times 512 \times 64$ in $< x, y, z >$ and 512 total | Scalar Processor (**SP**) or "Streaming **Core**"<br>Each core executes one thread at a time | **8** |
| **Warp** | Each 32 threads are statically assigned to a warp | **SP Pipeline**<br>A full warp (32 threads) executes in 4 clock cycles (pipelined 4-deep across 8 cores) | **16** |
| **Block** | Arranged in 2D grid not exceeding<br>$65535 \times 65535$ in $< x, y >$ | Streaming Multiprocessor (**SM**) | **30** |
| **Kernel Grid** | Problem or simulation representation | **GPU**<br>Only one kernel is running on the GPU at a time<br>(More are possible, but this is complicated). | **4** |

Table 1: CUDA software and hardware mapping. This table briefly summarizes the CUDA software architecture and its implementation on a T10 GPU.

control. As an extra performance boost, threads which are running the same instructions are optimized with “Single Instruction, Multiple Thread” (SIMT) hardware, sharing the Instruction Fetch (IF) and Decode (DEC) logic and efficiently pipelining operations. If conditional program control flow requires different instructions, the threads must then serialize some of these pipeline stages. Peak performance is achieved when all conditional control-flow is identical for threads in a single warp. In the case of Finite Difference Time Domain (FDTD) wave propagation code, it is generally possible to have all threads operating in SIMT mode. The boundary conditions at the edges of thread blocks, and at the edges of the simulation space, are currently the only exceptions to this SIMT mode.

## HARDWARE PLATFORM

During this research, testing was performed on an HP ProLiant with an attached Tesla S1070 GPGPU rack-mounted blade server. This unique platform implements the CUDA 2.1 software specification, with hardware Compute Capability 1.3 GPU acceleration (Nvidia, 2008). The S1070 provides four Tesla T10 GPUs, which provide vector-style parallelism for general purpose computing.

The basic architecture consists of a “Host System,” using regular CPUs and running a standard Linux operating system. Attached is the “Device,” a 1U rack-mounted GPGPU accelerator which provides the parallelism discussed in earlier sections. The CUDA technology uses the terms “host” and “device” to refer to the various hardware and software abstractions that apply to either CPU or GPU systems. Although the S1070 has four GPUs, it is considered one “device”; and similarly, there is one “host,” although it has 8 CPU cores in this system. Below is a summary of the system specifications:

**Host: HP ProLiant DL360 G5** (HP ProLiant series, 2009) (HP ProLiant DL360G5 Overview, 2009)

- 2× Quad-Core Intel Xeon E5430 @ 2.66 GHz
- 6144 KB L2 Cache (per core)
- L2 Cache Block size: 64 B
- 32 GB main memory
- 1333MHz Front Side Bus
- PCI-e #1: 8x pipes @ 250MB/s each
- PCI-e #2: 8x pipes @ 250MB/s each
- Gigabit Ethernet connection to SEP Intranet

**Device: Nvidia Tesla S1070 Computing System** (S1070 Product Information, 2009)

- 4× T10 GPU @ 1.44GHz
- 30 Streaming Multiprocessors (SM) per GPU

- 8 Scalar Processor (SP) cores per SM
- 32 threads per warp
- 16K 32-bit registers per block
- 16KB Shared Memory per block
- 4GB addressable main memory per GPU
- Memory controller interconnect for data transfer to host and other GPU address spaces
- Concurrent Copy & Execution feature: "Direct Memory Access" (DMA) style asynchronous transfer available on T10 GPUs
- Programmable in CUDA

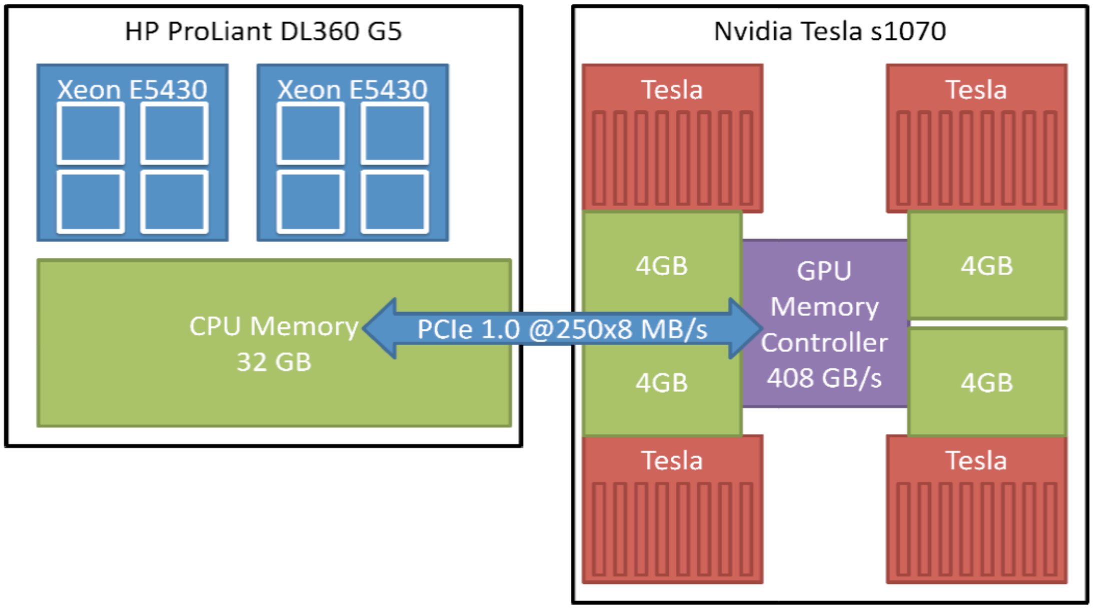


Figure 1: Schematic representation of the Host-Device (CPU-GPU) interconnection and memory structure. The compartmental memory structure on the Device side is problematic for multi-GPU programs, because it severely restricts shared memory methods. Much of my recent implementation efforts address this issue. **[NR]** nwmoussa1/. hostDeviceInterconnectSchem

The S1070 is a very recently released commercial platform specifically tailored for high-performance scientific computing. Nvidia recommends using it only with certain host hardware environments. The system installation procedure is explained in the appendix, along with solutions to difficulties that may be encountered.

The final system environment runs `CENTOS 5.2 for x86_64` and using the Nvidia Tesla Driver (`Linux x86_64 - 177.70.11`). Two Host Interconnect Cards (HIC) are installed and configured in the ProLiant. Both cards are connected to the S1070 unit via two PCI-e cables. This setup produces a reliable and functional system for GPGPU computational acceleration.

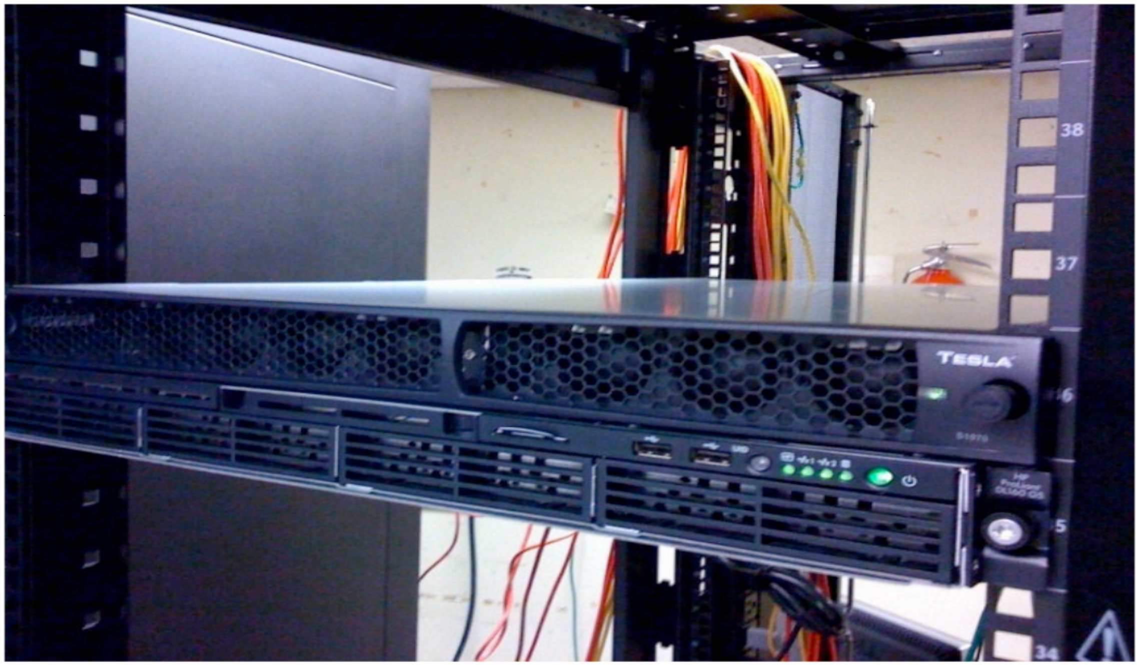


Figure 2: Rack mount view of the SEP Tesla system. At the top is the S1070 1U 4xGPU GPGPU Computing Appliance. Below is the HP ProLiant Xeon 64 bit, 8 core (2xSMP, 4xCMP) system, `tesla0.stanford.edu` which runs the host operating system. [**NR**] nwmoussa1/. teslaRackMountPhoto

## EVALUATION METRICS

There are many ways to compare and evaluate parallelization schemes for RTM. Because the GPGPU approach is so novel, it is difficult to perform direct comparison with other parallelization schemes for Reverse Time Migration. Other hardware platforms do not provide the same software abstractions. Many of the GPGPU metrics thus have no direct comparable equivalent on alternative systems. Of course, key performance metrics are directly comparable to serial or parallel CPU RTM implementations. These include:

- Total execution time
- Cost ($) per FLOPS
- FLOPS per Watt

Other internal performance metrics of my implementation can be compared to academic and industrial research progress in high-performance GPGPU wave propagation. Wave propagation has been previously implemented in Finite Difference Time Domain (FDTD) for nearly identical hardware (Micikeviciuis, 2008); the forward- and reverse-wavefield computation performance can be directly compared to such an implementation. FDTD performance measurements include:

- Maximum computational grid size
- Block subdivision size
- Wavefield grid points per second
- Numerical order of spatial derivatives

One goal of SEPs investigations into various parallelization technologies is to subjectively evaluate the feasibility for future performance, ease of development, and maintainability of code. Technologies like the CUDA/GPGPU approach are compared subjectively to other systems, such as the SiCortex SC072 "Desktop Cluster" as well as conventional multicore and multi-node CPU parallelization. The following metrics can be roughly estimated for each technology, noting that there is some ambiguity in direct comparisons across widely varying exotic architectures:

- Cost ($) per FLOPS
- FLOPS per Watt
- FLOP operations needed for complete migration
- Execution time for complete migration
- Accuracy of the wave propagation operator

As I am not trained as an interpretational geologist, subjective assessment of the image quality is difficult for me. Nonetheless, it has been widely established in industrial contexts that the correct implementation of RTM yields better images for decision-making and analysis. Certain computational architectures can enhance this effect by enabling higher-accuracy RTM, (e.g. using higher-order wavefield operators). By providing very cheap floating-point math, the GPGPU approach enables more operations per data point, allowing more accurate wave modeling with minimal execution time overhead. The overall speedup that a GPGPU implementation can provide can allow additional iterations as part of larger inversion problems, increasing the accuracy of these processes. The result is a subjectively better migrated image.

Finally, it is worth noting the benefits of GPGPU parallelization from a software engineering and code-maintenance standpoint. CUDA is designed to be simple, consisting of a set of extensions to standard C programming. The programming environment is easy to learn for most programmers. The code is systematically separated into host setup code and device parallelization code; and CUDA can interoperate with C or C++, allowing functional- or object-oriented system design, as the situation requires.

## IMPLEMENTATION

I developed a wave propagation kernel, implemented in CUDA, for use in forward- and reverse-time wave propagation. I also implemented a simple correlation imaging condition. Due to time constraints, I was not able to implement a more advanced imaging condition with true-amplitude correction, noise-removal, and angle-gather decomposition.

For the purposes of this report, I will focus on single-GPU kernels. I made significant progress towards multi-GPU asynchronous parallelization, but this code is not yet ready to provide benchmark results. The eventual goal is to perform the forward-wave, reverse-wave, and imaging condition subroutines on independent GPUs. However, preliminary benchmark results cast doubt on whether that approach will decrease total execution time, because the bottleneck appears to be host-device transfer time rather than computational limitations.

Implementation of an eighth-order spatial derivative added negligible computational overhead to the problem, as compared to the naive second-order wave operator. This suggests that other more sophisticated time-stepping methods, such as Arbitrary Difference Precise Integration (ADPI) wave solvers (Lei Jia and Guo, 2008), may also have negligible computational overhead. Such methods will enable coarser time-steps without the numerical stability limitations that are inherent in FDTD approaches. Thus the use of those methods may reduce overall execution time.

