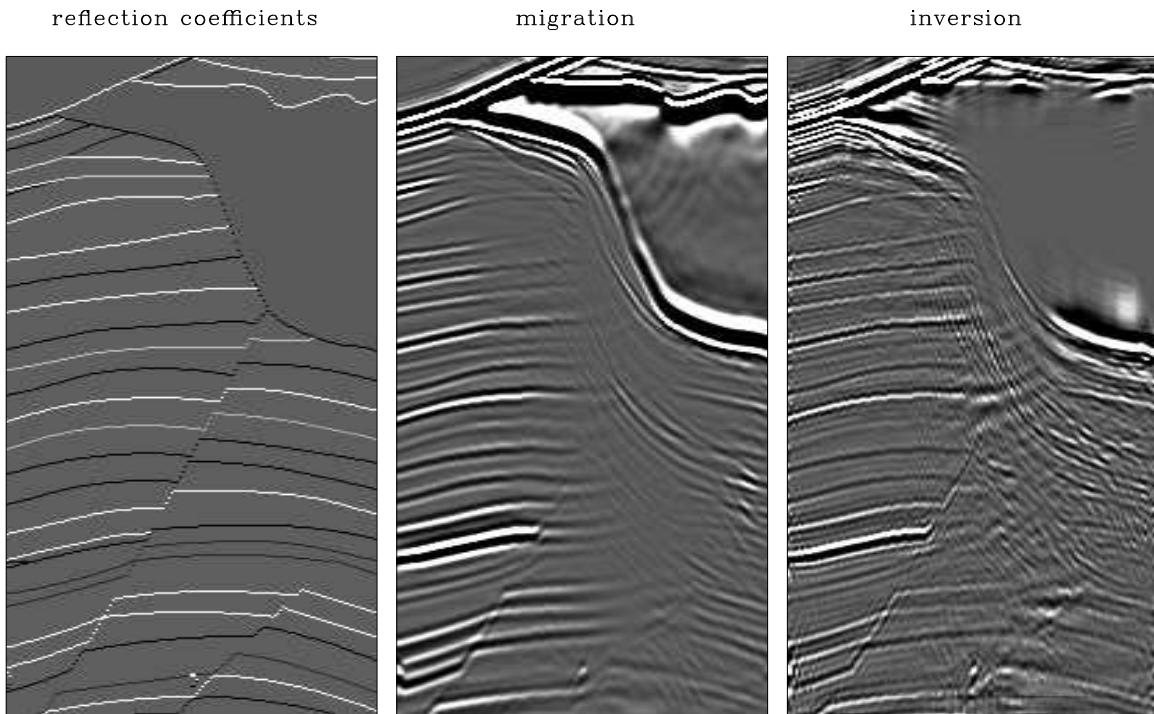


STANFORD EXPLORATION PROJECT

*James Berryman, Biondo Biondi, Robert Clapp, Haohuan Fu, Claudio Guerra,
Roland Gunther, Oskar Mencer, William Osbourne, Guojian Shan, Alejandro Valenciano,
and Francesco Zan*

Report Number 131, September 2007



Copyright © 2007

by by the Board of Trustees of the Leland Stanford Junior University

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

Preface

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

- ER** denotes Easily Reproducible and are the results of processing described in the paper. The author claims that you can reproduce such a figure from the programs, parameters, and makefiles included in the electronic document. The data must either be included in the electronic distribution, be easily available to all researchers (e.g., SEG-EAGE data sets), or be available in the SEP data library². We assume you have a UNIX workstation with Fortran, Fortran90, C, 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. SEP staff have only attempted to make sure that the makefile rules exist and the source codes referenced are provided. The primary reasons for the CR designation is that the processing requires 20 minutes or more, or commercial packages such as Matlab or Mathematica.
- M** denotes a figure that may be viewed as a movie in the web version of the report. A movie may be either ER or CR.
- 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. For more information on reproducing SEP's electronic documents, please visit
j<http://sepwww.stanford.edu/research/redoc/>.

¹<http://sepwww.stanford.edu/private/docs/sep131>

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

SEP131V2 — TABLE OF CONTENTS

Advanced imaging

<i>Alejandro A. Valenciano</i> , Target-oriented wave-equation inversion: regularization in the reflection angle	1
<i>Guojian Shan and Biondo Biondi</i> , Plane-wave migration in tilted coordinates	13

Angle gathers

<i>Guojian Shan and Biondo Biondi</i> , Angle domain common image gathers for steep reflectors	33
<i>Francesco De Zan and Biondo Biondi</i> , Phase unwrapping of angle-domain common image gathers	47
<i>Claudio Guerra</i> , Angle-domain parameters computed via weighted slant-stack	59

Velocity

<i>Roland Gunther and Biondo Biondi</i> , Ignoring density in waveform inversion	71
<i>James G. Berryman</i> , Exact seismic velocities for TI media and extended Thomsen formulas for stronger anisotropies	79

Computing

<i>Robert G. Clapp</i> , Prediction error filters to enhance differences	105
<i>Haohuan Fu and William Osbourne and Robert G. Clapp and Oskar Mencer</i> , Accelerating seismic computations using customized number representations on FPGAs	117
<i>SEP131V</i> phone directory	173
Research personnel	175
SEP131V article published or in press, 2007	??

Target-oriented wave-equation inversion: regularization in the reflection angle

Alejandro A. Valenciano¹

ABSTRACT

A complex velocity model produces shadow zones in an image due to focusing and defocusing of the seismic waves, and limited recording geometry. These shadow zones contain weak signal masked by artifacts. To recover the real signal, and reduce artifacts is necessary to go beyond migration. One option is to use a wave-equation target-oriented inversion scheme that explicitly computes the least squares inversion Hessian. The wave-equation target-oriented inversion has a big null space. It seeks to form an image where there is lack or very little data information. In this situation is where *a priori* information in the form of model regularization can help to stabilize the results. One choice for regularization, that makes physical sense, is to force the inverse image to be smooth with the reflection angle. It works by spreading the image from well illuminated to poorly illuminated reflection angles. In order to impose this smoothness constraint I implemented a chain of the subsurface-offset Hessian and a slant-stack (reflection angle to subsurface-offset) operator. Results on the Sigsbee synthetic model show that the inversion regularized in the reflection angle reduces the effect of the uneven illumination not only in the angle gathers but also in the stack image.

INTRODUCTION

Conventional imaging techniques such as migration cannot provide an accurate picture of poorly illuminated areas (Clapp, 2005). In such areas, migration artifacts or multiples can easily obscure the small amount of signal that exists, making difficult to obtain correct positioned reflectors with useful amplitudes. One reason that makes the structural image and the amplitudes unreliable in this areas is the different amount of energy illuminating the target reflectors at different angles. This is a consequence of the complexity of the subsurface and the limited acquisition geometry of the seismic experiment.

One way to improve the estimates of subsurface-acoustic properties is to use inversion (Tarantola, 1987). A linear version linking the reflectivity to the data has being applied to solve imaging problems (Nemeth et al., 1999; Kuhl and Sacchi, 2003; Clapp, 2005). This procedure computes an image by convolving the migration result with the inverse of the Hessian matrix. When the dimensions of the problem get large, the explicit calculation of the Hessian matrix and its inverse becomes unfeasible. That is why Valenciano and Biondi (2004) and Valenciano et al. (2006) proposed the following approximations: (1) to compute the one-way wave equation Green functions from the surface to the target (or vice versa); (2) to compute an approximate Hessian, exploiting its sparse structure; and (3) to compute

¹e-mail: valencia@sep.stanford.edu

the inverse image following an iterative inversion scheme. The last item renders unnecessary an explicit computation of the inverse of the Hessian matrix.

The wave-equation inversion problem has a big null space. That is why a model regularization needs to be added. Two different regularization schemes for wave-equation inversion have been discussed in the literature. First, a geophysical regularization which penalizes the roughness of the image in the offset-ray-parameter dimension (which is equivalent the reflection-angle dimension) (Prucha et al., 2000; Kuhl and Sacchi, 2003). Second, a differential semblance operator to penalize the energy in the image not focused at zero subsurface-offset (Shen et al., 2003; Valenciano, 2006, 2007).

In this paper I study the regularization in the reflection angle of the target-oriented wave-equation inversion. That choice for the regularization forces the inverse image to be smooth with the reflection angle. It works by spreading the image from well illuminated to poorly illuminated reflection angles. In order to impose this smoothness constraint I implemented a chain of the subsurface-offset Hessian and a slant-stack (reflection-angle to subsurface-offset) operator. I used the Sigsbee synthetic model to validate the methodology, showing that the inversion reduces the effect of the uneven illumination in the angle gathers and in the angle stack.

TARGET-ORIENTED WAVE-EQUATION INVERSION

Linear least-squares inversion

Tarantola (1987) formalizes the geophysical inverse problem. A linear version linking the reflectivity to the data has been discuss in the literature (Nemeth et al., 1999; Kuhl and Sacchi, 2003; Clapp, 2005). It provides a theoretical approach to compensate for experimental deficiencies (e.g., acquisition geometry, complex overburden), while being consistent with the acquired data. This approach can be summarized as follows: given a linear modeling operator \mathbf{L} , compute synthetic data \mathbf{d} using $\mathbf{d} = \mathbf{L}\mathbf{m}$ where \mathbf{m} is a reflectivity model. Given the recorded data \mathbf{d}_{obs} , a quadratic cost function,

$$S(\mathbf{m}) = \|\mathbf{d} - \mathbf{d}_{obs}\|^2 = \|\mathbf{L}\mathbf{m} - \mathbf{d}_{obs}\|^2, \quad (1)$$

is formed. The reflectivity model $\hat{\mathbf{m}}$ that minimizes $S(\mathbf{m})$ is given by the following:

$$\hat{\mathbf{m}} = (\mathbf{L}'\mathbf{L})^{-1}\mathbf{L}'\mathbf{d}_{obs} = \mathbf{H}^{-1}\mathbf{m}_{mig}, \quad (2)$$

where \mathbf{L}' (migration operator) is the adjoint of the linear modeling operator \mathbf{L} , \mathbf{m}_{mig} is the migration image, and $\mathbf{H} = \mathbf{L}'\mathbf{L}$ is the Hessian of $S(\mathbf{m})$.

The main difficulty with this approach is the explicit calculation of the inverse Hessian. In practice, it is more feasible to compute the least-squares inverse image as the solution of the linear system,

$$\mathbf{H}\hat{\mathbf{m}} = \mathbf{m}_{mig}, \quad (3)$$

by using an iterative inversion algorithm.

Regularization in the reflection angle

Equation 3 can be solved in different domains: poststack image domain (zero subsurface-offset) (Valenciano et al., 2006), prestack subsurface-offset image domain (Valenciano, 2006, 2007), or prestack reflection-angle image domain (this paper). Valenciano (2007) shows that a prestack regularization is necessary to reduce the noise in the result without smoothing the image space.

In this paper I discuss the use of the regularization in the reflection-angle domain (Prucha et al., 2000; Kuhl and Sacchi, 2003). The regularization operator is a derivative in the reflection-angle dimension that penalizes the roughness of the image. It works by spreading the image from well illuminated to poorly illuminated reflection angles.

The general fitting goals corresponding to the angle-domain inversion are:

$$\begin{aligned} \mathbf{H}(\mathbf{x}, \mathbf{h}; \mathbf{x}', \mathbf{h}') \mathbf{S}_{\Theta \rightarrow \mathbf{h}} \hat{\mathbf{m}}(\mathbf{x}, \Theta) - \mathbf{m}_{\text{ mig}}(\mathbf{x}, \mathbf{h}) &\approx 0, \\ \varepsilon \mathbf{D}_{\Theta} \hat{\mathbf{m}}(\mathbf{x}, \Theta) &\approx 0, \end{aligned} \quad (4)$$

where $\mathbf{H}(\mathbf{x}, \mathbf{h}; \mathbf{x}', \mathbf{h}')$ is the subsurface-offset Hessian (Valenciano, 2006), $\mathbf{S}_{\Theta \rightarrow \mathbf{h}}$ is a slant-stack operator that transforms the image from angle to subsurface-offset domain, \mathbf{D}_{Θ} is a derivative operator, $\mathbf{x} = (z, x, y)$ is a point in the image, $\mathbf{h} = (h_x, h_y, h_z)$ is the half subsurface-offset, and $\Theta = (\gamma, \theta)$ reflection, and azimuth angle.

In the next section I discuss on the numerical solution of the inversion problem stated in equation 4 applied to the imaging of Sigsbee model. Notice that in this paper I use a 2D example where only the h_x component of the subsurface-offset and the reflection-angle γ are used.

NUMERICAL RESULTS: SIGSBEE MODEL

The Sigsbee data set was modeled by simulating the geological setting found on the Sigsbee escarpment in the deep-water Gulf of Mexico. The model exhibits the illumination problems due to the complex salt shape, characterized by a rugose salt top (see Figure 1). Figure 2 shows the shot-profile migration image (using cross-correlation imaging condition) corresponding to the portion of Sigsbee model shown in figure 1. Notice how the amplitudes of the reflectors fade away as they get closer to the salt.

I choose a target zone close to the salt to evaluate the effects of illumination on imaging (rectangle in Figure 2). A good picture of the complexity of the focusing and defocusing of the seismic energy in this model is given by Figure 3, which shows the diagonal of the Hessian matrix in the target zone. Light gray correspond to hight amplitude and dark gray to low amplitudes. Notice how the concave and convex shape of the base of the salt, respectively, focus and defocus the seismic energy as waves propagate trough the medium.

Four rows of the target-oriented Hessian matrix are shown in Figure 4. They are $11 \times 21 \times 17$ coefficient filter for constant depth and constant subsurface-offset ($h_x = 0 \text{ ft}$) at four different x coordinates (from the sediments to the salt boundary Figure 3). Notice that only the elements of the matrix corresponding to one side of the diagonal are shown. Since the Hessian matrix is symmetric by definition half of the off diagonal terms are not computed. Figure 4a shows point 1, with coordinates $\mathbf{x} = (14000, 31000) \text{ ft}$ (far from the

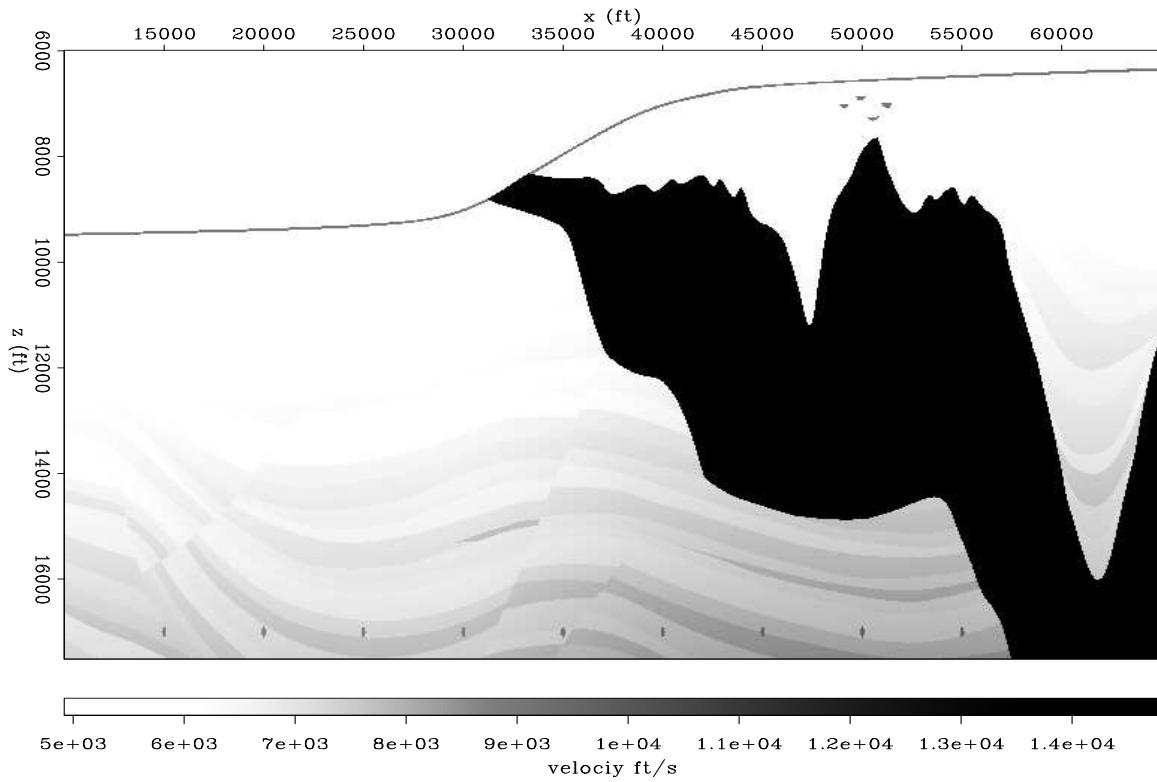


Figure 1: Sigsbee stratigraphic velocity model. alejandro1/. Sisvel

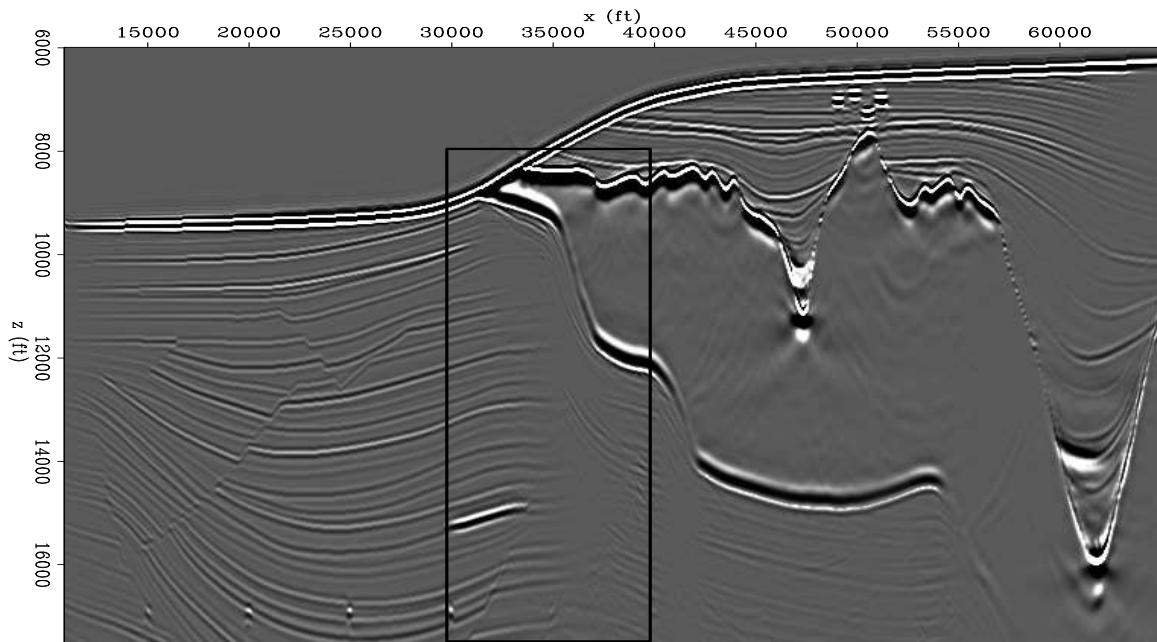


Figure 2: Sigsbee shot-profile zero subsurface-offset migration image using cross-correlation imaging condition. The velocity model corresponds to Figure 1 alejandro1/. migSis

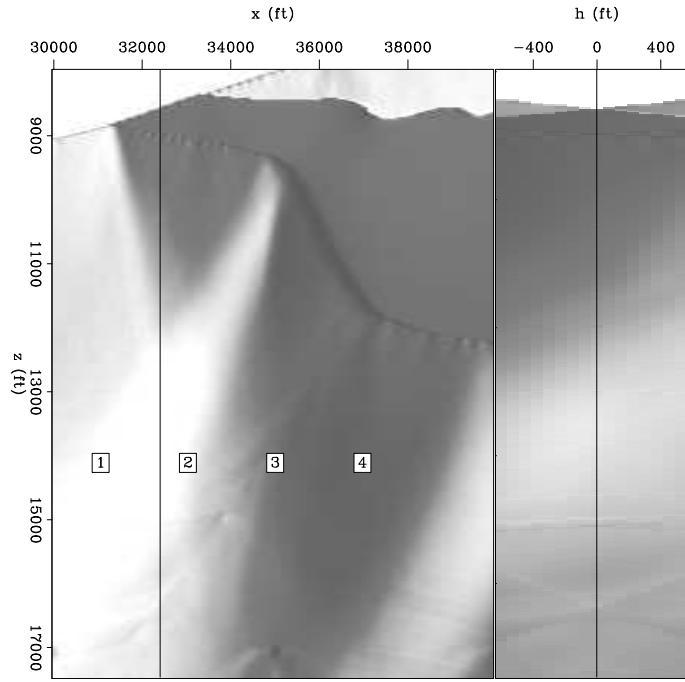


Figure 3: Diagonal of the target-oriented Hessian matrix. White correspond to hight amplitude `alejandro1/. diag`

salt). Figure 4b shows point 2, with coordinates $\mathbf{x} = (14000, 33000) \text{ ft}$. Figure 4c shows point 3, with coordinates $\mathbf{x} = (14000, 35000) \text{ ft}$. Figure 4d shows point 4, with coordinates $\mathbf{x} = (14000, 37000) \text{ ft}$. The shape of the filter is not dependent only on the acquisition geometry but the subsurface geometry (presence of the salt body). In the area less affected by the salt the energy is concentrated around the diagonal (center of the filter), but as we get closer to the salt, the illumination varies (in intensity and angle) and the filter behaves differently. This is due to a focusing and defocusing effect created by the salt. To correct this effect we computed the least-squares inverse image.

Postack comparison

Figure 5 shows the reflection coefficients, and the zero subsurface-offset migration. The zero subsurface-offset inversion (Valenciano et al., 2006), and the stack of the inversion with regularization in the reflection angle can be seen in Figure 6. In the migration result shown in Figure 5b the reflectors dim out in the areas of low illumination (see left panel of Figure 3 for reference). In contrast, the zero subsurface-offset inversion (Figure 6a) and the stack of the inversion with regularization in the reflection angle (Figure 6b) show that: the reflectors can be followed into the shadow zones with the correct kinematics, the resolution increases, the footprint of the irregular illumination is diminished, and the faults can be followed and interpreted closer to the salt body.

It is important to remark the differences between the two inversion results in Figure 6a and Figure 6b. The inversion with regularization in the reflection angle has better defined fault planes and sediments more accurately extended into the shadow zones than the zero

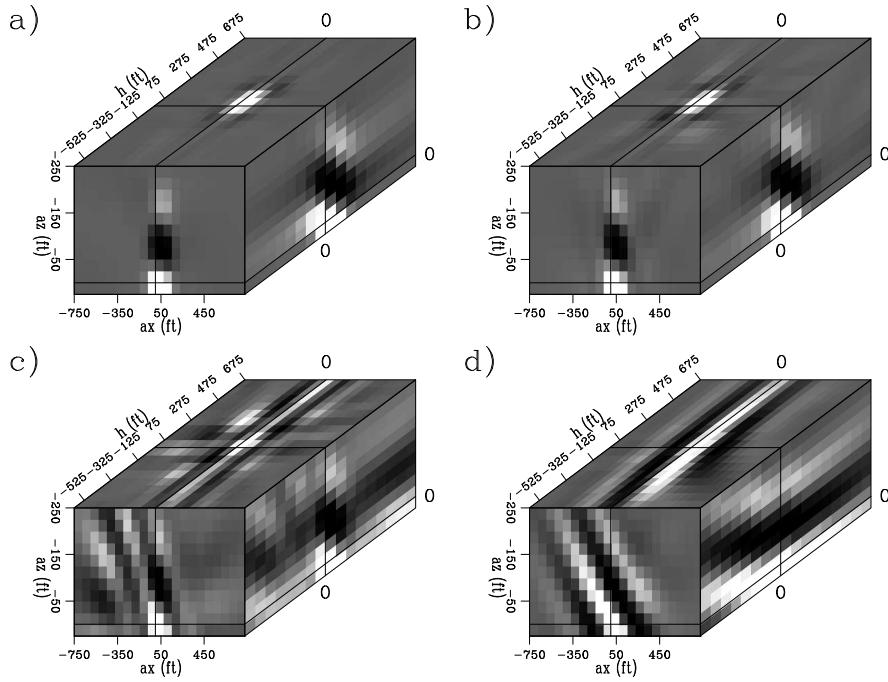


Figure 4: Four rows of the target-oriented Hessian matrix, (a) point 1 $\mathbf{x} = (14000, 31000)$ ft, (b) point 2 $\mathbf{x} = (14000, 33000)$ ft, (c) point 3 $\mathbf{x} = (14000, 35000)$ ft, and (d) point 4 $\mathbf{x} = (14000, 37000)$ ft. alejandro1/. hesianphaseSis

subsurface-offset inversion. Also, the level of noise in the zero subsurface-offset inversion is much higher. This is due to the fact that no regularization was applied in the zero subsurface-offset inversion. The regularization in the reflection angle helps to spread the image from well illuminated to poorly illuminated reflection angles, reducing the noise and eliminating non consistent artifacts.

The salt in the inversion images looks distorted because a residual weight designed to decrease the salt contribution was used (data values in the salt boundary are bigger than everywhere else Figure 2). This was necessary to avoid the solver expending most of the iterations decreasing the residuals in that area.

Prestack comparison

Figures 7 and 8 show the migration result in the subsurface-offset domain and the reflection-angle domain, respectively. The migration in the subsurface-offset domain is the "data", and corresponds to the right hand side of equation 4. The migration at the reflection-angle domain is shown for comparison purposes, since the model space corresponds to the reflection-angle domain.