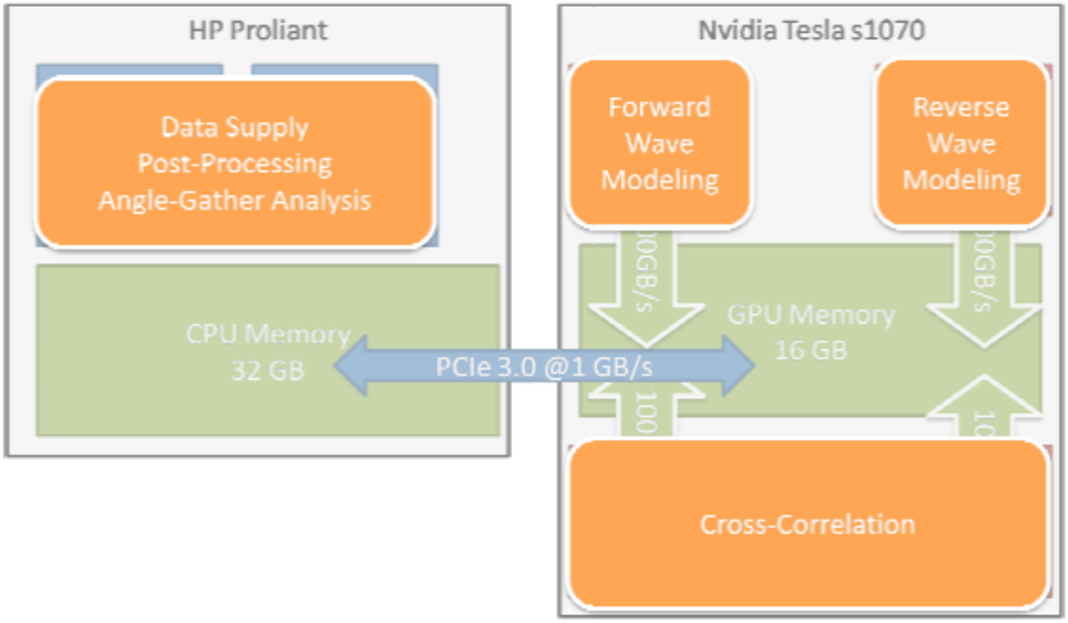


Figure 3: Schematic view of the multi-GPU algorithm for coarse-grained parallelism. The intent is to perform the overall RTM process with separate stages executing in parallel on independently controlled GPUs. This coarse parallelism can help pipeline the process and hide the memory transfer time. My current implementation and benchmark-code does not yet implement this strategy. **[NR]** nwmoussa1/. hostDeviceMultiGPUschem

The expansion of the solver to a full 3D model space will require significant extra programming. The code base for the 2D model is intended to be extensible, and the CUDA framework allows block indexing to subdivide a computational space into 3 dimensions, assigning an (X,Y,Z) coordinate to each block and each thread. Because of time constraints, I did not complete the full 3D modeling for benchmark comparison.

In the current iteration, the host code does not perform significant parallelism. Earlier efforts used `pthread` parallelism on the host CPUs for data preprocessing while the input loaded from disk, but the time saved by this workload parallelism was negligible compared to the overall execution time.

The result of my implementation is a propagation program, `waveprop0`, and an imaging program, `imgcorr0`, written in CUDA. These are piped together with a set of Unix shell scripts to manage the overall RTM sequence for forward and reverse-time propagation with an imaging condition.

Future implementations will seek to integrate these programs into one tool with several CUDA kernels, but the overlying data-dependence issue must be solved theoretically before the processes can be entirely converted to a streaming methodology. For trivial-sized problems, the entire computational result of forward and backward wave propagation can remain in graphics device memory for use, but this approach has inherent problem-size limitations. Other methods of eliminating the costly host-device transfers have been proposed (Clapp, 2009). Such methods eliminate the bottleneck by preserving the wavefield state in GPU memory at the final timestep, and backward-propagating to recompute the wavefield at arbitrary time. This takes advantage of the cheap and fast wave propagation kernel. Another approach is the effective pipelining of the RTM process to allow arbitrary-sized input data sets. Finally, a major area of continuing work is the complete linking of CUDA research code with the standard `SEPlib` programming environment and toolkit. This will be extremely beneficial from the standpoint of code portability and interoperability with other research areas.

## PERFORMANCE AND BENCHMARK SUMMARY

For the sake of simplicity and consistency, I tested my RTM code on synthetic data. I used a simple subsurface velocity model with a few reflecting layers. This same velocity model has been used by other SEP students and researchers, and although it does not represent the complex subsurface behavior of a real earth model, it provides sufficient complexity to evaluate the correct functionality of the RTM implementation. My current work to integrate `SEPlib` with the GPGPU environment will enable benchmarking and testing on more standard data, eventually including field recorded data sets. This will be an important step to verify and compare GPGPU performance to more traditional paralellism schemes.

Unless otherwise noted, the benchmark results I report were computed on a two dimensional wavefield space, with grid size 1,000 x 1,000.

GPU execution time is shown in Figure 5 for a 1,000,000 point grid, (1,000 x 1,000 2D computational space). It is compared to a serial implementation of RTM on the CPU. Due to time constraints, I was not able to compare the GPGPU parallelization to other parallel RTM versions.

Evidently, GPU parallelization has a dramatic effect on the total execution time, reducing it by a factor of more than 10x. With 240x as many cores, however, this is sublinear

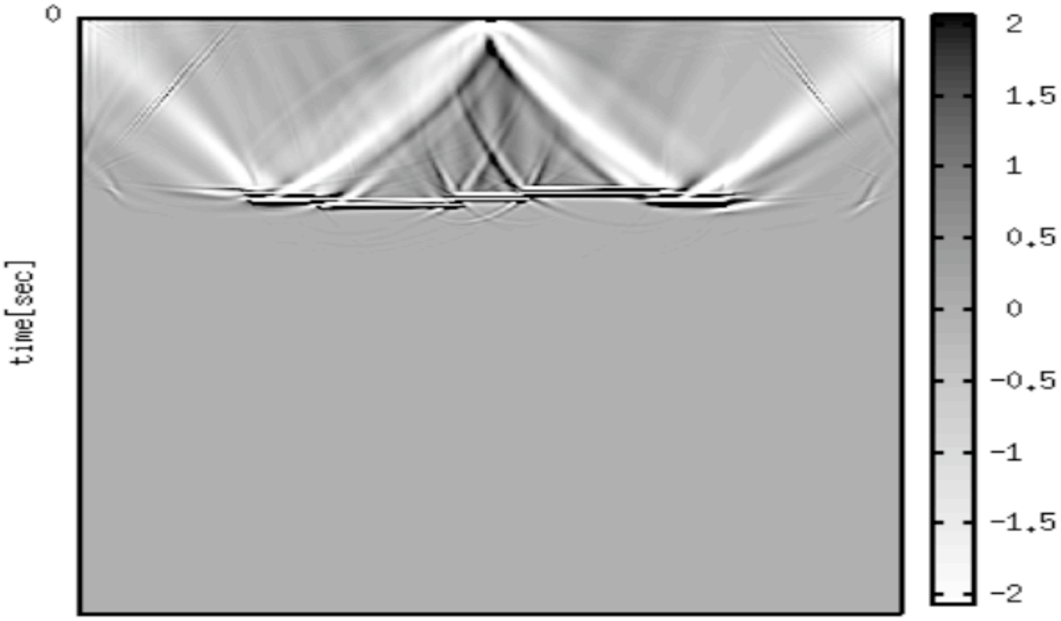


Figure 4: Preliminary RTM image results on a synthetic data set with a few simple horizontal reflectors. This test verified functionality of my preliminary RTM implementation on the GPGPU system. Wave diffraction is visible at the corners, probably due to the unrealistic, abrupt end of the layers in this synthetic model. [**NR**] nwmoussa1/. rtmImagingPreliminary

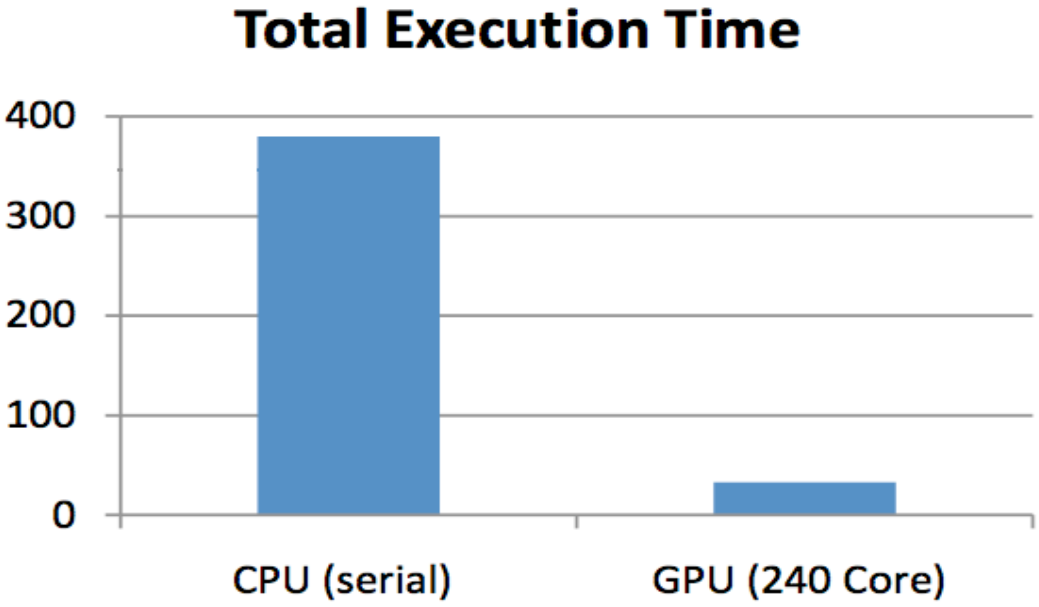


Figure 5: Total execution time for RTM imaging, comparing serial implementation on CPU (executed on the ProLiant Xeon host), compared to a single GPU CUDA parallelization. [**NR**] nwmoussa1/. totalExecutionTimeChart

parallelization. Closer profiling of the CUDA algorithm execution time revealed the computational breakdown shown in Figure 6. This profiling was accomplished using timer variables compiled in the device code, as standard code profilers have difficulty working with the GPGPU environment. Most of the bottleneck is clearly the memory transfers between host and device, which are required for the imaging condition. The primary focus of further research is to work around this limitation: first, by optimizing the memory transfers as much as possible; and more importantly, by developing numerical schemes that can perform the imaging step without as much expensive transfer overhead.

## CONCLUSION

The dramatic speedup of the computational kernel provides strong motivation for continued work in GPGPU parallelism. Benchmark results suggest that the most important area to tackle is Host-Device (PCI-e) bus bandwidth, which accounts for 90% of the total system utilization time.

At present, my implementation does not have any tasks for the high-performing Xeon processors on the host. These CPUs are suitable for performing a lot of useful work, such as data post-processing or visualization. An alternative architecture could tightly couple CPU and GPU processes to maximize system utilization.

Another suggested research area is the implementation of compression during transfer. Velocity models, which contain large quantities of redundant data, could easily be com-

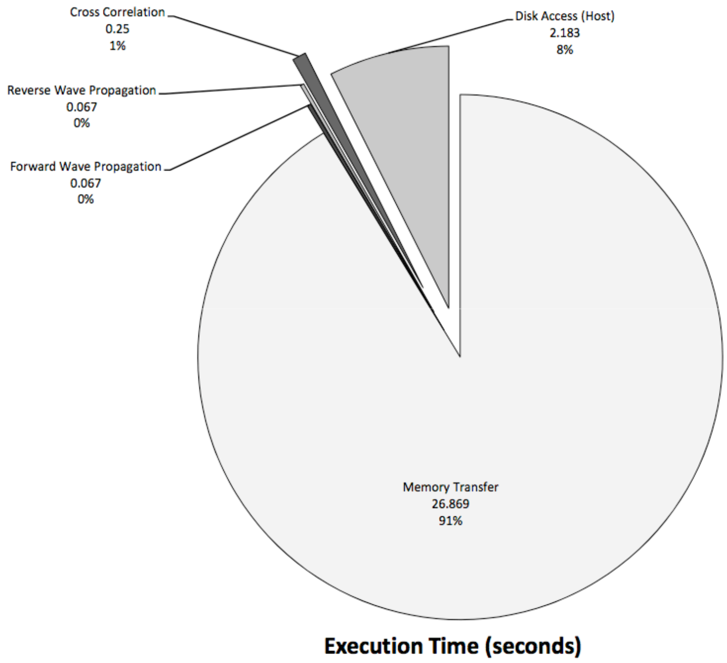


Figure 6: Breakdown of program execution time for the CUDA implementation. Very little time is spent executing numerical processing code (wave propagation or imaging condition). The vast majority of time is spent in host-device memory transfer over the PCI-e bus (between the ProLiant CPU system and the Nvidia Tesla S1070). **[NR]** nwmoussa1/. execTimePieChart

pressed; seismic records will probably not compress well with a lossless algorithm such as GZIP (LZ77) because they do not contain the same amount of redundancy as velocity models. In future work, I will quantify these compression ratios for real data sets, which will help validate the utility of compressed Host-Device communication.

The system-level transition towards exotic computing platforms is always an engineering tradeoff. The performance benefits of such an environment must be sufficiently high to offset the development and maintenance cost with the new system. The GPGPU platform and its CUDA programming environment is sufficiently familiar to a geophysical programmer and exposes massive parallel capability in a straightforward way. The immediate performance boost is evident from the preliminary benchmarks presented in this report. Significant further optimizations can be realized in future work via more analysis and refinement of this GPGPU approach.