Figures 9 and 10 show the inversion without and with regularization in the reflection-angle domain, respectively (compare to migration in Figure 8). The left panel shows a common angle section (24°). The migration shows a big shadow zone below the salt (Figure 8). In the inversion without regularization (Figure 9) the shadow zone has been filled

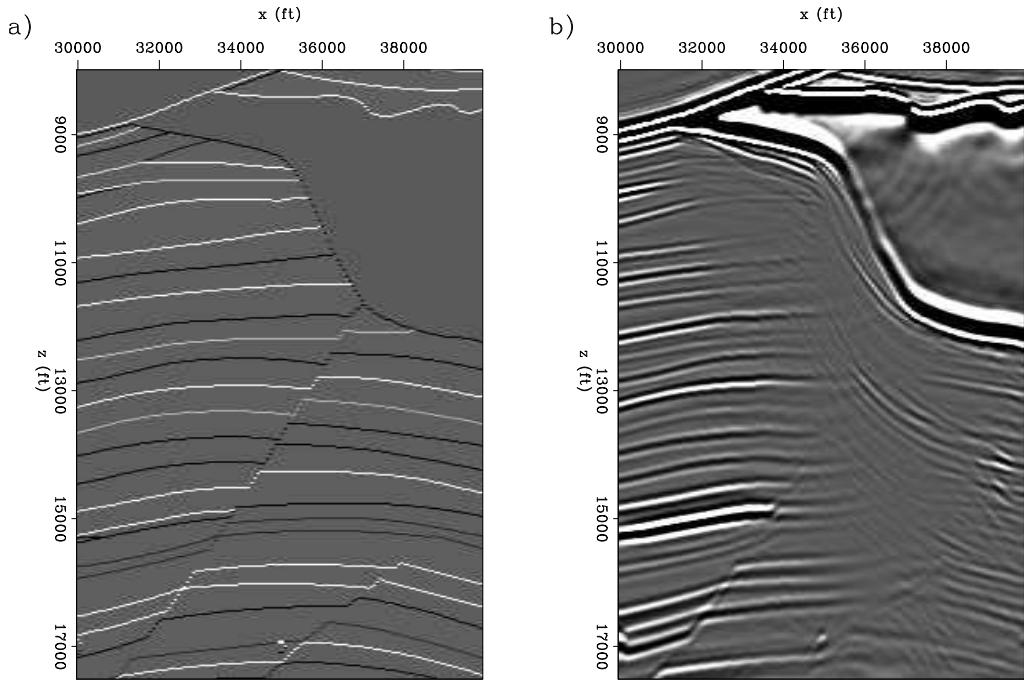


Figure 5: Target area comparison. (a) reflection coefficients, and (b) zero subsurface-offset migration. alejandro1/. compSisfull1

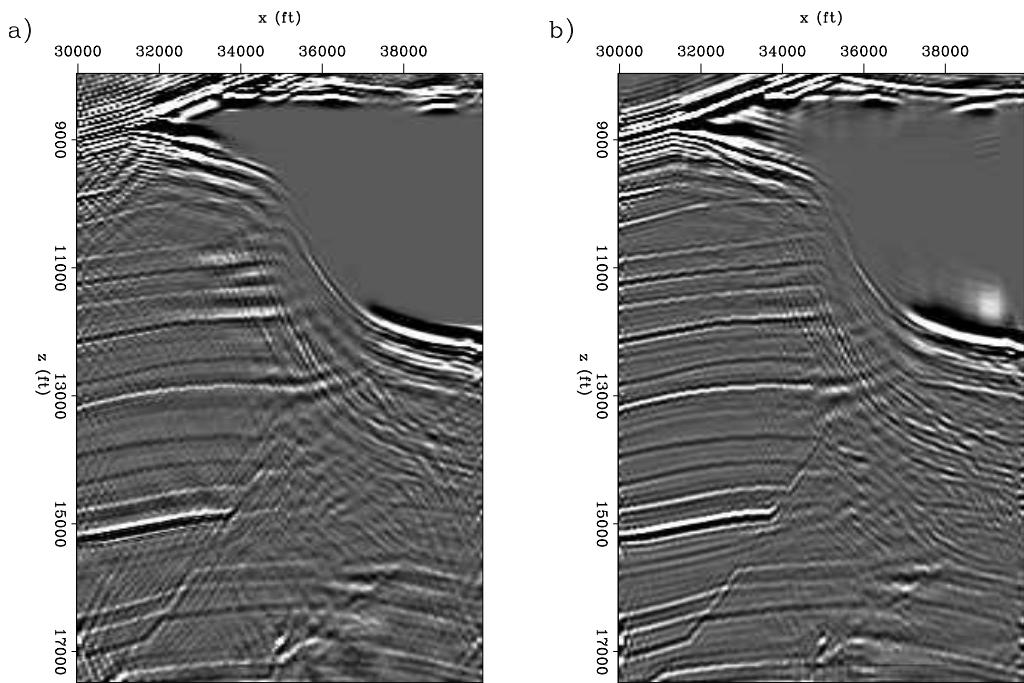


Figure 6: Target area comparison. (a) zero subsurface-offset inversion, and (b) stack of the inversion with regularization in the reflection angle (zero subsurface-offset). alejandro1/. compSisfull2

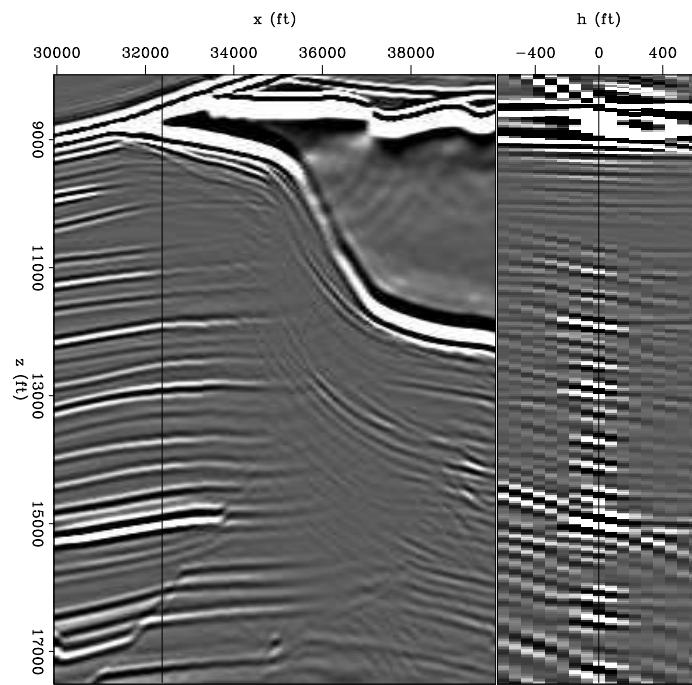


Figure 7: Sigsbee shot-profile migration(subsurface-offset) using cross-correlation imaging condition. alejandro1/. migoff

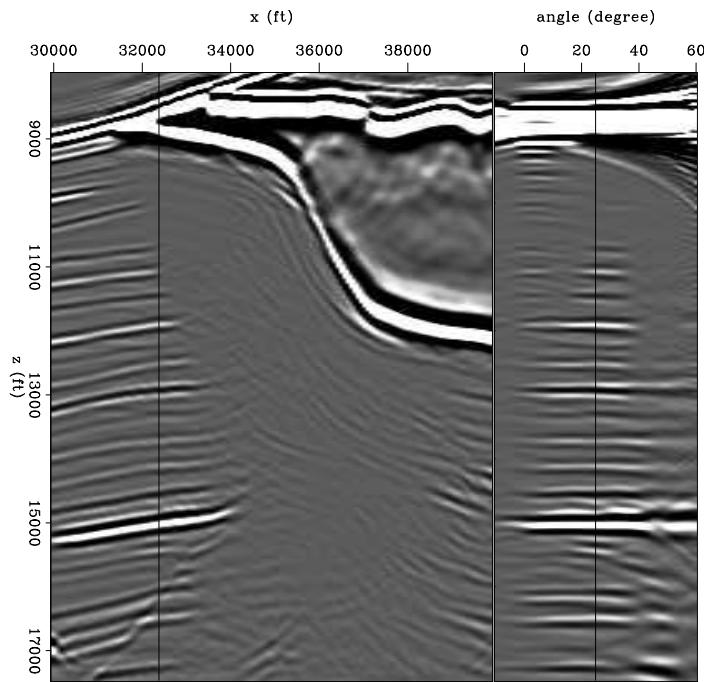


Figure 8: Sigsbee shot-profile migration (reflection angle) using cross-correlation imaging condition. alejandro1/. migang

partially but the results are very noisy. The regularized inversion image (Figure 10) gives a better result, less noisy and with some of the reflectors extended into the shadow zone.

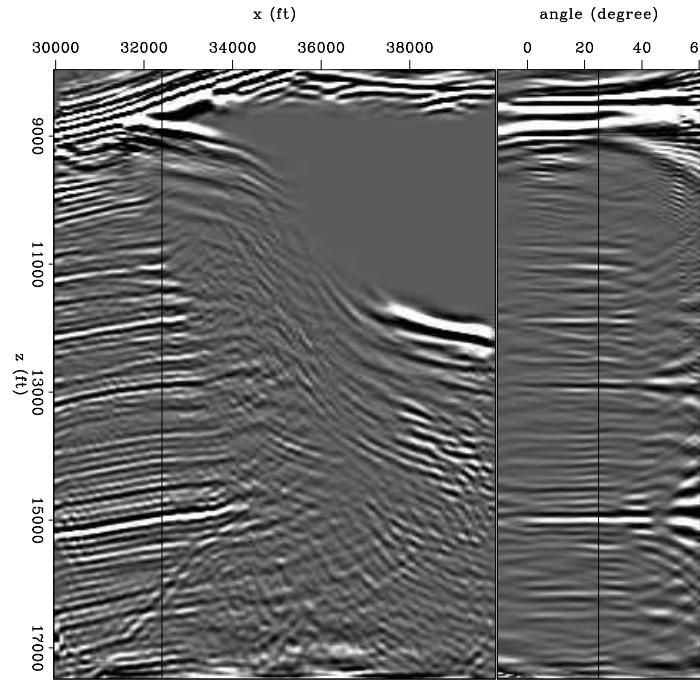


Figure 9: Sigsbee inversion without regularization (reflection angle). alejandro1/. invang

If we look in more detail into the angle gathers (Figures 11, 12, and 13) the effect of the inversion and the regularization can be understood separately. Figure 11 shows angle gathers at three different x coordinate positions $x = 32300 \text{ ft}$, $x = 33700 \text{ ft}$, $x = 35700 \text{ ft}$. Notice the holes in some of the reflectors, also notice that there are salt related multiple reflections (non-flat events in the angle gathers). After inversion without regularization (Figure 12) some the wholes have been filled but the noise has increase as well as the bandlimited related artifacts off the slant-stack, also the multiples had been increased in amplitude. The inversion with regularization (Figure 13), gives the best result. The wholes have being filled, the noise is reduced, and the far-angle-multiple energy is decreased.

CONCLUSIONS

The target-oriented wave-equation inversion, regularized in the reflection angle, reduces the effect of the uneven illumination not only in the angle gathers but also in the stack image. It gives better results than the zero subsurface-offset inversion (Valenciano et al., 2006) because the regularization helps to spread the image from well illuminated to poorly illuminated reflection angles, reducing the noise and eliminating non consistent artifacts. Results in Sigsbee data set show that even with very complex subsalt illumination the inversion can give a good image.