# APPENDIX

## Tesla S1070 system setup

This section details the procedure to install and configure the hardware for the Tesla S1070 GPGPU system.

During my early work, I configured the ProLiant system to run Ubuntu 8.10 and Nvidia 180.22 drivers. This required recompiling Nvidia Debian kernel modules (.ko files). The recompiled modules successfully connected to the S1070 system, but incorrectly identified it as an Nvidia C1060. Downgrading the operating system to Ubuntu 8.04 enabled the modules to correctly connect and recognize the Nvidia S1070, but also produced an unstable system, which occasionally crashed. Following advice from Nvidia, I switched the ProLiant operating system to CENTOS and had significantly more success. However, the Nvidia 180.22 drivers have not been fully tested on the 1U rack-mount S1070 systems with four GPUs. There were several system hang-ups and unexpected, non-repeatable crashes. It should be noted that the GPGPU driver for the 1U Tesla system interferes with some automatic configuration of the Linux operating system (specifically graphic configuration for X11). This happens because the S1070 appears to X11 to be a video accelerator and display driver even though it cannot be connected to a physical display monitor.

Another potential configuration problem arises from the presence of two Host Interconnects on the S1070 1U unit. The Nvidia documentation mentions that these interconnects allow the S1070 to optionally connect to two separate host CPU systems. However, even though only one host is used in our system, we found that both interconnects should be used because Connecting and configuring only one card results in access to only 2 out of the 4 available Tesla T10 GPUs on the S1070 1U server. Using both interconnects allows access to all four GPUs, and also doubles the PCI-e bandwidth available to the S1070 memory controller.

The final system environment runs `CENTOS 5.2 for x86_64` and using the Nvidia Tesla Driver `(Linux x86_64 - 177.70.11)`. Two Host Interconnect Cards (HIC) are installed and configured in the ProLiant. Both cards are connected to the S1070 unit via two PCI-e cables. This setup produces a reliable and functional system for GPGPU computational acceleration. Much difficulty can be avoided by using exactly these system and driver versions.

### Acronyms Quick Reference

| | |
|---|---|
| CUDA | Compute Unified Device Architecture |
| FDTD | Finite Difference, Time Domain (wave simulation) |
| GPU | Graphics processing unit |
| GPGPU | General purpose graphics processing unit |
| RTM | Reverse Time Migration |
| SEP | Stanford Exploration Project |
| SM | Streaming Multiprocessor |
| SP | Streaming Processor |
| SPMD | Single program, multiple data |

Table 2: Quick reference for CUDA acronyms.

## REFERENCES

Bording, R. P., 1996, Seismic modeling with the wave equation difference engine: SEG Expanded Abstracts, **15**, 666–669.

Clapp, R. G., 2009, Reverse time migration with random boundaries: SEP-Report, **138**.

HP ProLiant DL360G5 Overview 2009, HP ProLiant DL360 Generation 5 - overview. Hewlett Packard. (`http://h18004.www1.hp.com/products/quickspecs/12476_na/12476_na.html`).

HP ProLiant series 2009, HP ProLiant DL360 G5 Server series specifications. Hewlett Packard. (`http://h10010.www1.hp.com/wwpc/us/en/en/WF06a/15351-15351-3328412-241644-241475-1121486.html`).

Lei Jia, W. S. and J. Guo, 2008, Arbitrary-difference precise integration method for the computation of electromagnetic transients in single-phase nonuniform transmission line: IEEE Transactions on Power Delivery, **23**.

Micikeviciuis, P., 2008, 3d Finite Difference Computation on GPUs using CUDA.

Nvidia, ed., 2008, Nvidia CUDA Programming Guide, version 2.1: Nvidia Corporation.

S1070 Product Information 2009, Nvidia S1070 Computing System. Nvidia. (`http://www.nvidia.com/object/product_tesla_s1070_us.html`).

Zhang, Y. and J. Sun, 2009, Practical issues in reverse time migration: First Break, **27**, 53–59.

# Accelerating 3D convolution using streaming architectures on FPGAs

*Haohuan Fu, Robert G. Clapp, Oskar Mencer, and Oliver Pell*

## ABSTRACT

We investigate FPGA architectures for accelerating applications whose dominant cost is 3D convolution, such as modeling and Reverse Time Migration (RTM). We explore different design options, such as using different stencils, fitting multiple stencil operators into the FPGA, processing multiple time steps in one pass, and customizing the computation precisions. The exploration reveals constraints and tradeoffs between different design parameters and metrics. The experiment results show that the FPGA streaming architecture provides great potential for accelerating 3D convolution, and can achieve up to two orders of magnitude speedup.

## INTRODUCTION

The oil industry has always been one of the leading consumers of high performance computing systems. With the increasing of the CPU clock frequencies coming to an end, we can no longer double our computation speed by purchasing updated computers every eighteen months and need to adapt to new computation architectures, such as multi-core processors, General Purpose Graphic Processing Units (GPGPUs), and Field Programmable Gate Arrays (FPGAs).

Recent research work has shown that FPGAs can provide a customized solution for a specific application and achieve more than two orders of magnitude speedup compared to a single-core software implementation. Examples include cryptology applications (Cheung et al. 2005), finance and physics simulations (Zhang et al. 2005; Gokhale et al. 2004) as well as seismic computations (Nemeth et al. 2008).

The major difference between FPGA and other computation platform is the reconfigurability of the processing and storage units in the device, which enables an FPGA to be configured into arbitrary processing units and circuit structures. The reconfigurability of the FPGA leads to two major advantages over other computation platforms:

(1) A streaming computation architecture. While CPUs and GPGPUs take in a sequence of instructions that operate on corresponding data in memory, in FPGAs the instructions are mapped into circuit units along the path from input to output. The FPGA then performs the computation by streaming the data items through the circuit units. The streaming architecture makes efficient utilization of the computation device, as every part of the circuit is performing an operation on one corresponding data item in the data stream.

(2) Customizable number representations. While CPUs and GPGPUs can only handle 8-, 16-, 32- or 64-bit variables, FPGAs support arbitrary bit width for each variable in the design. By adjusting the bit widths according to the precision requirement, we can often

achieve significant reduction in the silicon area cost of arithmetic units and the bandwidth requirement between different hardware modules, thus improving the overall throughput of the entire system.

To investigate FPGA's capability on solving the convolution problem, we explore design options such as: (1) using different stencils; (2) fitting multiple stencil operators into the FPGA; (3) processing multiple time steps in one pass; (4) customizing the computation precisions. The exploration demonstrates constraints and tradeoffs between different design parameters and metrics. Experiment results show that the streaming computation architecture of FPGAs can provide up to two orders of magnitude speedup compared to a single-core software implementation.

# STREAMING ARCHITECTURE FOR CONVOLUTION

## Target Application

Our target application is a 512 by 512 by 512 finite difference problem, with 6th to 8th order in space and 2nd order in time accuracy. Each time step of the computation takes the current wave-field state, the wave-field state from the previous time step and the velocity model as inputs, and produces the next wave-field state as the output.

## FPGA Platform

Current Xilinx FPGAs contain three major categories of resources: (1) reconfigurable logic slices with 6-input lookup tables (LUTs) and flip flops (FFs); (2) DSP48E arithmetic units that can perform $18 \times 25$ multiplications; (3) 36-KBit Block RAM (BRAM)s used as local storage or FIFOs.

In our work, we use the Maxeler MAX2 acceleration card, which contains two Virtex-5 LX330T FPGA chips, 12 GB onboard memory, and a PCI-Express x16 interface to the host PC. Table 1 and 2 show the resource summary of our current FPGAs and the recently released Virtex-6 SX475T FPGA, and the basic cost for implementing single-precision floating-point units on FPGAs.

| FPGAs | #LUTs | #FFs | #DSP48Es | #BRAMs |
|---|---|---|---|---|
| LX330T | 207,360 | 207,360 | 196 | 324 |
| SX475T | 287,600 | 595,200 | 2,016 | 1,064 |

Table 1: Resource summary of the Virtex-5 LX330T and Virtex-6 SX475T FPGA.

| Operations | #LUTs | #FFs | #DSP48Es | #BRAMs |
|---|---|---|---|---|
| $+/-$ | 425 | 557 | 0 | 0 |
| $\times$ | 122 | 173 | 2 | 0 |

Table 2: Costs for single-precision floating-point units.

## Streaming Architectures

Finite difference based convolution operators normally perform multiplications and additions on a number of adjacent points. While the points are neighbors to each other in a 3D geometric perspective, they are often stored relatively far apart in memory. For example, in the 7-point 2D convolution performed on a 2D array shown in Figure 1, data items $(0,3)$ and $(1,3)$ are neighbors in the $y$ direction. However, suppose the array uses a row-major storage and has a row size of 512, the storage locations of $(0,3)$ and $(1,3)$ will be 512 items (one line) away. For a 3D array of the size $512 \times 512 \times 512$, the neighbors in $z$ direction will be $512 \times 512$ items away. In software implementations, this memory storage pattern can incur a lot of cache misses when the domain gets larger, and decreases the efficiency of the computation.

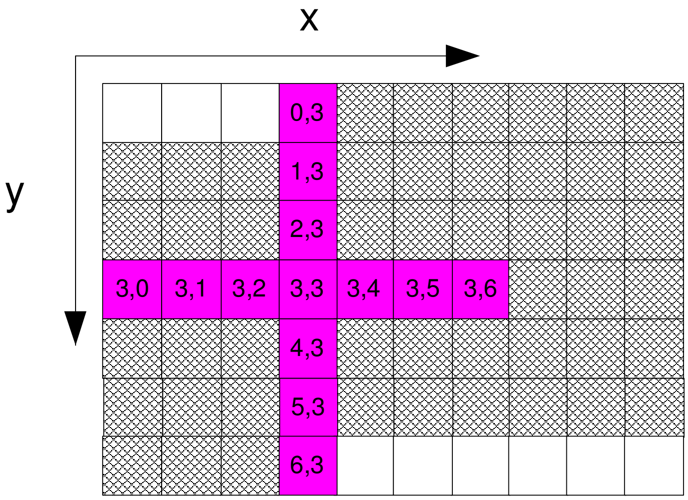


Figure 1: A streaming example of 2D convolution. [**NR**] bob3/. convolution-2D

In an FPGA implementation, we use a streaming architecture that computes one result per cycle. As shown in Figure 1, suppose we are applying the stencil on the data item $(3,3)$, the circuit requires 13 different values (solid, dark-color), two of which ($(0,3)$ and $(6,3)$) are three lines away from the current data item. As the data items are streamed in one by one, in order to make the values of $(0,3)$ and $(6,3)$ available to the circuit, we put a memory buffer that stores all the six lines of values from $(0,3)$ to $(6,3)$ (illustrated by the checker board pattern on the grid). For a row size of 512, this incurs a storage cost of $512 \times 6$ data items.

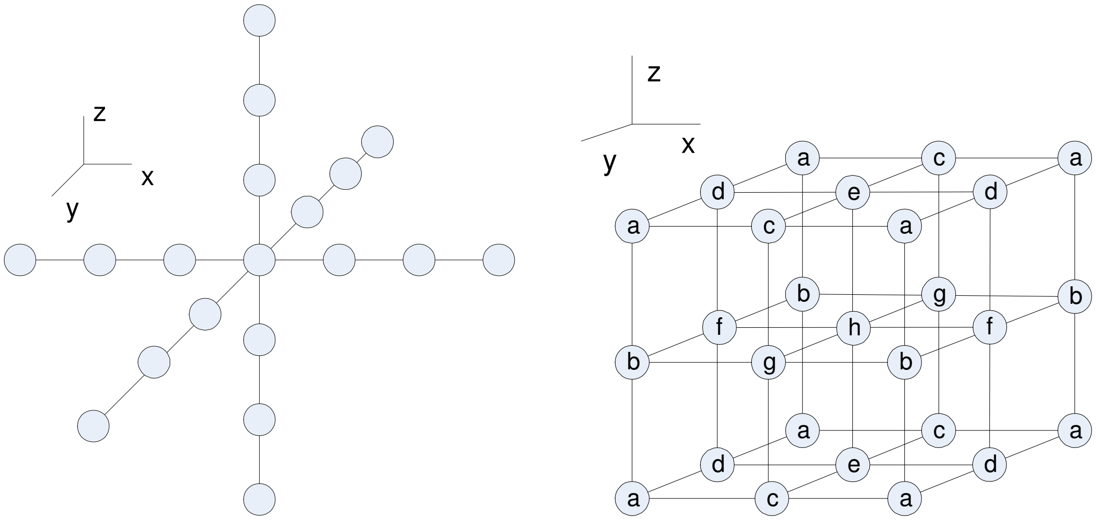
Similarly, for a 7-point 3D convolution on a $512 \times 512 \times 512$ array, the design requires a buffer for $512 \times 512 \times 6$ data items. Assume each data item is a single-precision floating-point number, the buffer size amounts to 6 MB for the $512 \times 512 \times 512$ example. The FPGA chip we currently use provides 1.4 MB of potential buffer size, which is not enough to store all the streamed-in values. We solve this problem by 3D blocking, i.e. dividing the original 3D array into smaller-size 3D arrays, and performing convolution on them separately.

3D blocking reduces the buffer requirement for an FPGA convolution implementation at a cost. Given a convolution stencil with $ns$ non-zero lags in each direction, we must send in a $(nx+ns) \times (ny+ns) \times (nz+ns)$ block to produce a $nx \times ny \times nz$ output block. As $nx$, $ny$, $nz$ becomes small, the blocking overhead can dominate. Meanwhile, the initialization cost for setting up the memory address registers and start the streaming process is also increased as we need to stream multiple blocks.

# EXPLORATION OF DESIGN OPTIONS

## Different Stencils

Our target application uses a a 7-point ‘star’ stencil (Figure 2(a)) to perform the 8th order finite difference. In our exploration, beside the ‘star’ stencil, we also consider a 3-by-3-by-3 ‘cube’ stencil (Figure 2(b)), which performs a 6th order finite difference (Spotz and Carey 1996).



(a) (b)

Figure 2: Different 3D stencils: ‘star’ vs. ‘cube’. [**NR**] bob3/. star,cube

In software implementations, the ‘cube’ and the ‘star’ stencils provide a similar performance. For the FPGA implementations, the resource costs for the ‘star’ and the ‘cube’ stencils are different. The upper part of Table 3 shows the straightforward implementations of ‘star’ and ‘cube’ stencils for a 120x120x120 array. The ‘cube’ consumes 20% more DSP48E arithmetic units than ‘star’, as it involves more multiplications. Meanwhile, the memory cost (BRAM) of the ‘cube’ is one third of the ‘star’, as the data buffering requirement decreases from 6 slices to 2 slices.

For the FPGA designs, we can reduce the count of arithmetic operations by exploiting the symmetry of the coefficients. For example, in the ‘cube’ stencil shown in Figure 2(b), the stencil coefficients are the same for the points marked with the same letters, as both the Laplace derivatives and the scaling ratio determined by the sampling rate of different axes are the same for these points. Therefore, instead of computing $a1 \times c + a2 \times c + a3 \times c$, we compute $(a1 + a2 + a3) \times c$. Applying this technique, the computation for the ‘cube’ stencil reduces from 27 multiplications and 26 additions to 8 multiplications and 26 additions, while the ‘star’ stencil reduces from 19 multiplications and 18 additions to 10 multiplications and

| Normal Stencils | | 'star' | 'cube' |
|---|---|---|---|
| FPGA resource costs | #slices | 5618 | 7072 |
| | #BRAMs | 87 | 30 |
| | #DSP48Es | 50 | 60 |

| Optimized Stencils | | 'star' | 'cube' |
|---|---|---|---|
| FPGA resource cost | #slices | 5207 | 6256 |
| | #BRAMs | 87 | 30 |
| | #DSP48Es | 32 | 18 |

Table 3: Resource costs of 'star' and 'cube' 3D stencils for 120x120x120 3D arrays.

18 additions. The lower part of Table 3 shows the resource costs for multiplication-reduced 'cube' and 'star'. While the cost of BRAMs remains the same, the number of DSP48Es reduces significantly for the 'cube'. After the multiplication reduction, the 'cube' consumes much less than the 'star' for both DSP48Es and BRAMs. The 'star' consumes less logic slices as it involves fewer additions.

As the stencil operator only consumes 8 or 10 multiplications and 26 or 18 additions, the FPGA has the capacity for multiple copies of the stencil operators. Therefore, we have two different ways to improve the performance of the FPGA: (1) using multiple stencil operators to work on multiple data items in parallel; (2) processing multiple time steps in one pass. The following sections discusses these two options in more detail.

## Multiple Stencil Operators

To make a full utilization of all the units on an FPGA, we can try to fit as many stencil operators as possible into the chip. For the example shown in Figure 1, instead of processing only (3,3), we can process consecutive data items (such as (3,2), (3,3), and (3,4)) in parallel.

However, increasing the number of stencil units does not always improve the overall performance due to the constraint of the bandwidth between the FPGA and the onboard memories, which is approximately 13 GB/s in our platform. Considering the controlling overheads, the bandwidth for pure input and output data is around 8 GB/s. When the input streams for the multiple stencil operators approach the saturation point of the memory bandwidth, increasing the number of stencil operators may not improve the performance any more.

Using measured experiment results, we built a software tool that models the costs and performance of various FPGA designs. Figure 3 shows the estimated performance for processing a $512 \times 512 \times 512$ 3D convolution using different number of computation cores on an FPGA. The FPGA circuit is running at 125 MHz. The speedup is calculated against a single-core software implementation running on Intel Xeon 2.0 GHz. Due to the constraint of logic slices, the FPGA can fit six concurrent 'cube' stencils or eight concurrent 'star' stencils. For all the different number of stencil operators, the 'cube' provides a slightly better performance than the 'star'. Both the 'cube' and the 'star' arrive at the saturation point of around 25x speedup with four stencil operators.

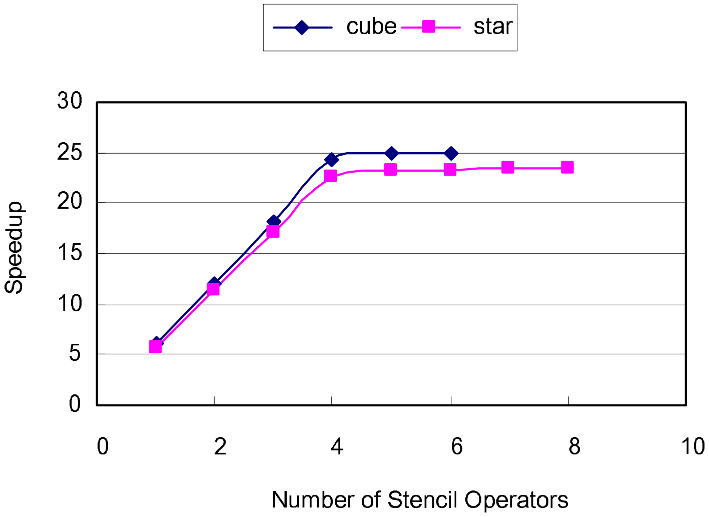


Figure 3: Speedups for processing a $512 \times 512 \times 512$ 3D convolution using multiple stencil operators. [**NR**] bob3/. multiple-cores

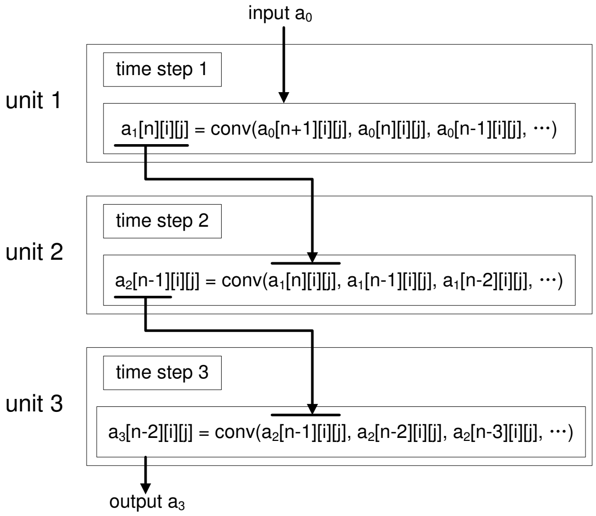

input a0
unit 1
time step 1
a1[n][i][j] = conv(a0[n+1][i][j], a0[n][i][j], a0[n-1][i][j], ···)
unit 2
time step 2
a2[n-1][i][j] = conv(a1[n][i][j], a1[n-1][i][j], a1[n-2][i][j], ···)
unit 3
time step 3
a3[n-2][i][j] = conv(a2[n-1][i][j], a2[n-2][i][j], a2[n-3][i][j], ···)
output a3

Figure 4: Basic circuit structure for processing multiple time steps ($a_i$ denotes the wave-field data in the time step $i$). [**NR**] bob3/. multi-steps

## Processing Multiple Time Steps

Instead of putting concurrent cores, another strategy is to process multiple time steps in one pass. Figure 4 shows the basic structure of a circuit that processes three time steps in one pass. The three units process three time steps separately with the output of each unit as the input of the next unit. The example in the figure uses a 3-by-3-by-3 'cube' stencil. In general, the computation of a wave-field data in slice $n$ requires the wave-field data in slices $(n + 1)$, $n$, and $(n - 1)$ in the previous time step. Therefore, when the unit 1 starts processing slice $n$, the unit 2 can start processing slice $(n - 1)$. Meanwhile, unit 2 needs intermediate buffers to store the results for slices $(n - 1)$ and $n$ from unit 1.

An advantage of processing multiple time steps over putting multiple stencil operators is that the performance will not be constrained by the memory bandwidth, as the unit for each time step is getting inputs from the previous time step, and does not consume the memory bandwidth of the FPGA.

However, on the data side, as we are doing a 3D blocking of the array, processing multiple time steps requires extra data items to start with. Given a convolution stencil with $ns$ non-zero lags in each direction, to process $n$ time steps in one pass for a $nx \times ny$ array, we need to start with an array of the size $(nx + 2 \times n \times ns) \times (ny + 2 \times n \times ns)$. Considering doing 10 time steps for a 100x100 size, the data overhead is 44% for the 'cube' and 156% for the 'star'.

Meanwhile, as the unit at each time step needs to store the results of the previous time step, this approach also increases the requirement for BRAM resources. Therefore, to increase the number of time steps, we need to reduce the blocking size, and thus increasing the cost of streaming overlapping data items and doing a larger number of streams.

Another advantage of this multiple-time-step architecture is that we can improve the order of time accuracy with relatively small costs. For example, for the unit 3 in Figure 4, instead of only getting the previous wave-field data $a_2$ from unit 2, we can get in the wave-field data $a_2$ and $a_1$ from both units 2 and 1 to achieve 4th order in time accuracy. The cost for improving the time order is the extra buffer to store the wave-field data from unit 1 and the increased number of adders and multipliers.

Figure 5 shows the estimated performance for FPGA convolution designs that process multiple time steps in one pass. The 'star', the 2nd and 4th order 'cube' are compared here. For this approach, the 'cube' stencil shows a much better performance than the 'star' stencil due to its smaller requirement for BRAM resources ('star' needs to buffer six slices for the convolution operation, while 'cube' only needs to buffer two). Due to the constraint of logic slices, the FPGA can fit eight time steps for the 'star', six and five steps for the 2nd and 4th order 'cube'. The 'star' gets its peak performance of 11x speedup with four time steps. After that, the performance becomes worse with more time steps. The 2nd order 'cube' stencil increases all the way to 29x speedup with 6 time steps. The 4th order 'cube' achieves 25x speedup with 5 time steps.

## Different Precisions

As mentioned above, one of FPGA's advantages is the support for customizable number representations. Our previous work (Fu et al. 2008) has shown that, in certain cases of seismic computations, reduced precision provides equivalent results within acceptable tolerances. For FPGA designs, a reduced precision can significantly reduce the area cost and I/O bandwidth of the design, and multiply the performance with more computation units on the FPGA.

Figure 6 shows the performance we can achieve using a reduced floating-point precision. With a 16-bit floating-point precision, the multiple-stencil approach provides 49x speedup and the multiple-time-step approach provides 46x speedup.

# ACCELERATION RESULTS

We have implemented the 2nd order 'cube' with 6 time steps and the 4th order 'cube' with 5 time steps onto the Maxeler acceleration card. The 2nd order 'cube' processes 6 time steps in 1.383 seconds, and the 4th order 'cube' processes 5 time steps in 1.346 seconds. Compared to the 6.36 seconds to process one time step in 2nd order, the 2nd and 4th

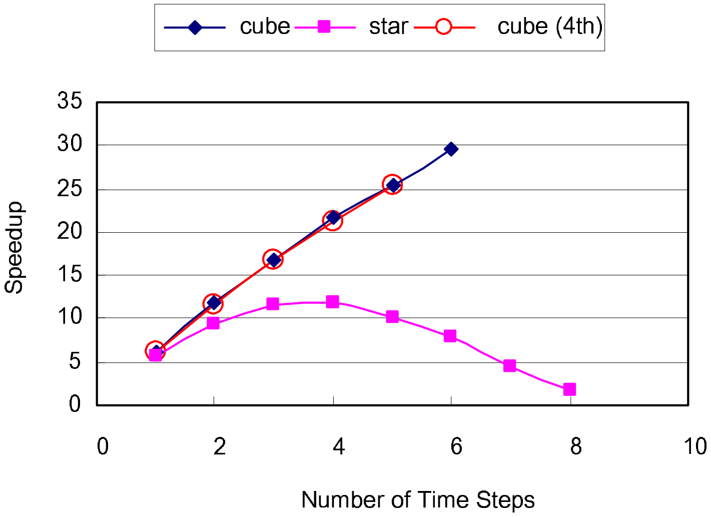


Figure 5: Speedups for processing different number of time steps processed in one pass. [**NR**] bob3/. multiple-timesteps

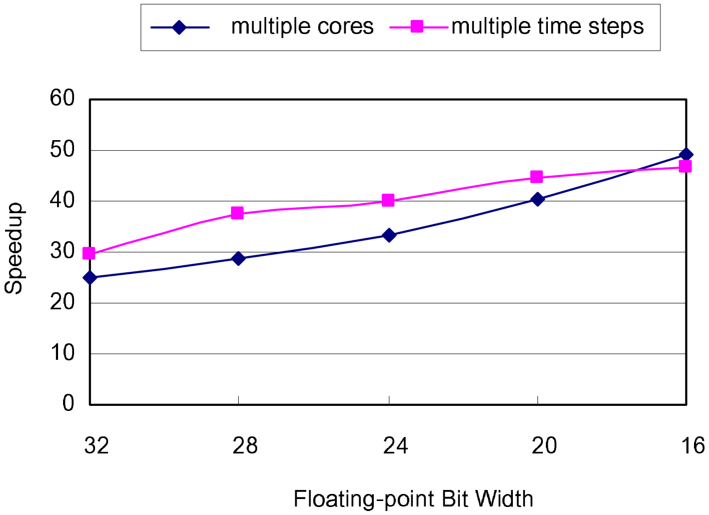


Figure 6: Speedups for different floating-point precisions. [**NR**] bob3/. float-prec

order 'cube' designs provide 27.5x and 23.5x speedups, slightly lower than our estimated performance.