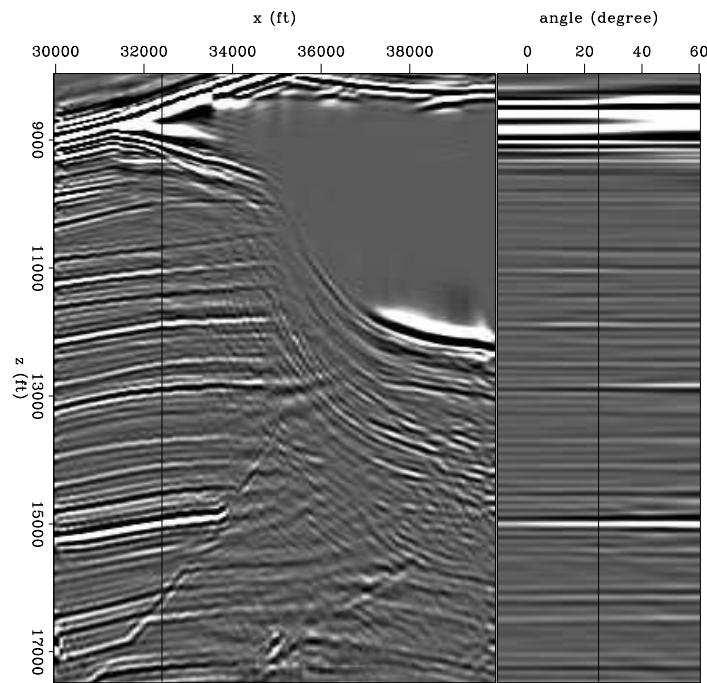


Figure 10: Sigsbee inversion with regularization (reflection angle). alejandro1/. invregang

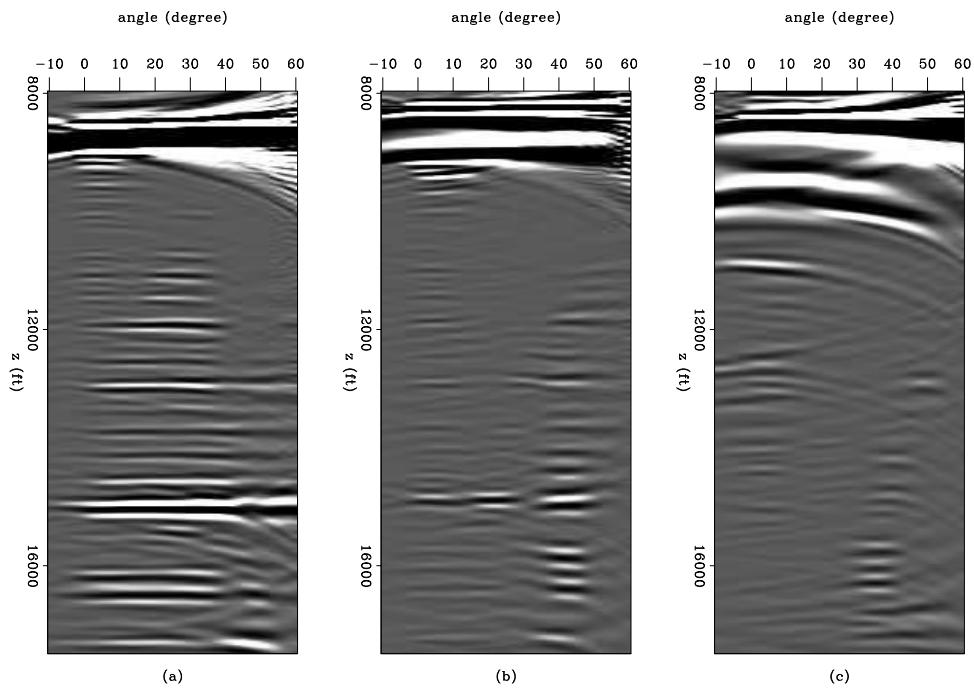


Figure 11: Sigsbee migration (reflection angle). Angle gathers at three different x coordinate positions (a) $x = 32300$ ft, (b) $x = 33700$ ft, and (c) $x = 35700$ ft alejandro1/. compmigang

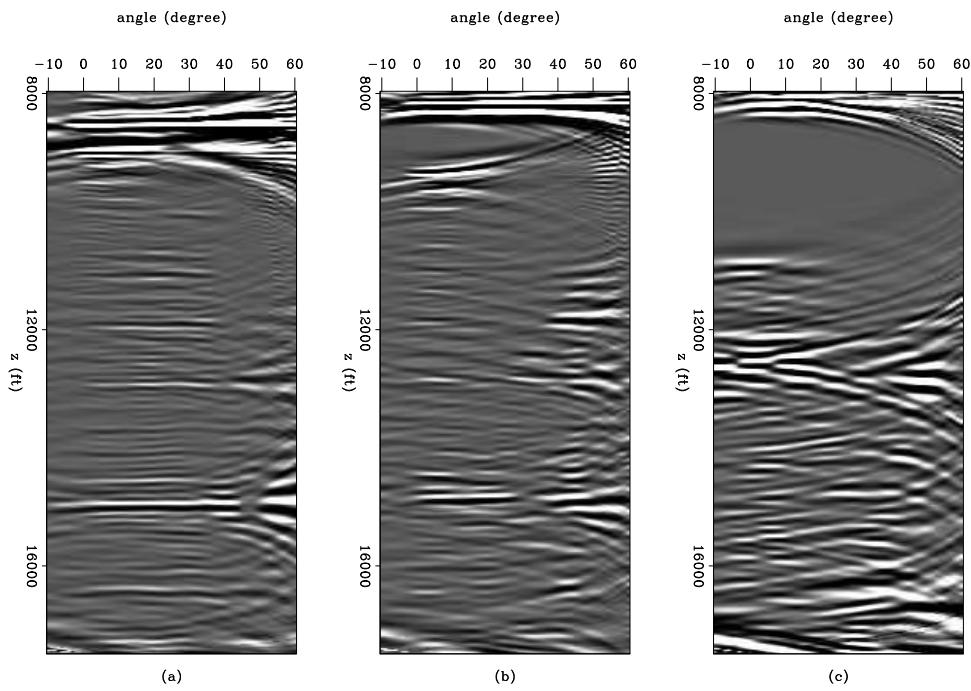


Figure 12: Sigsbee inversion without regularization (reflection angle). Angle gathers at three different x coordinate positions (a) $x = 32300 \text{ ft}$, (b) $x = 33700 \text{ ft}$, and (c) $x = 35700 \text{ ft}$

`alejandro1/. compinvang`

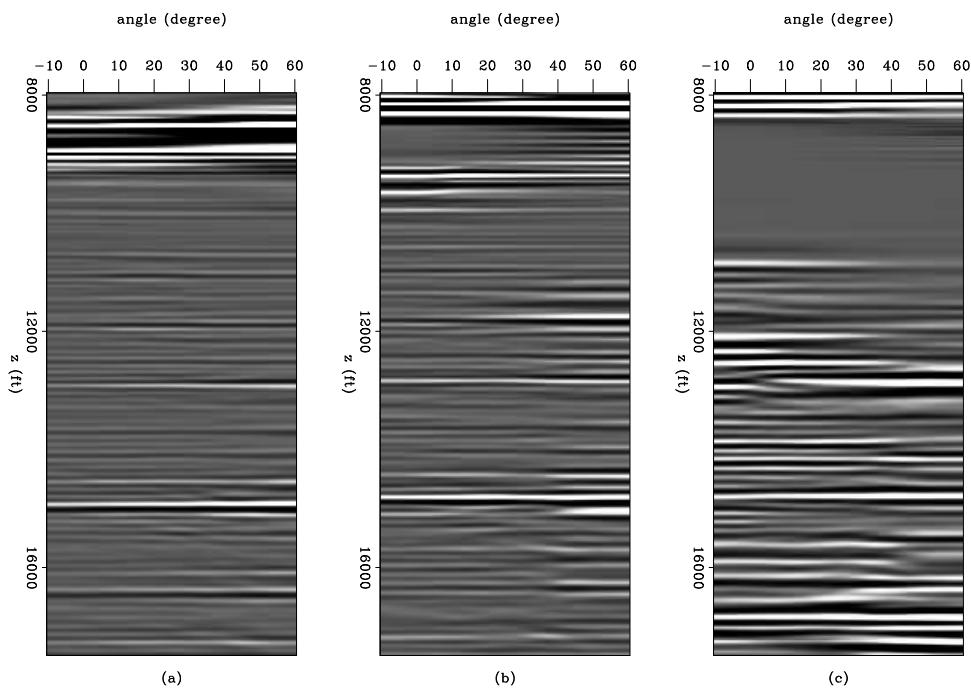


Figure 13: Sigsbee inversion with regularization (reflection angle). Angle gathers at three different x coordinate positions (a) $x = 32300 \text{ ft}$, (b) $x = 33700 \text{ ft}$, and (c) $x = 35700 \text{ ft}$

`alejandro1/. compinvregang`

ACKNOWLEDGMENTS

I would like to thank the SMAART JV for the synthetic data used in the experiments, as well as the Stanford Exploration Project sponsors for financial and technical support.

REFERENCES

- Clapp, M. L., 2005, Imaging under salt: illumination compensation by regularized inversion: PhD thesis, Stanford University.
- Kuhl, 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.
- Prucha, M. L., R. G. Clapp, and B. Biondi, 2000, Seismic image regularization in the reflection angle domain: *SEP-Report*, **103**, 109–119.
- Shen, P., W. Symes, and C. C. Stolk, 2003, Differential semblance velocity analysis by wave-equation migration: 73st Annual International Meeting, SEG, Expanded Abstracts, Expanded Abstracts, 2132–2135.
- Tarantola, A., 1987, Inverse problem theory: Methods for data fitting and model parameter estimation: Elsevier Science Publication Company, Inc.
- Valenciano, A. A., 2006, Target-oriented wave-equation inversion with regularization in the subsurface offset domain: *SEP-Report*, **124**, 85–94.
- , 2007, Target-oriented wave-equation inversion: Sigsbee model: *SEP-Report*, **129**, 65–74.
- Valenciano, A. A. and B. Biondi, 2004, Target-oriented computation of the wave-equation imaging Hessian: *SEP-Report*, **117**, 63–76.
- Valenciano, A. A., B. Biondi, and A. Guitton, 2006, Target-oriented wave-equation inversion: *Geophysics*, **71**, A35–A38.

Plane-wave migration in tilted coordinates

Guojian Shan and Biondo Biondi¹

ABSTRACT

Most existing one-way wave-equation migration algorithms have difficulty in imaging steep dips in a medium with strong lateral velocity variation. We propose a new one-way wave-equation-based migration, called “plane-wave migration in tilted coordinates.” The surface data are converted to plane-wave source data by slant-stacking processing, and each resulting plane-wave source dataset is migrated independently in a tilted coordinate system with an extrapolation direction determined by the source plane-wave direction at the surface. For most waves illuminating steeply dipping reflectors, the extrapolation direction is closer to their propagation direction in the tilted coordinates. Therefore, plane-wave migration in tilted coordinates can correctly image steeply dipping reflectors, even by applying one-way extrapolators. In a well-chosen tilted coordinate system, waves that overturn in conventional vertical Cartesian coordinates do not overturn in the new coordinate system. Using plane-wave migration in tilted coordinates, we can image overturned energy with much lower cost compared to reverse-time migration.

INTRODUCTION

Kirchhoff migration has been widely applied in seismic processing due to its relatively low cost and flexibility. However, it cannot provide reliable images where multi-pathing occurs. Wave-equation migration, which is performed by recursive wavefield extrapolation, has been demonstrated to overcome these limitations and produce better images in areas of complex geology.

It is well known that in a single-shot experiment waves propagate upward and downward simultaneously. Reverse-time migration (Whitmore, 1983; Baysal et al., 1983; Biondi and Shan, 2002), which solves the full wave equation directly and mimics wave propagation naturally, is expensive for routine use in today’s computing facilities. As a consequence, downward continuation migration Claerbout (1985), which are based on one-way wave-equation wavefield extrapolation and are much cheaper than reverse-time migration, are widely used in the industry.

Conventional downward-continuation methods extrapolate wavefields using the one-way wave equation in vertical Cartesian coordinates. For a medium without lateral velocity variation, the phase-shift method (Gazdag, 1978) can be applied, and the one-way wave-equation can model waves propagating in a direction up to 90° away from the extrapolation direction. But in a laterally varying medium, it is very difficult to model waves propagating in a direction far from the extrapolation direction using a one-way wavefield extrapolator.

¹e-mail: valencia@sep.stanford.edu

Many methods have been developed to improve the accuracy of the one-way wavefield extrapolator in laterally varying media, such as Fourier finite-difference (Ristow and Ruhl, 1994; Biondi, 2002), the general screen propagator (de Hoop, 1996; Huang and Wu, 1996) and optimized finite difference (Lee and Suh, 1985) with a phase correction (Li, 1991). Even if we could model waves accurately up to 90° using the one-way wavefield extrapolator in laterally varying media, overturned waves, which travel downward first and then curve upward, are filtered away during the wavefield extrapolation because of the assumption that the waves propagate vertically only in one direction: downward for source wavefields and upward for receiver wavefields. However, overturned waves and waves propagating at high angles play a key role in imaging steeply dipping reflectors, such as salt flank and faults. As a consequence, imaging these steeply dipping reflectors remains a major problem in downward continuation migration.

Work has been done to image the steeply dipping reflectors with one-way wavefield extrapolators by coordinate transformation. This includes tilted coordinates (Higginbotham et al., 1985; Etgen, 2002), the combination of downward continuation and horizontal continuation (Zhang and McMechan, 1997), or wavefield extrapolation in general coordinates, such as ray coordinates (Nichols, 1994) and Riemannian coordinates (Sava and Fomel, 2005; Shragge, 2006).

In tilted coordinates, waves traveling along the extrapolation direction are most accurately modeled, and the maximum angle of their propagation direction from the extrapolation direction that can be handled is determined by the accuracy of the wavefield extrapolator. For a point source, where waves travel in all directions from a point, it is impossible for one tilted coordinate system to cover all these directions. But for a plane-wave source, waves travel in a similar direction from all spacial points at the surface, and thus most of them can be modeled accurately in a tilted coordinate system with a well-chosen tilting direction. In this paper, we apply plane-wave migration (Whitmore, 1995; Rietveld, 1995; Duquet et al., 2001; Liu et al., 2002; Zhang et al., 2005) in tilted coordinates. Plane-wave migration has been demonstrated to be a useful tool in seismic imaging. By slant-stacking, the recorded surface data are synthesized into areal plane-wave-source gathers, which are what would be recorded if plane-wave sources were excited at the surface. A plane-wave source is characterized by a ray parameter, and its take-off angle can be calculated from the ray parameter, given the velocity at the surface. Each areal plane-source gather is migrated independently, similar to shot-profile migration, and the image is formed by stacking the images of all possible plane-wave sources. Given a plane-wave source, we tilt the coordinate system according to its take-off angle. For most waves, the resulting extrapolation direction is closer to the propagation direction, and thus we can image steeply dipping reflectors correctly using one-way wavefield extrapolators. Plane-wave migration is potentially more efficient than shot-profile migration (Zhang et al., 2005; Etgen, 2005). To image steeply dipping reflectors or overturned waves, a large migration aperture is required to cover the whole propagation path of source and receiver waves. In shot-profile migration, this requires large padding in space. In contrast, plane-wave migration uses the whole seismic survey as the migration aperture. It is well known that one-way wave-equation shot-profile migration is much cheaper than reverse-time migration. Compared to conventional plane-wave migration, the cost of plane-wave migration in tilted coordinates is a little higher because of the data and velocity model interpolation, but it is still much lower than reverse-time migration.