The speedup discussed so far is achieved by using one FPGA of the acceleration card. The acceleration card contains two FPGAs of the same settings. There is also a inter-FPGA link which can update the overlapping boundaries between the FPGAs in parallel with the computation performed on the FPGAs. Therefore, by dividing the array into two parts and computing in two FPGAs concurrently, we can get another 2x and achieve up to 55x and 47x speedup in total.

Note that the FPGAs we are using are Xilinx Virtex-5 LX330T chips released several years ago. Projecting our designs into the recently announced Xilinx Virtex-6 SX475T FPGAs (shown in Table 1), we can fit up to 13 time steps in one FPGA and achieve up to 55x speedup. With two FPGAs working concurrently on an acceleration card, we can achieve up to 110x speedup compared to a single-core CPU version.

## CONCLUSIONS

Our exploration on FPGA convolution designs shows that, the 'cube' stencil fits the FPGA streaming architecture much better than the 'star' stencil. We especially investigate the architecture that processes multiple time steps in one pass. This approach removes the constraints of the memory bandwidth, and improves the performance at the cost of extra data buffering and streaming overhead. Experiment results show that the FPGA streaming architecture provides great potential for accelerating 3D convolution, and can achieve up to two orders of magnitude speedup.

## ACKNOWLEDGMENTS

We would like to thank Maxeler Technologies for providing the hardware device and the Center for Computational Earth and Environmental Science in Stanford University for funding this research.

## REFERENCES

Cheung, R., N. Telle, W. Luk, and P. Cheung, 2005, Customisable elliptic curve cryptosystems: IEEE Transactions on VLSI Systems, **13**, 1048–1059.

Fu, H., W. Osborne, R. Clapp, and O. Pell, 2008, Accelerating seismic computations on fpgas: From the perspective of number representations: Presented at the .

Gokhale, M., J. Frigo, C. Ahrens, J. Tripp, and R. Minnich, 2004, Monte Carlo radiative heat transfer simulation on a reconfigurable computer: Proc. FPL, LNCS 3203, 95–104.

Nemeth, T., J. Stefani, W. Liu, R. Dimond, O. Pell, and R. Ergas, 2008, An implementation of the acoustic wave equation on FPGAs: Presented at the .

Spotz, W. and G. Carey, 1996, A high-order compact formulation for the 3d poisson equation: Numerical Methods for Partial Differential Equations.

Zhang, G., P. Leong, C. Ho, K. Tsoi, C. Cheung, D. Lee, R. Cheung, and W. Luk, 2005, Reconfigurable Acceleration for Monte Carlo based Financial Simulation: Proc. FPT, 215–222.

# Short note: SEP data catalog

*Abdullah Al Theyab, Gboyega Ayeni, and Yunyue (Elita) Li*

## ABSTRACT

Motivated by the long-recognized need for bookkeeping of the datasets in the SEP data library, we have implemented a data catalog database supported by a web-based front end. The new database facilitates searching, referencing, and, most importantly, maintenance of our data library. The database design enables direct connection between each dataset and any relevant internal and external publications. We summarize the database design, data catalog structure, project progress, and future directions.

## INTRODUCTION

Being a data-oriented academic research consortium, Stanford Exploration Project (SEP) puts significant effort into keeping its large data library organized. In the past, meta-data organization was based on text and latex log files written by previous SEP students (Clapp et al., 1999). Many years after they were introduced, most of the log files broke down as a result of system and software changes or file deletion and reorganization. This breakdown of the log system makes accuracy and consistency difficult for data maintainers and users. Because of the large number of datasets and their non-systematic organization, searching for datasets related to particular problems is both cumbersome and time consuming.

The premise for the current effort is the recognition that, by linking datasets to: (1) *keywords* of geophysical problems and data types, (2) data providers, and (3) relevant documents and correspondences, data search and maintenance can be more efficient. Because soft linking using log files is non-trivial (if not impossible), a different catalog design paradigm is required.

Databases are better suited for meta-data organization. They are easy to use, maintain, and query for needed information. Because databases enable complex queries, they can save a lot of research and data maintenance time. They can also be used as a monitoring tool for meta-data consistency. Our database-based catalog system will enable:

- Easy meta-data access and maintenance.
- Direct links between datasets, providers and provider contacts.
- Automated yearly notifications of expired data licenses.
- References to online and offline copies of data files.
- Automated daily checks and reports on status of data files.
- Automated bi-yearly reread (rotate) for backup disks (tapes).

- Interrelation of different datasets using geophysical- and data-type keywords.
- Direct links between each dataset and internal or external publications.

In this report, we summarize the new SEP data catalog design and show snapshots of the working website for future reference.

## THE ESSENTIAL SOFTWARE

The new database is hosted by the MySQL database management system. The front-end website, run by an Apache server and available only to SEP researchers, is interfaced with the database by using Python modules. Figure 1 shows the basic software elements used in building the new data catalog. Although the database is accessible in several ways, the website is the most user-friendly.

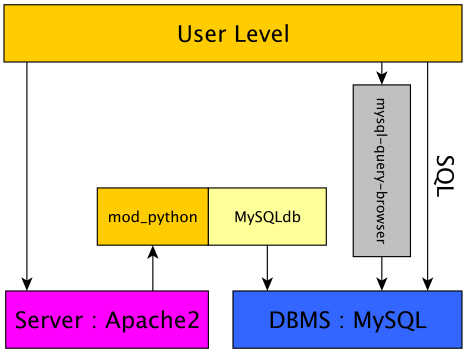


Figure 1: Software elements for the SEP data catalog.**[NR]** altheyab2/. proj7-crop

## THE CATALOG SCHEMA

The database design (Figure 2) is composed of many entities revolving around the *dataset* entity. The *dataset* entity is a table of attributes related to each dataset in our data library. Examples of such attributes include:

- *name* : a distinctive name for the dataset.
- *proprietary_info* : information about who can use the dataset, required permissions before publishing results from this dataset, and provider contact(s) for such permissions.
- *lic_exp* : dataset license expiration year (where applicable).
- *sep_handler* : a list of previous and current SEP researchers who obtained and have maintained the dataset.

The *associated data* entity represents the subsets of a dataset. Examples include the velocity, density, and raw data files associated with a particular dataset. *Associated data* entities have both online and/or offline copies.

The *paper* entity relates each dataset to publications in which such dataset has been used.

The *document* entity holds any file that is of a value to future users and maintainers of a dataset.

Each entity in the database has a FIXME flag and log attributes (not shown in the ER-diagram). These two attributes are used to report problems and fixes to data by maintainers and users.

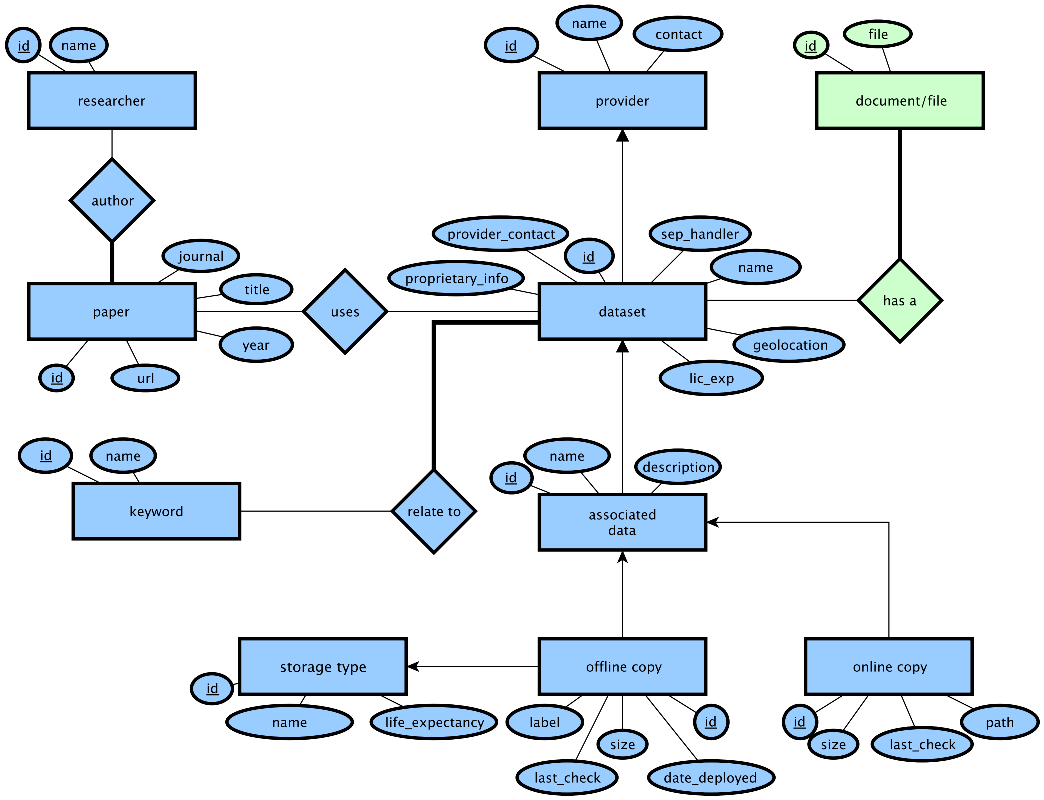


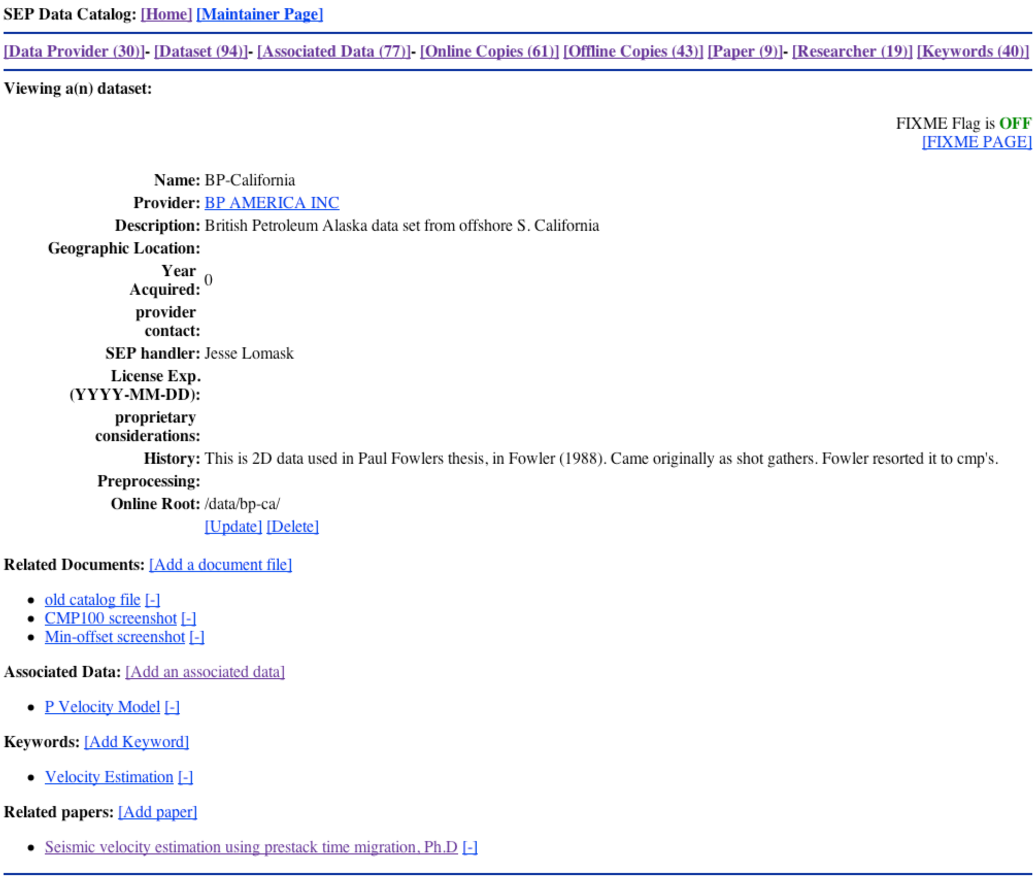
Figure 2: Entity-relation diagram of the SEP data catalog database.**[NR]**
altheyab2/. erdiag-crop

## THE PROJECT PROGRESS

The database and the front-end website are active. Entries for legacy and new datasets are being created and the exploratory analysis of these will be uploaded into the database. Figure 3 shows one entry of the database.

## FUTURE DIRECTION

Information entry— for the datasets and associated entities —into the database will soon be completed. To ensure that the data catalog up-to-date and consistent with online and offline data copies, monitoring and notification tools will be implemented. Because this



**SEP Data Catalog:** [Home] [Maintainer Page]

[Data Provider (30)]- [Dataset (94)]- [Associated Data (77)]- [Online Copies (61)] [Offline Copies (43)] [Paper (9)]- [Researcher (19)] [Keywords (40)]

**Viewing a(n) dataset:**

FIXME Flag is **OFF**
[FIXME PAGE]

**Name:** BP-California
**Provider:** BP AMERICA INC
**Description:** British Petroleum Alaska data set from offshore S. California
**Geographic Location:**
**Year Acquired:** 0
**provider contact:**
**SEP handler:** Jesse Lomask
**License Exp. (YYYY-MM-DD):**
**proprietary considerations:**
**History:** This is 2D data used in Paul Fowlers thesis, in Fowler (1988). Came originally as shot gathers. Fowler resorted it to cmp's.
**Preprocessing:**
**Online Root:** /data/bp-ca/
[Update] [Delete]

**Related Documents:** [Add a document file]

- old catalog file [-]
- CMP100 screenshot [-]
- Min-offset screenshot [-]

**Associated Data:** [Add an associated data]

- P Velocity Model [-]

**Keywords:** [Add Keyword]

- Velocity Estimation [-]

**Related papers:** [Add paper]

- Seismic velocity estimation using prestack time migration, Ph.D [-]

Figure 3: A snapshot of the SEP data catalog website.[**NR**] altheyab2/. snapshot

database is flexible and easy to maintain, we hope that it will be continually expanded by current and future generations of SEP students.

## REFERENCES

Clapp, R. G., M. Brown, L. Vaillant, C. Mora, M. Prucha, and Y. Zhao, 1999, SEP's data library: SEP-102.

# Visualization and data reordering using EcoRAM

*Robert G. Clapp*

## ABSTRACT

Memory size and input/output (IO) performance have not kept pace with the ever increasing size of seismic data volumes. Processing steps that involve random, or pseudo-random, access to data (such as visualizing, sorting, and transposing) further degrade performance. I use Spansion's EcoRAM to replace out-of-core visualization and data transpose schemes. I show between one and two orders of magnitude improvement in performance over conventional out-of-core solutions.

## INTRODUCTION

A growing number of seismic applications are becoming IO bound. In some cases, an application can be IO bound because the amount of computation per input sample is low, such as with Normal Move-Out. Another category is applications where the input, intermediate, and output space which are to large to simultaneously reside in physical Random Access Memory (RAM), such as conventional 3-D Reverse Time Migration (RTM) (Baysal et al., 1983).

Several different methods exist to improve IO performance. Raid based disk arrays stripe data over many different disk drives. Network storage uses a collection of nodes as temporary storage. Large shared-memory and distributed-memory machines can distribute the required memory storage over 100s to 1000s of nodes, therefore allowing the entire problem to sit in RAM. Recently, solid state disks have offered promise. These disks, when used in a raid system, can offer 2-5x the IO performance of conventional disks. A new weapon in the IO bottleneck battle is EcoRAM from Spansion. EcoRAM sits in conventional RAM slots but is composed of a series of solid state disks. An accelerator replaces one of the CPU motherboard slots to facilitate low latency, read speeds of 30x conventional disk and $2-10$x write speeds.

I show that applications that are, or can be made, read dominant can benefit dramatically from EcoRAM with minimal programming changes. I begin by comparing IO characteristic of EcoRAM versus conventional RAM, disk, network, and solid state disk. I then describe how, and the results of, writing a visualization and transpose program to maximize EcoRAM performance.

## BACKGROUND

EcoRAM is a new product from Spansion. One of its goals is to fill the very large performance gap between DRAM and conventional (and solid state) disks. Its main competitor in this market place is Fusion IO. Its four most relevant claimed advantages are: $\frac{1}{8}$ power use of conventional DRAM; four times the memory per DRAM slot; $2-10$x write speed advantage over conventional disk and solid state drives; and most importantly, read speeds on the same order of magnitude as conventional DRAM (1.3 GB/s vs 4-6GB/s). The first claim, though important, would not make it a worthwhile investment at the University level. The last three claims were of more interest to us. Two commonly run applications at SEP, visualization and data transposition, are extremely IO bound for large data sizes.

### Visualization

The first example application visualization application (Clapp et al., 2008; Clapp and Nagales, 2008). Four and five dimensional spaces are common in processing. Often we are comparing multiple volumes of these sizes. For even 'small' sized problems, we quickly exceed conventional RAM. Several approaches can be taken to get around this bottleneck.

**Out-of-core** The data can be read 'on the fly'. In this approach, the desired view is created only when the user requests it. Predictive algorithms can be used to guess the next desired view. The downside of this approach is the significant difference in IO performance between RAM and disk, made worse when the user request a view that the smart algorithm didn't expect.

**Compression** The data can be compressed using techniques ranging from conversion of four to one byte to more sophisticated compression schemes such as curvelets. The downsides of these approaches include the difficulty associated with building an appropriate clip function, the need for higher order accuracy (such as seeing lower amplitude events) when interacting with the data, artifacts associated with compression, and for high compression schemes, some viewing options are difficult to achieve.

**Cluster** The data can be stored in RAM on a series of nodes. Requested views are translated into requests to different nodes and reconstructed by the host machine. The disadvantages of this scheme include the significant cost of these extra machines and latency af networking bottlenecks.

All of these approaches also suffer from a significant additional coding overhead.

### Transpose

Transposing a dataset is a common seismic processing step. A given processing step might need data in offset gathers while the data exists in common offset sections. For source-receiver based wave equation migration, the natural axis order is: midpoint, offset, and depth while further processing steps need the data in depth, offset, and midpoint. For large data volumes (significantly beyond the CPUs memory) what looks like a rather trivial operation can take hours to days using a single CPU. The transpose time is dominated for large problems by disk IO. A solution to this problem is to stream a data volume to a cluster where the transpose can be done completely in-core. This is an effective technique but again adds significant programming overhead.

## RESULTS

I used EcoRAM for the both visualization and transpose application. The server had 32 GB RAM, 512 GB of EcoRAM , and two AMD quad-core 2.6GHz processors.

### Visualization

The visualization test does not lend itself well to paper form. The only change to the basic visualization application was to introduce `mmap` buffer and IO module. The total time to add this feature was less than 3 hours. I observed that I could move to random locations within the volume with the delay more a function of the Gigabit netwrok connection between the host EcoRAM server and the X11 terminal.

### Transpose

All transposes can be mapped into a single five-dimensional template $(n1, n2, n3, n4, n5)$. The first axis is the the product of the data element size and the length of the fastest axes that are not be transposed over. The second axis is the fastest axis that needs to be transposed. The third axis is the product of the axes lengths between transposed axes. The fourth axis is the slowest axis that we want to transpose over. The fifth axis is the product of all of the axes slower than the last axis we wish to transpose over. We can extend the definition of the second and fourth axes to compose sets of adjacent axes to enable more sophisticated transposes. For example, in the case of wave equation migration, we begin with a five dimensional dataset with element size *esize* that is ordered (midpoint x, midpoint y, hx, hy, depth) or of size $(ncmpx, ncmpy, nhx, nhy, nz)$ and we wish to end up with $(nz, nhx, nhy, ncmpx, ncmpy)$. We can simulate merging the first two axes of the input and map to our template $(esize, ncmpx * ncmpy, nhx * nhy, nz, 1)$.

Using this template, the transpose algorithm can be written in a push (loop over input) or pull (loop over output) manner. For EcoRAM , and in many other cases (such as wanting to pipe the output), the pull method is more efficient. The basic transpose algorithm using the five dimensional template then takes the following form. This simple algorithm assumes that you can hold both the input and output matrices in RAM. A problem with this simple

**Algorithm 4** Simple transpose

```
iout=0;
for i5=0; i5 ¡ n5;i5++ do
  for i2=0; i2 ¡ n2;i2++ do
    for i3=0; i3 ¡ n3; i3++ do
      for i4=0; i4 ¡ n4; i4++,iout+=n1 do
        $iin = (i2 + i3 + i4 + i5) * n1 + (i3 + i4 + i5) * n2 + (i4 + i5) * n3 + i5 * n4$
        memcpy(&out[iout],&in[iin],n1)
      end for
    end for
  end for
end for
```

approach is the very poor use of input cache lines for small $n1$.

If we can hold $n1 * n2 * n3 * n4$ is memory we can get acceptable performance with either algorithm 4 or slight modification that processes each $i5$ block in turn. We could still use the basic template for large problems by `mmapping` the input and output file but the cache miss problem would be further exacerbated. A better alternative is to introduce two temporary buffers, `tin` and `tout`. These buffers are of size $n1 * n4 * n3 * nmx$, where $nmx$ is chosen so that the combined size of `tin` and `tout` does not exceed DRAM*. The buffered algorithm then takes the following form. The larger $nmx$, the better the performance. Figure 1 shows that even a RAM-based system benefits from the buffered approach. Note how we can gain a performance advantage of greater than six with larger $n1 * nbuf$ sizes.

**Algorithm 5** Simple transpose

```
for i5=0; i5 ¡ n5;i5++ do
  $i2 = 0$
  while i2¡ n2 do
    $nbuf = min(n2 - i2, nmx)$
    read tin
    iout=0;
    for i2=0; i2 ¡ nbuf;i2++ do
      for i3=0; i3 ¡ n3; i3++ do
        for i4=0; i4 ¡ n4; i4++,iout+=n1 do
          $iin = (i2 + i3 + i4) * n1 + (i3 + i4) * n2 + (i4) * n3$
          memcpy(&out[iout],&in[iin],n1)
        end for
      end for
    end for
    $i2 = i2 + nbuf$
  end while
end for
```

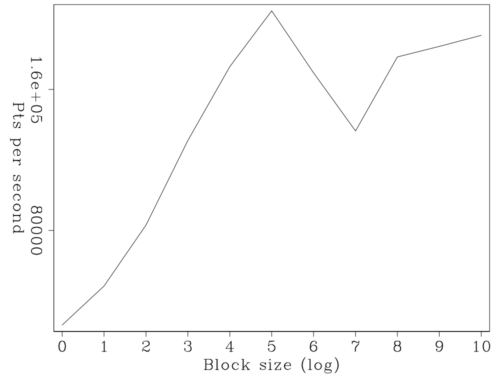


Figure 1: The number of elements per second vs $log(nbuf * n1)$ that can be read using a completely in-core solution. The problem size, using the generic template, is $(1, 500, 72, 131072, 1)$. Note how we can achieve a factor of six improvement by better cache line use. [**NR**] bob2/. ram

Figure 2 compares the performance of EcoRAM versus RAM. Note the similarity of the two curves. The '*' in the figure shows the comparable disk approach. The disk approach is $\frac{1}{10}$th speed of the slowest EcoRAM result and $\frac{1}{200}$th the optimal buffer choice.

*This approach only works if $n1 * n4 * n3 * 2$ does not exceed DRAM size.

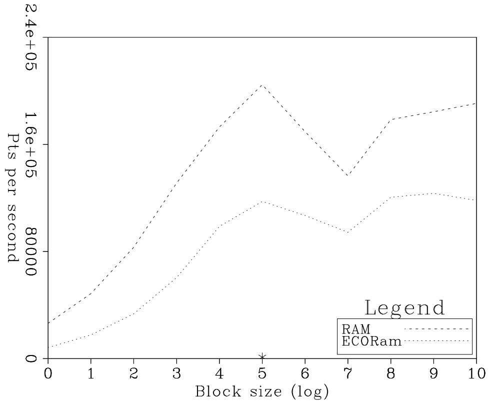


Figure 2: The number of elements per second vs. $log(nbuf * n1)$. Note the '*' indicating the disk IO performance. The problem size, using the generic template, is $(1, 500, 72, 131072, 1)$. [**NR**] bob2/. disk

As a final test, we transposed a float dataset of size $(672, 216, 72, 1, 2000)$ switching axes 1,2 with axis 5. Using the buffered approach of algorithm 5, a conventional disk took 1293 minutes (using an intermediate buffer size of 1 GB) while the same dataset took 22 minutes using EcoRAM , a $60x$ performance improvement.

## OTHER APPLICATIONS

The characteristics of EcoRAM seem best suited for large databases where reads dominate. In addition to the visualization and transpose algorithm described above, data sorting is an obvious area that could benefit from EcoRAM . Another interesting application is the ability to visualize pre-stack volumes without any preprocessing. As processing continues to outstrip IO, more applications will begin to benefit significantly from EcoRAM . Interferometry, Hessian-based inversions, and even 3-D surface related multiple attenuation, are, or soon will be, read-dominant. One can envision a future where raw data sits on an EcoRAM -like system and all processing steps are carried out on the fly, eliminating the need for the multitude of intermediate data volumes that are now common.