This paper is organized as follows: we begin with a brief review of one-way wave-equation migration and plane-wave migration. Then we introduce how to extrapolate the wavefield in a tilted coordinate system and describe plane-wave migration in tilted coordinates. Finally, we demonstrate our technique with synthetic data examples.

ONE-WAY WAVE EQUATION MIGRATION

Surface seismic data are usually recorded as shot gathers. Each shot gather represents a point-source exploding experiment. The most straightforward way to obtain the subsurface image of the earth is shot-profile migration, in which we obtain the local image of each experiment by migrating each shot gather independently and form the whole image of the subsurface by stacking all the local images. Migrating one shot gather using a typical shot-profile migration algorithm includes two steps. First, source and receiver wavefields are extrapolated from the surface to all depths in the subsurface. Second, the images are constructed by cross-correlating the source and receiver wavefields.

The propagation of waves in the subsurface is approximately governed by a two-way acoustic wave equation. In an isotropic medium, it is defined as follows:

$$\frac{1}{v^2} \frac{\partial^2}{\partial t^2} P = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} \right) P, \quad (1)$$

where $P = P(x, z, t)$ is the pressure field and $v = v(x, z)$ is the velocity of the medium. To reduce computational costs, we usually use the one-way instead of two-way wave equations for wavefield extrapolation:

$$\frac{\partial}{\partial z} S = -\frac{i\omega}{v} \sqrt{1 + \left(\frac{v}{\omega} \frac{\partial}{\partial x} \right)^2} S, \quad (2)$$

$$\frac{\partial}{\partial z} R = +\frac{i\omega}{v} \sqrt{1 + \left(\frac{v}{\omega} \frac{\partial}{\partial x} \right)^2} R, \quad (3)$$

for wavefield extrapolation, where ω is angular frequency, $S = S(s_x, x, z, \omega)$ is the source wavefield, $R = R(s_x, x, z, \omega)$ is the receiver wavefield, and s_x is the source location. Given the propagation direction of the source and receiver wavefields, we use the down-going one-way wave equation (equation 2) for the source wavefield and the up-going one-way wave equation (equation 3) for the receiver wavefield. Both are obtained by splitting the two-way acoustic equation (Zhang, 1993). After the wavefield extrapolation, we have the source and receiver wavefields at all depths and the image is constructed by cross-correlating the source and receiver wavefields as follows:

$$I_{s_x} = \int S^*(s_x, x, z, \omega) R(s_x, x, z, \omega) d\omega, \quad (4)$$

where S^* is the complex conjugate of the source wavefield S . Finally the whole image is generated by stacking the images of all the shots as follows:

$$I = \int I_{s_x} ds_x. \quad (5)$$

If there is no lateral velocity variation, equations 2 and 3 can be solved by the phase-shift method in the frequency-wavenumber domain with accuracy up to 90°. Otherwise, an

approximation for the square root operator has to be made to solve equations 2 and 3 numerically. The accuracy of a wavefield extrapolator determines the maximum angle between the propagation direction and the vertical direction that can be modeled accurately. most algorithms can model waves that propagate almost vertically downward. For example, the classic 15° equation (Claerbout, 1971) can handle waves propagating 15° from the vertical direction. However, most algorithms cannot model waves propagating almost horizontally in a medium with strong lateral variation. Finite-difference methods handle lateral variation of the media well, but the cost of improving the accuracy at high angles is high. Hybrid algorithms such as Fourier finite-difference take advantage of both the finite-difference and phase-shift methods. When the lateral variation of the medium is mild, phase-shift plays the important role and can achieve good accuracy. The finite-difference part becomes more important where the actual velocity value is far from the reference velocity, but again is difficult to propagate high-angle energy accurately with a reasonable cost. It is difficult to solve the one-way wave equation accurately to model high-angle energy in a medium with strong lateral variation.

One-way wave equations also function as dip filters. During the source wavefield extrapolation, only the down-going energy is permitted using the down-going one-way wave equation; up-going energy is filtered out. Similarly, the down-going energy is filtered out during the receiver wavefield extrapolation. Therefore, overturned energy is filtered out in both source and receiver wavefields in conventional downward continuation migration.

Conventional downward continuation migration is not sufficient for imaging steeply dipping reflectors, since they are mainly illuminated by high-angle and overturned energy. These are the two main migration issues that we attempt to resolve with plane-wave migration in tilted coordinates.

PLANE-WAVE SOURCE MIGRATION

Shot gathers can also be synthesized into a new dataset to represent a physical experiment that is not performed in reality. One of the most important examples is to synthesize shot gathers into plane-wave source gathers. A plane-wave source gather represents what would be recorded if a planar source were excited at the surface with geophones covering the whole area. It can also be regarded as the accurate phase-encoding of the shot gathers (Liu et al., 2002). Plane-wave source gathers can be generated by slant-stacking receiver gathers. The process can be described as follows:

$$R_p(p_x, r_x, z = 0, \omega) = \int R(s_x, r_x, z = 0, \omega) e^{i\omega s_x p_x} ds_x, \quad (6)$$

where p_x is the ray parameter for the x -axis, s_x is the source location, and r_x is the receiver location at the surface. Its corresponding plane-wave source wavefield at the surface is

$$S_p(p_x, r_x, z = 0, \omega) = e^{i\omega r_x p_x}. \quad (7)$$

As with the Fourier transformation, we can transform the plane-wave source gathers back to shot gathers by inverse slant-stacking (Claerbout, 1985) as follows:

$$R(s_x, r_x, z = 0, \omega) = \int \omega R_p(p_x, r_x, z = 0, \omega) e^{-i\omega s_x p_x} dp_x. \quad (8)$$

In contrast to the inverse Fourier transformation, the kernel of the integral is weighted by the angular frequency ω . This inverse transformation weighting function is also called ρ filter in Radon-transform literature.

As with shot-profile migration, there are two steps to migrate a plane-wave source gather by a typical plane-wave migration method. First, the source wavefield S_p and receiver wavefield R_p are extrapolated into all depths in the subsurface independently, using the one-way wave equations 2 and 3, respectively. Second, the image of a plane-wave source with a ray parameter p_x is constructed by cross-correlating the source and receiver wavefields weighted with the angular frequency ω :

$$I_{p_x}(x, z) = \int \omega S_p^*(p_x, x, z, \omega) R_p(p_x, x, z, \omega) d\omega, \quad (9)$$

where S_p^* is the conjugate complex of the source wavefield S_p . The whole image is formed by stacking the images of all possible plane-wave sources:

$$I_p = \int \int I_{p_x}(x, z) dp_x. \quad (10)$$

Because both slant-stacking and migration are linear operators, the image of the plane-wave migration I_p is equivalent to the image obtained by shot-profile migration (Liu et al., 2002; Zhang et al., 2005). In the discrete form, in practice we need a sufficient number of p_x to make the two images equivalent.

WAVEFIELD EXTRAPOLATION IN TILTED COORDINATES

The extrapolation direction plays a key role in one-way wave-equation wavefield extrapolation, since the waves traveling along the extrapolation direction are modeled the most accurately. However, the extrapolation direction has no physical meaning and it is only a direction artificially assigned in numerical algorithms. In conventional downward continuation migration, we use vertical Cartesian coordinates and extrapolate wavefields vertically. The extrapolation direction can be changed by rotating the coordinates. It is well known that the acoustic equation (equation 1) is invariant to coordinate rotations as follows:

$$\begin{pmatrix} x' \\ z' \end{pmatrix} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ z \end{pmatrix}. \quad (11)$$

We call the new coordinate system (x', z') a tilted Cartesian coordinate system (or tilted coordinate system) and the angle θ the tilting angle for the coordinate system.

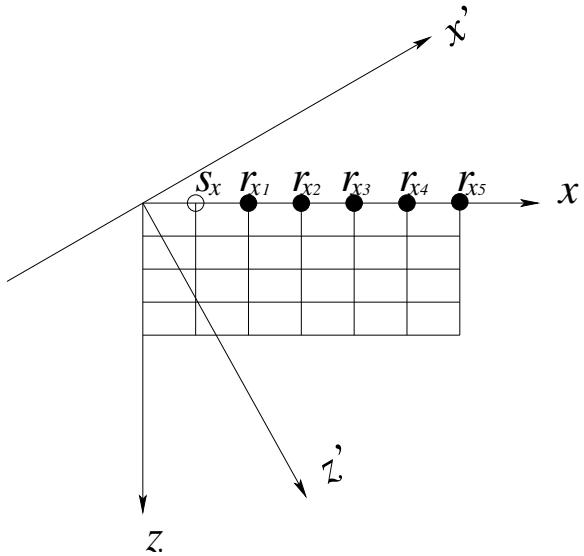
As in the vertical Cartesian coordinates, the up-going and down-going one-way wave equations can be obtained by splitting the acoustic wave equation in the tilted coordinate system (x', z') :

$$\frac{\partial}{\partial z'} S = -\frac{i\omega}{v} \sqrt{1 + \left(\frac{v}{\omega} \frac{\partial}{\partial x'}\right)^2} S, \quad (12)$$

$$\frac{\partial}{\partial z'} R = +\frac{i\omega}{v} \sqrt{1 + \left(\frac{v}{\omega} \frac{\partial}{\partial x'}\right)^2} R. \quad (13)$$

Figure 1: Coordinate system rotation: (x, z) are conventional vertical Cartesian coordinates, (x', z') are tilted coordinates, s_x represents the source location, and $r_{xi}, i = 1, 2, \dots, 5$ represent receiver locations. The source and receivers are on regular grids in vertical Cartesian coordinates.

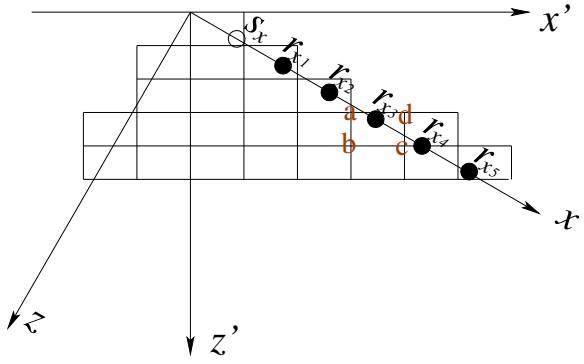
`guojian2/. tiltcoordinate1`



The extrapolation direction of equations 12 and 13 parallels the z' axis, which is θ from the vertical direction. Figure 1 illustrates the coordinate transformation, where (x, z) are vertical Cartesian coordinates and (x', z') are tilted coordinates, s_x represents the source location and $r_{x1}, r_{x2}, \dots, r_{x5}$ represent the corresponding receiver locations. The accuracy of the one-way wavefield extrapolators is still very important for wavefield extrapolation in tilted coordinates. The more accurately we design the wavefield extrapolator, the less sensitive the migration is to the coordinates. With an extrapolator that is not very accurate, such as the 15° equation, waves well handled in one coordinate system are not handled in one that is slightly rotated. In contrast, with an accurate extrapolator, waves can be handled in both tilted coordinate systems. Since one-way wave equations in tilted coordinates are exactly the same as those in vertical Cartesian coordinates, all the methods used to improve the accuracy in the conventional Cartesian coordinates still work in tilted coordinates.

Figure 2: Source and receivers in grids of a tilted coordinate system: (x, z) are conventional vertical Cartesian coordinates, (x', z') are tilted coordinates, s_x represents the source location, and $r_{xi}, i = 1, 2, \dots, 5$ represent receiver locations. Neither source nor receiver locations are on regular grids in the tilted coordinate system. Their wavefield values must be interpolated onto regular grids around the slanted line in tilted coordinates. The wavefield on r_{x3} is interpolated onto the grids a, b, c, and d.