## CONCLUSIONS

Read-dominant applications such as visualization and data reordering are IO bound at large sizes. EcoRAM , with its low latency and high bandwidth (particularly compared to disk at small read sizes) proves to be one to two orders faster than conventional disk approaches. These speed improvements can be achieved with minimal programming changes.

## ACKNOWLEDGMENTS

I would like to thank Lou Gagliardi of Spansion for useful discussions on how to optimize application performance.

## REFERENCES

Baysal, E., D. D. Kosloff, and J. W. C. Sherwood, 1983, Reverse time migration: Geophysics, **48**, 1514–1524.

Clapp, R. G., D. M. Chen, and S. Luo, 2008, Hypercube viewer: SEP-Report, **134**, 179–192.

Clapp, R. G. and N. Nagales, 2008, Hyercube viewer: New displays and new data-types: SEP-Report, **136**, 125–130.

## SEP PHONE DIRECTORY

| **Name** | **Phone** | **Login Name** |
|---|---|---|
| Al Theyab, Abdullah | 723-0463 | altheyab |
| Ayeni, Gboyega | 723-6006 | gayeni |
| Berryman, James | – | berryman |
| Biondi, Biondo | 723-1319 | biondo |
| Cardoso, Claudio | 723-1250 | claudio |
| Claerbout, Jon | 723-3717 | jon |
| Clapp, Bob | 725-1334 | bob |
| de Ridder, Sjoerd | 723-1250 | sjoerd |
| Guitton, Antoine | – | antoine |
| Halpert, Adam | 723-6006 | adam |
| Lau, Diane | 723-1703 | diane |
| Li, Elita | 723-9282 | myfusan |
| Maysami, Mohammad | 723-9282 | mohammad |
| Moussa, Nader | 723-0463 | nwmoussa |
| Shen, Xukai | 723-0463 | xukai |
| Tang, Yaxun | 723-1250 | tang |
| Wong, Mandy | 723-9282 | mandyman |
| **SEP fax number:** | (650) 723-0683 | |

## 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, 2008-2009

| Name | Company | Tel # | E-Mail |
|---|---|---|---|
| Raymond Abma | BP | (281)366-4604 | abmar1@bp.com |
| Biondo Biondi | SEP | (650)723-1319 | biondo@sep.stanford.edu |
| Robert Bloor<br>(Co-chair, 2nd year) | ION/GX Technology | (281)781-1141 | robert.bloor@iongeo.com |
| Jon Claerbout | SEP | (650)723-3717 | jon@sep.stanford.edu |
| Richard Cook | Shell | (713)245-7195 | richard.cook@shell.com |
| Stewart A. Levin | Halliburton | (303)488-3062 | stewart.levin@halliburton.com |
| Yi Luo<br>(Co-chair, 1st year) | Saudi Aramco | (966)38738520 | yi.luo@aramco.com |

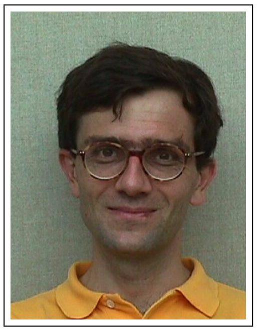
# Research Personnel

**Abdullah Al Theyab** graduated from the University of Tulsa with a double major in Geosciences and Computer Science. In 2005, he joined the Exploration Organization in Saudi Aramco working in seismic data processing and interpretation. Currently, he is pursuing a M.S. degree with Stanford Exploration Project.

**Gboyega Ayeni** received his B.Sc. in Applied Geophysics from Obafemi Awolowo University, Nigeria in 2004. He was a Shell sholar at University of Leeds, where he obtained an M.Sc with Distinction in Exploration Geophysics. Gboyega joined SEP in September 2006 to work towards his Ph.D in Geophysics. He is a member of SEG, EAGE, AGU, SPE and AAPG.

**James G. Berryman** received a B.S. degree in physics from Kansas University (Lawrence) in 1969 and a Ph.D. degree in physics from the University of Wisconsin (Madison) in 1975. He subsequently worked on seismic prospecting at Conoco. His later research concentrated on seismic waves in rocks and sediments – at AT&T Bell Laboratories (1978-81) and at Lawrence Livermore National Laboratory (1981- ), where he is currently a physicist in the Energy and Environment Directorate. He received the Maurice Anthony Biot Medal of the ASCE in May, 2005, for his work in the mechanics and acoustics of porous media containing fluids. Continuing research interests include acoustic, seismic, and electrical methods of geophysical imaging and studies of waves in porous media. He is a member of APS, AGU, ASA, and SEG.

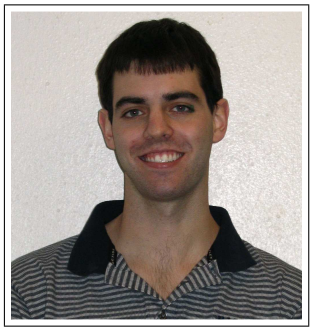
**Biondo L. Biondi** graduated from Politecnico di Milano in 1984 and received an M.S. (1988) and a Ph.D. (1990) in geophysics from Stanford. SEG Outstanding Paper award 1994. During 1987, he worked as a Research Geophysicist for TOTAL, Compagnie Francaise des Petroles in Paris. After his Ph.D. at Stanford, Biondo worked for three years with Thinking Machines Co. on the applications of massively parallel computers to seismic processing. After leaving Thinking Machines, Biondo started 3DGeo Development, a software and service company devoted to high-end seismic imaging. Biondo is now Associate Professor (Research) of Geophysics and leads SEP efforts in 3-D imaging. He is a member of SEG and EAGE.



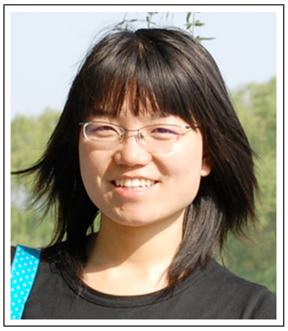
**Robert Clapp** received his B.Sc. (Hons.) in Geophysical Engineering from Colorado School of Mines in May 1993. He joined SEP in September 1993, received his Masters in June 1995, and his Ph.D. in December 2000. He is a member of the SEG and AGU.

**Claudio Guerra** received his B.Sc. in Geology from Federal University of Rio de Janeiro, Brazil in 1988 and a M.Sc. from State University of Campinas, Brazil in 1999. Since 1989, he has been working for Petrobras, Brazil. He joined SEP in 2006 and is currently pursuing a Ph.D. in geophysics at Stanford University. He is member of SEG and SBGf.

**Seth Haines** received a B.A. in Geology and Physics (double major) from Middlebury College in 1997, and began graduate work in the Stanford Geophysics Department in 1998 studying the Tibetan crust with seismic methods. He received his M.Sc. in 2001, and a Ph.D. in 2004 (SEP119) with a thesis titled ”Seismoelectric imaging of shallow targets”. He is currently a Research Geophysicist with the USGS Central Energy Resources Team in Denver, working on near-surface seismic methods for various applications including gas hydrates.



**Adam Halpert** graduated from Rice University in May 2007 with a B.Sc. in Earth Science and a B.A. in Policy Studies. He joined SEP in the summer of 2007, and is currently working toward a Ph.D. in Geophysics. He is a student member of the SEG and AGU.



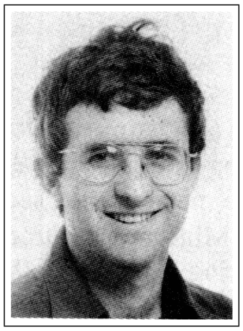
**Yunyue (Elita) Li** graduated from China University of Petroleum, Beijing in July 2008 with a B.S. in Information and Computational Science. She joined SEP in the fall of 2008, and is currently working toward a Ph.D. in Geophysics. She is a student member of the SEG.

**Mohammad Maysami** graduated in 2005 with two B.Sc. in Electrical Engineering and Petroleum Engineering from Sharif University of Technology, Tehran, Iran. After completing a M.A.Sc. degree in January 2008 in Geophysics at University of British Columbia in Vancouver, Canada, he joined SEP. He is working toward a Ph.D. in Earth, Energy, and Environmental Sciences (EEES) which is jointly supervised by SEP and SCRF (Stanford Center for Reservoir Forecasting). He is a student member of SEG, AGU, and SPE.

**Nader Moussa** graduated from North Carolina State University, where he studied Physics and Electrical and Computer Engineering. In 2008, he finished his M.Sc. at Stanford in Electrical Engineering, with focus on experimental radioscience and electromagnetic sensor instrumentation design. He is now working towards a Ph.D. in Geophysics with the Stanford Exploration Project.

**Sjoerd de Ridder** received his B.Sc. (Dec. 2004) in Earth Sciences at Utrecht University, the Netherlands. He was a graduate exchange student at the Department of Geophysics, Colorado School of Mines, USA, in the fall of 2004 and the spring of 2005. He obtained an M.Sc. (Aug. 2007) at Delft University of Technology, the Netherlands, for his thesis 'Simulation of interferometric seismoelectric Green's function recovery' in Applied Earth Sciences with specialisation in applied geophysics. He joined SEP in the fall of 2007 and is currently persuing a Ph.D. in Geophysics at Stanford University. Recipient of the "Jon F. Claerbout - Chevron Fellowship". Currently working on exploration seismology with low-frequency ambient seismic noise.

**Shuki Ronen** is a geophysicist with Geco-Prakla working on various aspects of seismic data processing and acquisition. Previously he worked with GeoQuest on seismic data interpretation; with Schlumberger on reservoir characterization; with the Institute of Petroleum Research and Geophysics on seismic data processing and acquisition; with the Colorado School of Mines as a visiting professor; and with Saxpy Computer company as an engineer. He has a Ph.D. from Stanford in Geophysics, and a B.Sc. in Physics and Geology from Hebrew University. (Photo: December 1985, Geophysics, p. 2919)



**Xukai Shen** graduated from Tsinghua University in July 2007 with a B.E. in Electrical Engineering. He joined SEP in the fall of 2007, and is currently working toward a Ph.D. in Geophysics. He is a student member of the SEG.

**Jeff Shragge** graduated in 1998 with a BScH in Honours Physics from Queen's University in Kingston, Canada. After completing a MSc degree in 2001 in teleseismic imaging at the UBC in Vancouver, Canada, he spent 2001 and 2002 working for a small geophysical consulting company based out of Woburn, MA. He joined SEP in 2002, and is working towards a Ph.D. in Geophysics. His main research interest is migration and wavefield inversion. He is a member of SEG and AGU.

**Yaxun Tang** received his B.Sc. (Jul. 2003) and M.Sc. (Aug. 2005) in Geophysics from School of Ocean and Earth Science, Tongji University, Shanghai. He joined SEP in 2005 and is currently working towards a Ph.D. in Geophysics at Stanford University. He is a member of SEG.

**Mandy Wong** graduated in 2004 with a B.Sc. in Physics and Mathematics from the University of British Columbia (UBC) in Vancouver, Canada. In 2006, she obtained a M.Sc. degree in Condensed Matter Thoery at UBC. Afterward, Mandy worked for a geophysical consulting company, SJ Geophysics, based in Vancouver, Canada. Mandy joined SEP in 2008, and is working towards a Ph.D. in Geophysics. Her main research interest is imaging with multiples.

## SEP ARTICLES PUBLISHED OR IN PRESS

Ayeni, G., A. Huck, and P. de Groot, 2008, Extending reservoir property prediction with psedo-wells: First Break. **26**, 11–15.

Biondi, B., 2007a, Angle-domain common image gathers from anisotropic migration: Geophysics, **72**, S81–S91.

———, 2007b, Residual moveout in anisotropic angle-domain common-image gathers: Geophysics, **72**, S93–S103.

Biondi, B. L., 2006, 3D Seismic Imaging: Society of Exploration Geophysicists.

———, 2007c, Concepts and Applications in 3D Seismic Imaging: Society of Exploration Geophysicists, in press.

de Ridder, S., Prieto, G.A., 2008, Seismic Interferometry and the Spatial Auto-Correlation Method on the Regional Coda of the Non-Proliferation Experiment: AGU Fall Meeting, S31A-1885.

Gray, S., D. Trad, B. Biondi, and L. Lines, 2006, Towards wave-equation imaging and velocity estimation: Recoder, **31**, 47–53.

Guitton, A., B. Kaelin, and B. Biondi, 2007, Least-squares attenuation of reverse-time-migration artifacts: Geophysics, **72**, G19–G23.

Gunther, R. and M. Reshef, 2007, Dip corrections for velocity analysis in super-gathers: J. Seis. Expl., **16**.

Haines, S., A. Guitton, and B. Biondi, 2007a, Seismoelectric data processing for surface surveys of shallow targets: Geophysics, **72**, G1–G8.

Haines, S., S. Pride, S. Klemperer, and B. Biondi, 2007b, Seismoelectric imaging of shallow targets: Geophysics, **72**, G9–G20.

Halpert, A., Clapp, R.G., Lomask, J. and B. Biondi, 2008, Image segmentation for velocity model construction and updating: SEG Technical Program Expanded Abstracts **27**, 3088-3092.