`guojian2/. tiltcoordinate2`



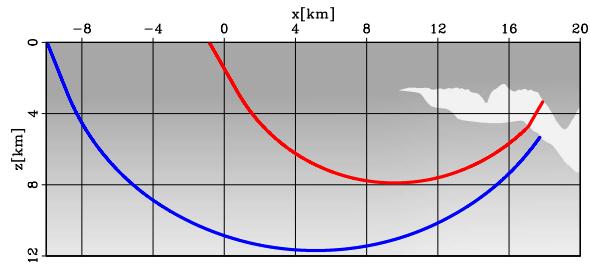
To extrapolate wavefields in a tilted coordinate system, it is necessary to interpolate the surface dataset, velocity model and image between the coordinate systems and migrate the dataset on a slanted line in implementation. In Figure 1, the source and receivers are on regular grids in conventional Cartesian coordinates. Figure 2 shows the source and receivers in meshes in the tilted coordinates (x', z') . Source and receivers are on an inclined line defined by the equation

$$x' \cos \theta - z' \sin \theta = 0. \quad (14)$$

They are not on regular grids in tilted coordinates. To run wavefield extrapolation, the dataset received at the surface has to be interpolated onto the regularized grids around the inclined line in the new coordinate system (x', z') . For instance, the value of the wavefield at r_{xz} has to be interpolated onto the grids a,b,c and d in Figure 2. The velocity must also be interpolated onto the grids in the coordinates (x', z') . In tilted coordinates, the survey is taken on a long, slanted line defined by equation 14. We extrapolate the wavefield with the surface dataset on the slanted line injected at each depth step. We begin the wavefield extrapolation at the point $z' = 0$. For the i -th step extrapolation, when the depth level $z' = i\Delta z$ intersects the slanted line, we add the measured wavefield on the slanted line to the wavefield extrapolated from its previous depth level. After we inject the wavefields on the slanted line, the wavefield extrapolation is the same as the conventional one.

Figure 3 shows a velocity model revised from the Sigsbee 2A model (Sava, 2006). The sediment part of the model is extended vertically and horizontally to receive the overturned waves from the overhanging salt flank at the surface. The rays correspond to the overturned waves from the overhanging flanks on opposite sides of the salt. Figure 4 shows the model and rays in a tilted coordinate system with a tilting angle of 70° . Figures 3 and 4 illustrate that the waves that overturn in vertical Cartesian coordinates do not overturn in a tilted coordinate system with a well-chosen tilting direction.

Figure 3: A velocity model revised from Sigsbee 2A. The sediment parts of the model are extended to allow the overturned waves from the overhanging salt flanks to be received at the surface. The rays represent the overturned waves from the overhanging salt flank.
guojian2/. zigvelwithraycart



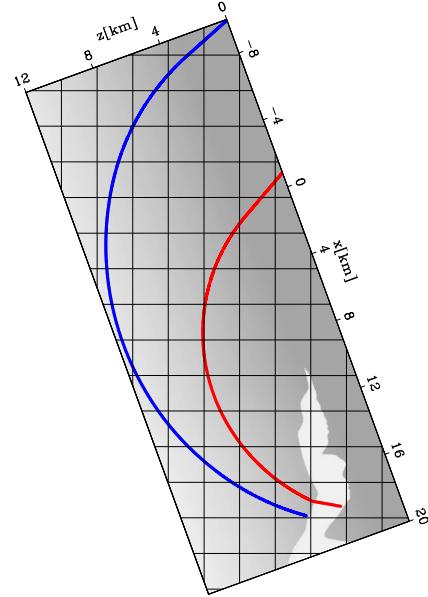
PLANE-WAVE MIGRATION IN TILTED COORDINATES

We introduced the concepts of plane-wave migration and migration in tilted coordinates in previous sections. In this section, we discuss the combination of these two and provide a powerful method for migrating steeply dipping and overturned events. We first discuss why point-source migration in tilted coordinates would not be effective. Then we describe how to design tilted coordinates for each plane-wave source. Finally, we discuss how reciprocity improves plane-wave migration in tilted coordinates.

Waves from a point source propagate radially, and waves start from one spatial location

Figure 4: The velocity model and overturned waves in a tilted coordinate system. The overturned waves in vertical Cartesian coordinates do not overturn in the tilted coordinate system.

guojian2/. zigvelwithraytilt



and travel along all directions. Therefore, it is impossible for a tilted coordinate system to cover all the propagation directions of a point source. Figure 5a illustrates the waves from a point source in tilted coordinates. In Figure 5a the coordinates (x, z) are rotated counter-clockwise, where the high-angle energy can be well modeled on the right side, but the left-side energy (represented by dash-lines) cannot be modeled accurately, even for small-angle energy in vertical Cartesian coordinates. However, the propagation direction of a plane-wave source at different spatial locations is usually similar (Figure 5b). In plane-wave migration, we decompose the wavefield into plane-wave source gathers by slant-stacking, and each plane-wave source gather is characterized by a ray-parameter p_x . Given the velocity at the surface v_{z_0} , the propagation direction of the plane-wave source is defined by the vector (q_x, q_z) , where $q_x = p_x v_{z_0}$ and $q_z = \sqrt{1 - q_x^2}$. Therefore, the ray parameter p_x defines the propagation direction of the plane-wave source at the surface. The take-off angle α of the plane-wave source can be calculated as follows:

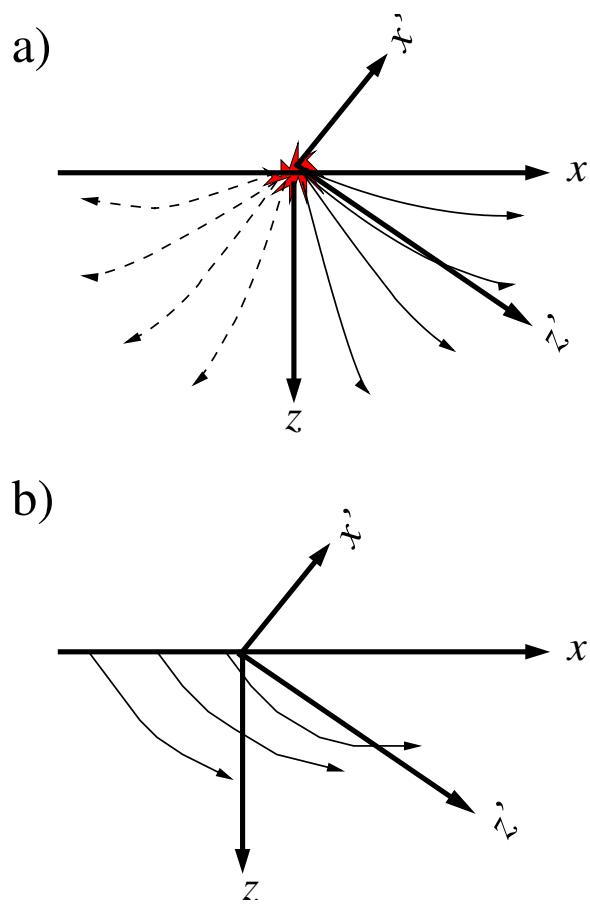
$$\alpha = \arccos(q_x). \quad (15)$$

If we assume the velocity to be invariant at the surface, the propagation direction of the plane-wave source defined in equation 7 at the surface is the same for all spatial points. This is true for a marine dataset, and nearly true for a land dataset, if the velocity does not vary strongly at the surface. Therefore, a tilted coordinate system can cover most of the propagation directions of a plane-wave source from different spatial points, although the propagation direction of the plane-wave may change due to velocity heterogeneities.

Given a plane-wave source with a take-off angle of α , we use tilted coordinates (x', z') , with a tilting angle θ close to its take-off angle α . Usually, velocity increases with depth and the propagation direction of waves becomes increasingly horizontal, so in practice the tilting angle θ is a little larger than the take-off angle. Figure 6 shows three typical plane-wave sources and their tilted coordinate systems. Plane-wave sources with a small take-off angle mainly illuminate reflectors that are almost flat, so we extrapolate wavefields vertically. In

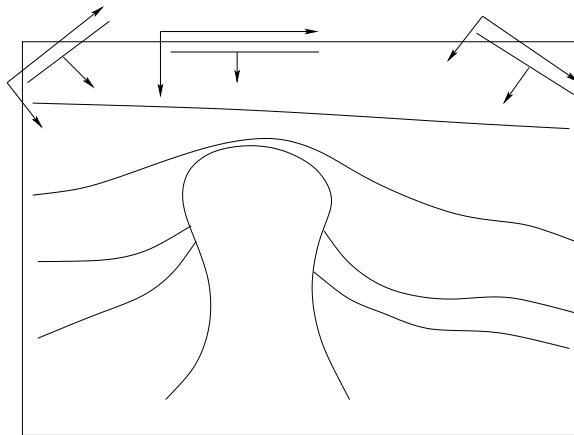
Figure 5: Point source (a) and plane-wave sources (b) in tilted coordinates. Waves from a point source propagate radially, and the waves represented by the dash line rays in panel (a) can not be caught when we rotate the coordinates counter-clockwise. In contrast, the propagation directions of the plane-wave source are similar in different spatial points, so most of them can be extrapolated accurately in a tilted coordinate system.

guojian2/. pointplane



contrast, plane-wave sources with a large take-off angle mainly illuminate steeply dipping reflectors, so we use a tilted coordinate system with a large tilting angle. Usually, these wavefields are difficult to extrapolate accurately by downward continuation migration, but in tilted coordinates their propagation direction is close to the extrapolation direction, so they can be imaged correctly. Waves overturning in vertical Cartesian coordinates do not overturn in a well-chosen tilted coordinate system. Therefore, in plane-wave migration in tilted coordinates, each plane-wave source has its own tilted coordinate system in which the extrapolation direction is close to the propagation direction, and steep reflectors and overturned waves can be imaged correctly.

Figure 6: Plane-wave sources and their tilted coordinates. The tilting direction for the coordinates corresponding to the plane-wave source depends on its take-off angle. There are three typical plane-wave sources, and they have 0, negative and positive ray parameters, respectively. For $p = 0$, we use conventional Cartesian coordinates. For $p > 0$, we rotate coordinates counter-clockwise and for $p < 0$, we rotate coordinates clockwise. guojian2/. planetilt



Usually, in streamer acquisition we only record one-sided offset data at the surface. But we can obtain the data for the other side by reciprocity. Merging the original data and the data obtained by reciprocity, we obtain a dataset that would be recorded if we would have had a split-spread recording geometry. In plane-wave migration for a dataset with a split-spread geometry, the aperture of each plane-wave source is almost the same as one-sided offset dataset, and thus the computation cost is also almost the same. But a split-spread recording geometry improves the plane-wave gathers and the signal-to-noise ratio of the image (Liu et al., 2006).

Reciprocity yields other benefits for plane-wave migration in tilted coordinates. Figure 7 illustrates how reciprocity helps to image steep salt flanks when the source ray does not overturn but the receiver ray does. In Figure 7, the source location is s and receiver location is r . For the original data, we run plane-wave migration for this event using the coordinates (x_s, z_s) , whose tilting angle is determined by the source ray direction at the surface. The source plane wave starts at the surface almost vertically, and the tilting angle of its corresponding coordinates (x_s, z_s) is small. As a consequence, the overturned receiver wave cannot be accurately modeled, and the event cannot be correctly imaged. Reciprocity exchanges the source and receiver locations. For the data obtained by reciprocity, we run plane-wave migration for this event using the coordinates (x_r, z_r) , whose direction is determined by the receiver ray direction at the surface. In the coordinates (x_r, z_r) , both source and receiver waves can be accurately modeled, and the overturned energy can be correctly imaged. When we run plane-wave migration in tilted coordinates for a split-spread dataset, we design the coordinates considering the direction of both the source and receiver waves at the surface.

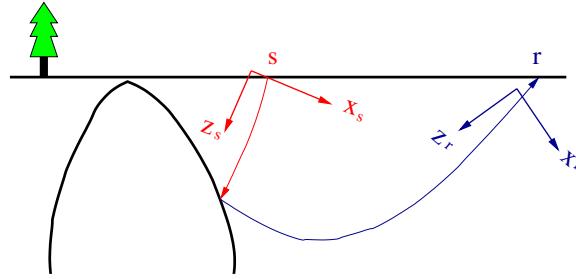


Figure 7: Reciprocity improves plane-wave migration in tilted coordinates. The source location is at s , and the receiver location is at r . For this event, the source ray does not overturn, but the receiver ray does. If we run plane-wave migration in tilted coordinates on the original one-sided offset data, we will use the coordinates (x_s, z_s) , whose direction is determined by the source ray direction at the surface. In this coordinate system, the source waves can be handled but the overturned receiver wave cannot. If we run the same migration on the other side offset data generated by reciprocity, we will use the coordinates (x_r, z_r) for wavefield extrapolation, whose direction is determined by the receiver ray direction at the surface. In this coordinate system, both the source and receiver waves can be handled.

`guojian2/. reciprocity`

NUMERICAL EXAMPLES

An exploding-reflector dataset with overturned waves

Our first example is a synthetic dataset designed to test imaging of overturned waves (Sava, 2006). Figure 3 shows the model with typical overturned rays. The exploding reflector data are modeled from the boundary of the salt and recorded at the surface. The data are modeled using the time-domain two-way wave equation. Figure 8 shows the exploding reflector data received at the surface. The overturned events are recorded from $x = -20$ to 5 km at $t = 10$ to 25s.