Lomask, J., R. G. Clapp, and B. Biondi, 2007, Application of image segmentation to tracking 3D salt boundaries: Geophysics, in press.

Maysami, M., and F. J. Herrmann, 2008, Lithology constraints from seismic waveforms: application to opal-A to opal-CT transition: SEG Technical Program Expanded Abstracts **27**, 2011-2015.

Rosales, D. and B. Biondi, 2006, Converted-wave azimuth moveout: Geophysics, **71**, S99–S110.

Rosales, D. A., S. Fomel, and B. Biondi, 2007, Angle-domain common-image gathers for converted-wave: Geophysics, in press.

Shragge, J., 2007, Waveform inversion by one-way wavefield extrapolation: SEG Technical Program Expanded Abstracts, **26**, 2040–2044.

———, 2008, Riemannian wavefield extrapolation: Nonorthogonal coordinate systems: Geophysics, **73**, T11–T21.

Shragge, J. and G. Shan, 2007, Prestack depth migration in elliptic coordinates: SEG Technical Program Expanded Abstracts, **26**, 2185–2189.

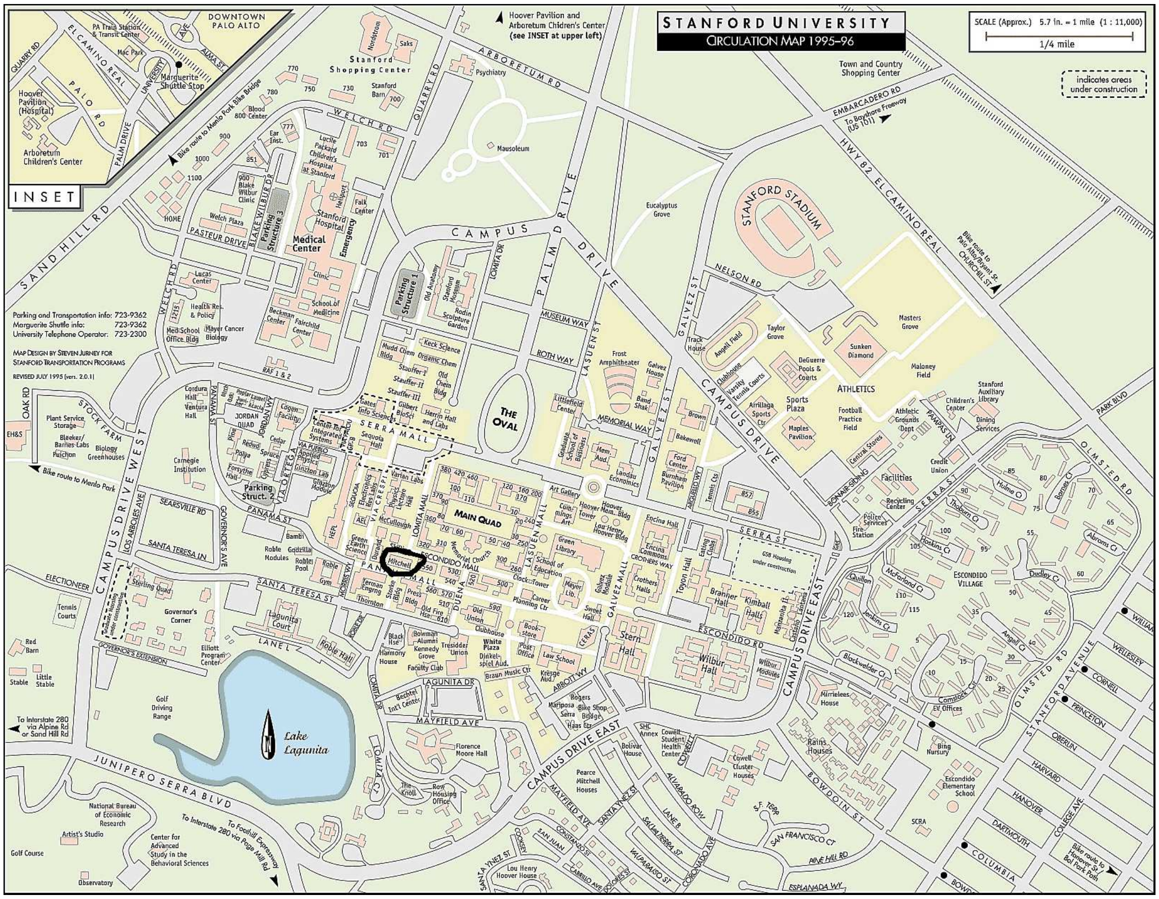
Tang, Y. and R. G. Clapp, 2006, Selection of reference-anisotropy parameters for wavefield extrapolation by Lloyd's algorithm: SEG Technical Program Expanded Abstracts **25**, 189-193.

Tang, Y., 2007, Selective stacking in the reflection-angle and azimuth domain: SEG Technical Program Expanded Abstracts **26**, 2320-2324.

———, 2008, Wave-equation Hessian by phase encoding: SEG Technical Program Expanded

Abstracts **27**, 2201-2205.

Valenciano, A., B. Biondi, and A. Guitton, 2006, Target-oriented wave-equation inversion: Geophysics, **71**, A35.


STANFORD UNIVERSITY
CIRCULATION MAP 1995–96
SCALE (Approx.) 5.7 in. = 1 mile (1 : 11,000)
1/4 mile
indicates areas under construction
INSET
DOWNTOWN PALO ALTO
PA Train Station & Transit Center
Mac Park
QUARRY RD
EL CAMINO REAL
UNIVERSITY AVE
ALMA ST.
Marguerite Shuttle Stop
PALO RD
PALM DRIVE
Hoover Pavilion (Hospital)
Arboretum Children's Center
Hoover Pavilion and Arboretum Children's Center (see INSET at upper left)
Parking and Transportation info: 723-9362
Marguerite Shuttle info: 723-9362
University Telephone Operator: 723-2300
MAP DESIGN BY STEVEN JURNEY FOR STANFORD TRANSPORTATION PROGRAMS
REVISED JULY 1995 (vers. 2.0.1)
SAND HILL RD
Bike route to Menlo Park Bike Bridge
Nordstrom
Saks
Stanford Shopping Center
770
750
730
780
Stanford Barn
700
Blood 800 Center
900
1000
1100
851
777
Ear Inst.
ARBORETUM RD
Psychiatry
QUARRY RD
WELCH RD
Lucile Packard Children's Hospital at Stanford
703
701
Mausoleum
900 Blake Wilbur Clinic
HOME
Welch Plaza
PASTEUR DRIVE
BLAKE WILBUR DR
Parking Structure 3
Heliport
Stanford Hospital
Falk Center
Medical Center
Emergency
CAMPUS DRIVE
PALM DRIVE
Lucas Center
Clinic
School of Medicine
Health Res. & Policy
1215
Med School Office Bldg
Mayer Cancer Biology
Beckman Center
Fairchild Center
RAF 1 & 2
Parking Structure 1
Old Anatomy
Stanford Museum
Rodin Sculpture Garden
LOMITA DR
MUSEUM WAY
ROTH WAY
LASUEN ST
Mudd Chem Bldg
Keck Science
Organic Chem
Stauffer I
Stauffer II
Stauffer III
Old Chem Bldg
Gates Info Sciences
Gilbert BioSci
Herrin Hall and Labs
THE OVAL
Littlefield Center
Eucalyptus Grove
OAK RD
EH&S
STOCK FARM
Plant Service Storage
Bleeker/ Barnes Labs
Puichon
Biology Greenhouses
Bike route to Menlo Park
CAMPUS DRIVE WEST
Cordura Hall
Ventura Hall
PANAMA ST
Birch
Oak
Poplar
Laurel
Juniper
Acacia
JORDAN QUAD
JORDAN WY
Cogen Facility
Pine
Redwd
Cedar
Polya
Spruce
Cypress
Forsythe Hall
Carnegie Institution
Parking Struct. 2
VIA ORTEGA
VIA PUEBLO
Applied Physics
Ginzton Lab
Ginzton Module
Center for Integrated Systems
VIA PALOU
GP-B
Sequoia Hall
SERRA MALL
Varian Labs
SEQUOIA
HEPL
Electronics Res Labs
VIA CRESPI
Physics Lecture Hall
AEL
McCullough
LOMITA MALL
Green Earth Science
Durand
Mitchell
380 420 460
100
120 160 200
170
370 90
110
1
10
360 80
MAIN QUAD
20 240
70 60
30
50 40
250
320 310
Memorial Church
300
260
ESCONDIDO MALL
550
530
540
520
500
560 570
510
Old Fire Hse 610
590
PANAMA MALL
DUENA
LASUEN MALL
Art Gallery
Cummings Art
Hoover Tower
Hoover Mem. Bldg
Lou Henry Hoover Bldg
Green Library
School of Education
Clock Tower
Career Planning Ctr
Meyer Lib.
Sweet Hall
Graduate School of Business
Mem. Aud.
Landau Economics
MEMORIAL WAY
Frost Amphitheater
Galvez House
Band Shak
GALVEZ ST
Brown
Bakewell
Ford Center
Burnham Pavilion
ARGUELLO WY
Tennis Cts
857
855
Encina Hall
Encina Commons
CROTHERS WAY
Crothers Halls
Galvez Module
GALVEZ MALL
Toyon Hall
Eating Clubs
SERRA ST
GSB Housing under construction
Branner Hall
Kimball Halls
Manzanita Pk
Castano
Lantana
CAMPUS DRIVE EAST
ESCONDIDO RD
Stern Hall
CERAS
Wilbur Hall
Wilbur Modules
SEARSVILLE RD
LOS ARBOLES AVE
SANTA TERESA LN
GOVERNOR'S AVE
Bambi
Roble Modules
Godzilla
Roble Pool
Roble Gym
MORRIS WY
Terman Engrng
Stanford Bldg
Press Bldg
Thornton
SANTA TERESA ST
ELECTIONEER
Tennis Courts
Graduate Housing under construction
Sterling Quad
Governor's Corner
Red Barn
Stable
Little Stable
GOVERNOR'S EXTENSION
Elliott Program Center
Lagunita Court
LANE L
Roble Hall
ROBLE DR
Black Hse
Harmony House
Bowman Alumni
Kennedy Grove
Faculty Club
Tresidder Union
Old Union
Clubhouse
White Plaza
Dinkelspiel Aud.
Book-store
Post Office
Law School
Braun Music Ctr
Kresge Aud.
ABBOTT WY
LAGUNITA DR
Bechtel Int'l Center
MAYFIELD AVE
LOMITA DR
Rogers
Mariposa
Serra
Bike Shop
Bridge
Haas Ctr
To Interstate 280 via Alpine Rd or Sand Hill Rd
Golf Driving Range
N
Lake Lagunita
JUNIPERO SERRA BLVD
Florence Moore Hall
LOMITA CT
The Knoll
Row Housing Office
National Bureau of Economic Research
Artist's Studio
Golf Course
Center for Advanced Study in the Behavioral Sciences
To Interstate 280 via Page Mill Rd
To Foothill Expressway
Observatory
CAMPUS DRIVE EAST
Pearce Mitchell Houses
MAYFIELD AVE
SANTA YNEZ ST
CONSTANZO ST
SAN JUAN
COOKSEY
Lou Henry Hoover House
CABRILLO AVE
DOLORES ST
SHC Annex
Bolivar House
Cowell Student Health Center
COWELL
Cowell Cluster Houses
ALVARADO ROW
LANE B
SALVATIERRA ST
VALPARAISO ST
CORONADO AVE
S.F. TERR
SAN FRANCISCO CT
PINE HILL RD
ESPLANADA WY
BOWDOIN ST
Town and Country Shopping Center
EMBARCADERO RD
To Bayshore Freeway (US 101)
HWY 82 EL CAMINO REAL
Bike route to Palo Alto/Bryant St.
CHURCHILL ST
STANFORD STADIUM
NELSON RD
Track House
Angell Field
Taylor Grove
Masters Grove
Sunken Diamond
Clubhouse
Varsity Tennis Courts
DeGuerre Pools & Courts
Maloney Field
ATHLETICS
Arrillaga Sports Ctr
Sports Plaza
Maples Pavilion
Football Practice Field
Athletic Grounds Dept
PAMPAS LN
Children's Center
Stanford Auxiliary Library
Dining Services
PARK BLVD
CAMPUS DRIVE
Central Stores
BONAIR SIDING
Facilities
Credit Union
Recycling Center
Police Services
Fire Station
SERRA ST
OLMSTED RD
85
80
70
90
Hulme Ct
Barnes Ct
75
Thoburn Ct
65
100
55
105
Hoskins Ct
95
Abrams Ct
ESCONDIDO VILLAGE
Dudley Ct
60
Quillen
McFarland Ct
50
110
115
120
Jenkins Ct
35
45
Angell Ct
40
5
15
30
Blackwelder Ct
1
10
20
25
Comstock Ct
EV Offices
Mirrielees House
Rains Houses
Bing Nursery
Escondido Elementary School
SCRA
STANFORD AVENUE
WILLIAMS
WELLESLEY
CORNELL
PRINCETON
OBERLIN
HARVARD
HANOVER
COLLEGE AVE
DARTMOUTH
COLUMBIA
BOWDOIN
Bike route to Hanover St./Bol Park Path