To verify the extrapolation of overturned waves in tilted coordinates, we mute the non-overturned events that are received at the surface earlier than 10s. We migrate the dataset using a tilted coordinate system with a tilting angle of 70°, as shown in Figure 4. As demonstrated in the previous section, the waves illuminating the overhanging salt flanks do not overturn in the tilted coordinate system (Figure 4). For comparison, we also migrate the dataset using reverse-time migration. Figure 9 compares the images from these two methods. Figure 9a is the migrated image obtained by plane-wave migration in tilted coordinates, and Figure 9b is the image obtained by reverse-time migration. The image from reverse-time migration has lower frequency; otherwise they are comparable. The comparison shows that most of the overturned energy is imaged by the migration in tilted coordinates, and all the overhanging salt flanks are imaged correctly.

Impulse responses

Our second example is a smooth sediment velocity field embedded with a salt body with steeply dipping flanks. Figure 10 is a comparison of the impulse responses of the two-

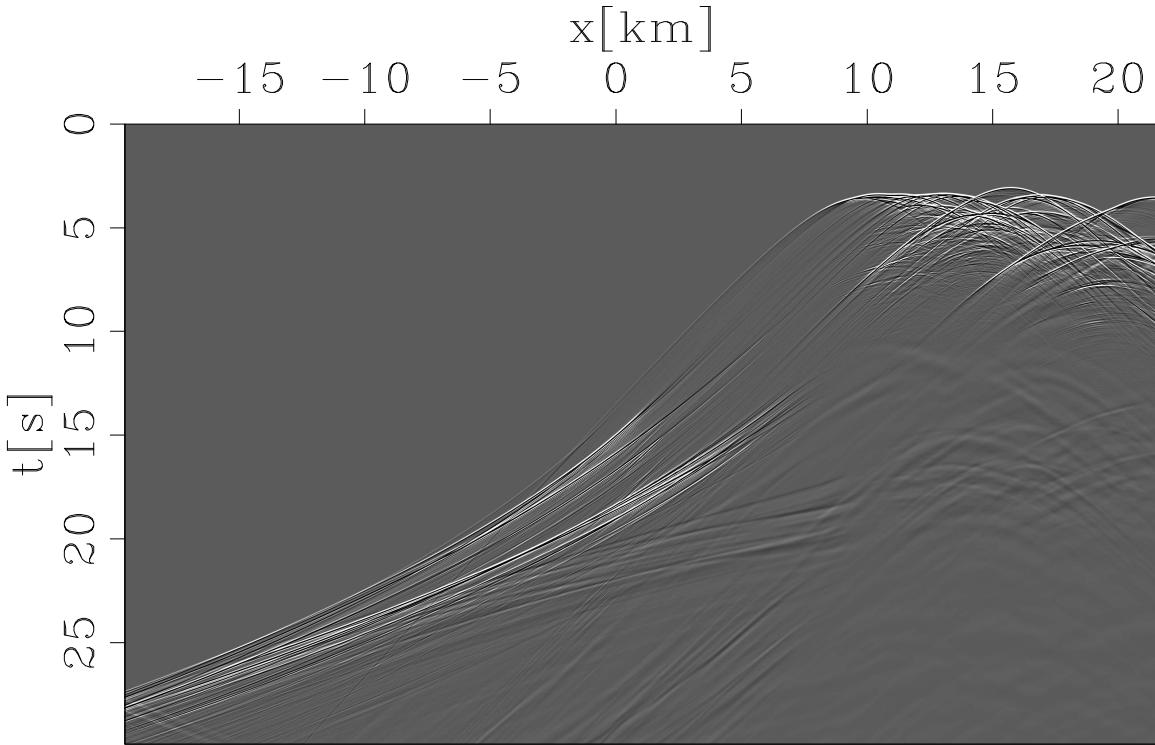


Figure 8: The exploding reflector dataset from the revised Sigsbee 2A model. The overturned energy is recorded from -20 to 5 km at $t = 10$ to 25 s . The energy recorded earlier than 10 s is muted before migration to verify the imaging of overturned waves.
guojian2/. zigdata

way wave equation (Figure 10a), one-way wave-equation downward continuation (Figure 10b) and plane-wave migration in tilted coordinates (Figure 10c). From Figures 10a and b, we observe that the one-way wave equation mimics the two-way wave equation well for energy that propagates with small angles from the vertical direction, but its accuracy drops for energy that propagates almost horizontally. Energy that overturns is lost entirely. Comparing Figure 10c with Figure 10a, we notice that there are no reflections or multiples in Figure 10c. This is not surprising, since the one-way wave-equation extrapolator is applied. But the wave front of the direct arrival matches that of the two-way wave equation very well, even at high angles and with overturned waves, despite being extrapolated with the one-way wave equation. The impulse-response comparison shows the potential to image the steeply dipping reflectors and overturned waves by plane-wave migration in tilted coordinates.

BP 2004 velocity benchmark dataset

The BP 2004 velocity benchmark dataset is designed to test velocity estimation. Figure 1 shows the velocity model of the dataset. One of the challenges for velocity analysis of this dataset is the delineation of the two salt bodies. The salt body on the left, modeled after a salt body in the Gulf of Mexico, is a complex, multi-valued salt body with a greatly rugose top. Some parts of its top, flanks and the sediment intrusion inside the salt are steeply

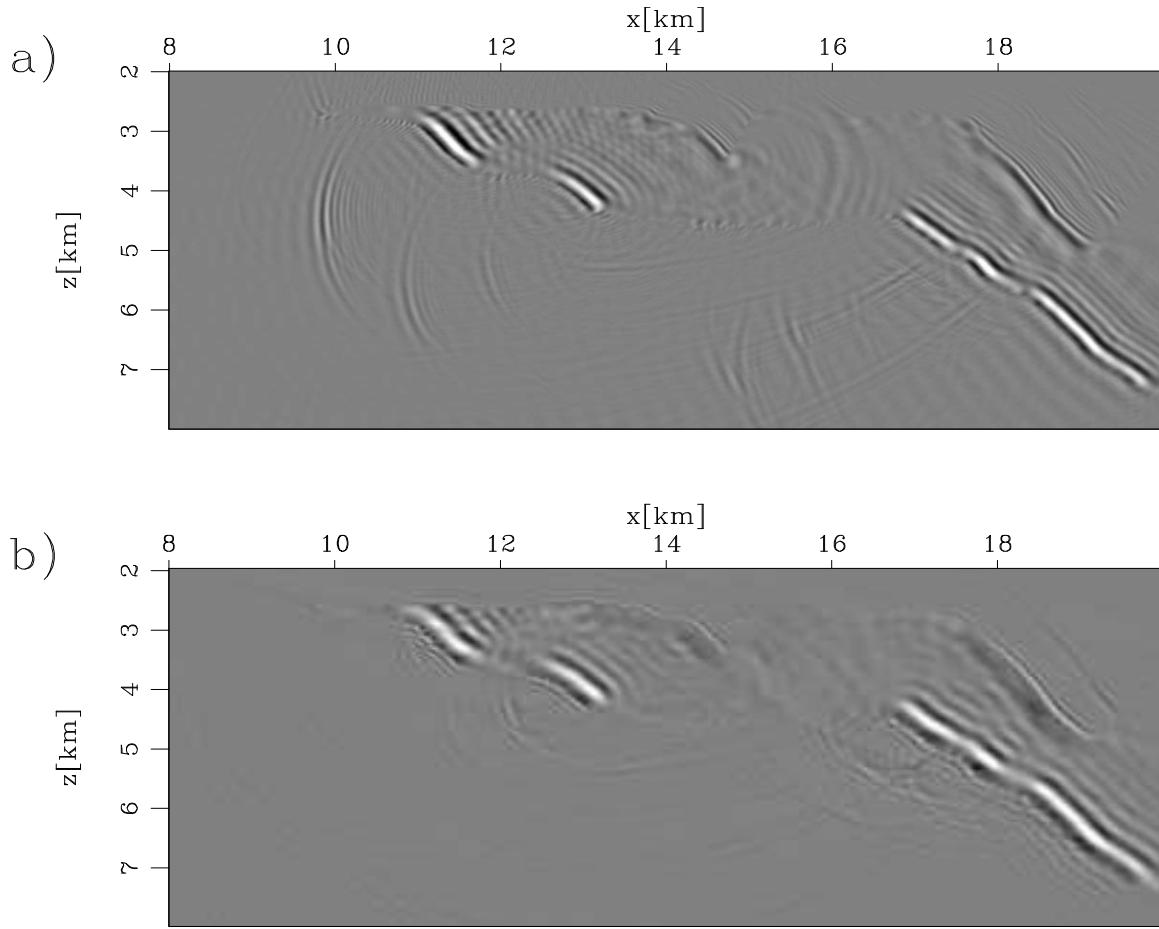


Figure 9: Migrated image of the overturned waves: migration in tilted coordinates (a) and reverse-time migration (b). `guojian2/. zigimage`

dipping. It is difficult for downward continuation migration to image these features. The salt body on the right, modeled after a salt body in the western Africa, is deeply rooted, and its roots are very steep. Overturned and prismatic waves play a key role in imaging the two roots of the right salt body. Downward continuation loses the overturned energy and cannot connect these two roots. Even with the true velocity, it is challenging to image these complex salt bodies.

We run both plane-wave migrations in tilted coordinates and downward continuation migration for comparison. Two hundred plane-wave sources are generated in total, and the take-off angles at the surface range from -45° to 45° . No attempt is made to attenuate multiples, thus the images are contaminated by the multiples. The 80° finite-difference one-way extrapolator (Lee and Suh, 1985) is applied for both migrations.

Figure 12 shows the velocity model of the left salt body. Figure 13 and Figure 14 compare the images from the two migrations. Notice that remarks A, B, C, D, E, F, G and H in Figures 12, 13 and 14 are in exactly the same locations. Figures 13 and 14 are the images obtained by plane-wave migration in tilted coordinates and downward continuation migration, respectively. In both figures, the bottom of the big salt canyon is well imaged.

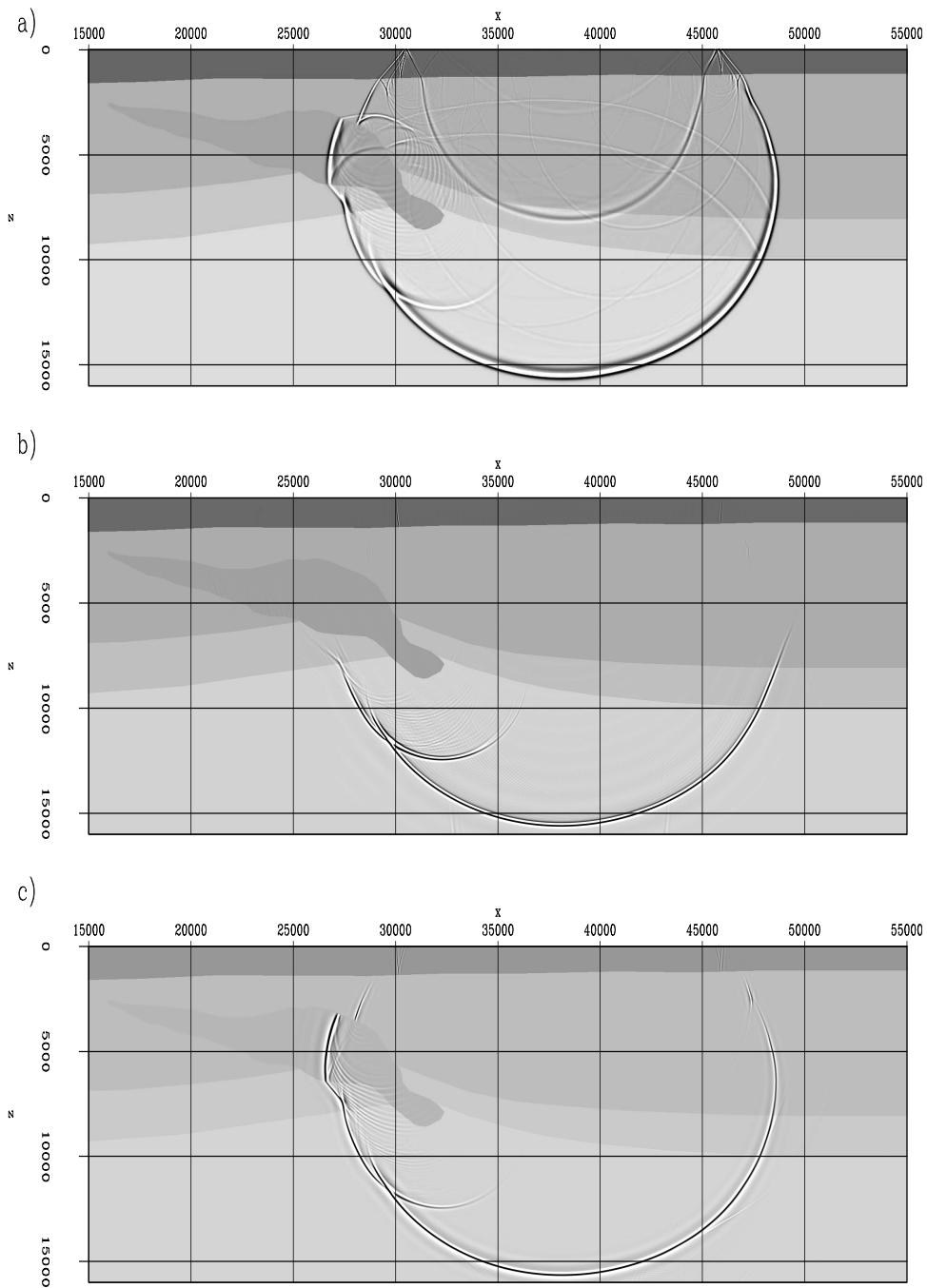


Figure 10: Impulse response comparison among (a) two-way wave equation, (b) one-way wave-equation downward continuation and (c) plane-wave migration in tilted coordinates.

`guojian2/. impulse`

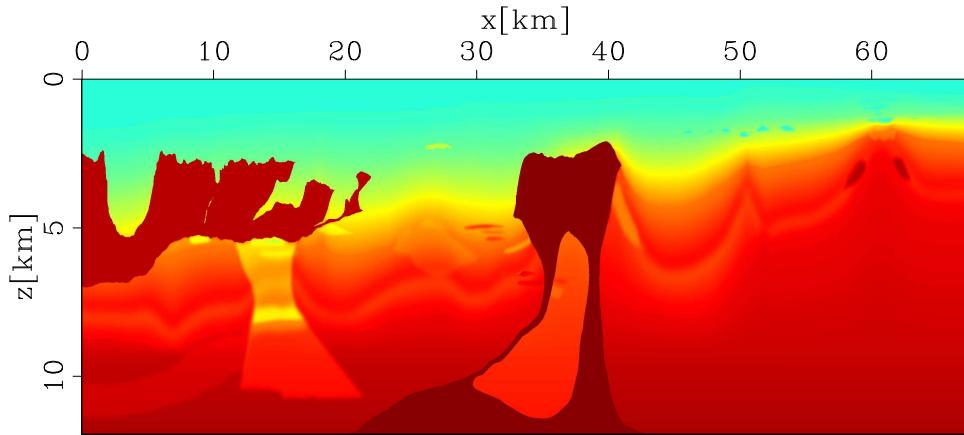


Figure 11: The velocity model of the BP velocity benchmark. `guojian2/. bpvel`

But the steep flanks of the canyon at A and B, which are absent in Figure 14, are correctly imaged in Figure 13. This is also true for the small salt canyon at C. Although the salt canyon flank at D is imaged by downward continuation migration in Figure 14, it is not positioned correctly due to the limited accuracy of the operator compared to the model (Figure 12). The rugose top of the salt in Figure 13 is more continuous than that in Figure 14. The steep salt flanks in the multi-valued part at E, F and G and the sediment intrusion below the small salt canyon at H are greatly improved in Figure 13 by plane-wave migration in tilted coordinates, because they are illuminated by overturned or high-angle energy, which cannot be handled by downward continuation migration.

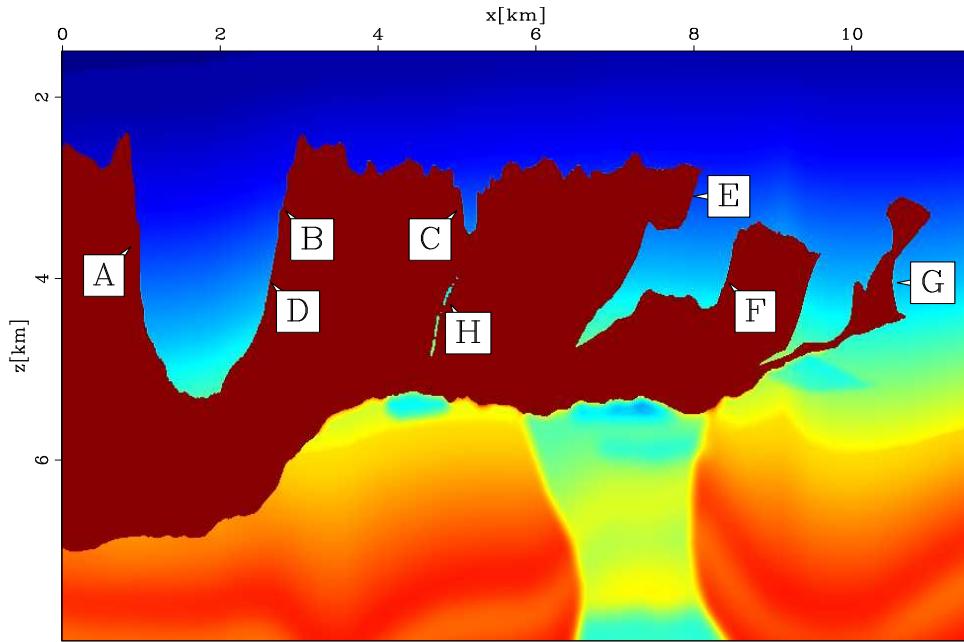


Figure 12: The velocity model of the left salt body. `guojian2/. bpleftvel`

Figure 15 shows the velocity model of salt body on the right. Figure 16 and Figure 17

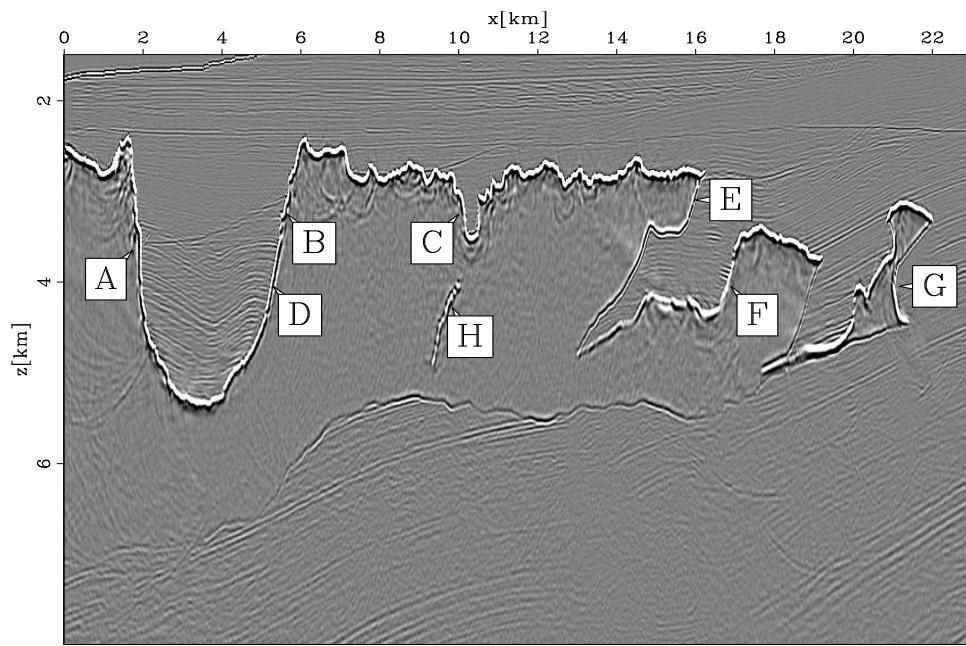


Figure 13: The images of the left salt body obtained by plane-wave migration in tilted coordinates. `guojian2/. bpleftsalttilt`

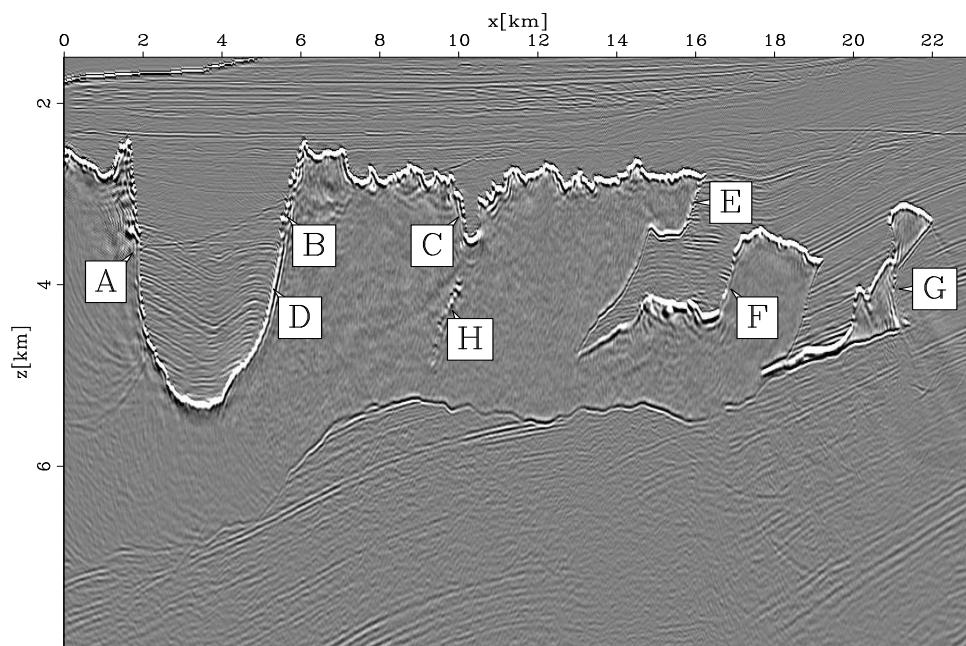


Figure 14: The images of the left salt body obtained by downward continuation migration. `guojian2/. bpleftsaltnotilt`

compare the images from the two migrations. Notice that remarks A, B, C, D and E in Figures 15, 16 and 17 are in exactly the same locations. Figure 16 is obtained by plane-wave migration in tilted coordinates, and Figure 17 is obtained by downward continuation migration. The top of the salt and sediments inside the salt are well imaged in both figures. But the salt flanks at A, B and D that are illuminated by the overturned or high-angle energy in vertical Cartesian coordinates are absent in Figure 17. In contrast, this overturned energy is handled by plane-wave migration in tilted coordinates, producing a good image of the flanks of the salt roots. In Figure 17, we can see the steep flank at C, but it is not correctly positioned compared to Figure 16 because of the limited accuracy of the wavefield extrapolator. Note that the salt flank at E is absent in both images. This flank is illuminated mainly by prismatic waves which bounce off the salt root below E. The propagation direction of the prismatic waves varies greatly before and after the bounce at the salt boundary, and it is difficult to model them accurately in one coordinate system.

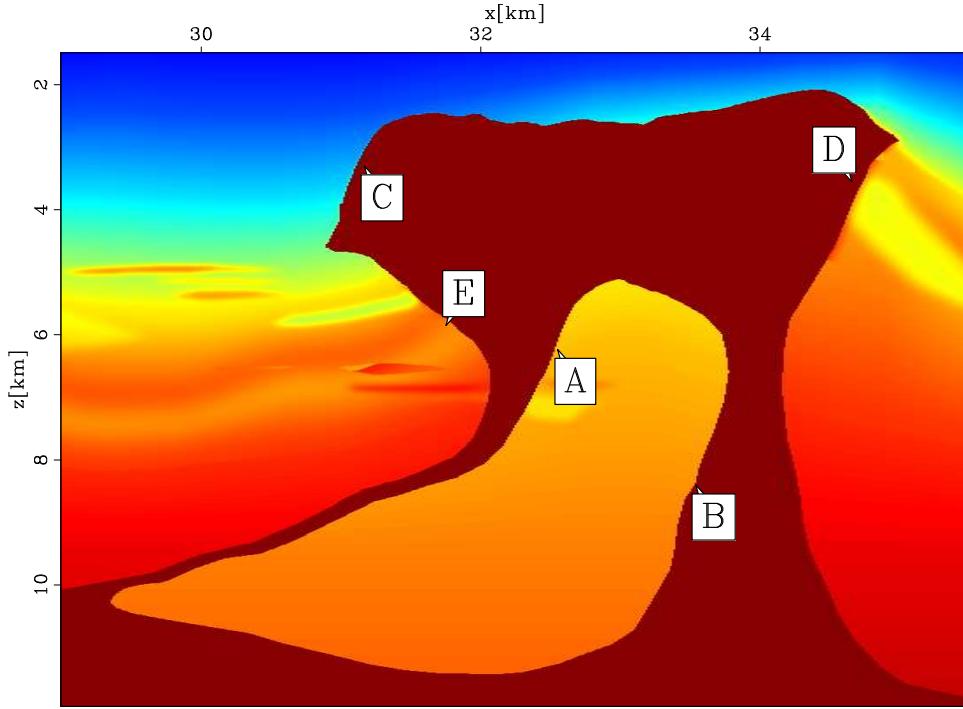


Figure 15: The velocity model of the right salt body. guojian2/. bprightvel

Figures 13, 14, 16 and 17 show that plane-wave migration in tilted coordinates can handle overturned and high-angle energy and delineate complex salt bodies much better than downward continuation migration.

CONCLUSION

Plane-wave migration in tilted coordinates makes the extrapolation direction close to the actual propagation direction in the subsurface by assigning a well-chosen tilted coordinate system for each plane-wave source. One-way wave equations in tilted coordinates are exactly the same as those in vertical Cartesian coordinates, therefore we can still use the accurate one-way extrapolator methods developed for vertical Cartesian coordinates in last

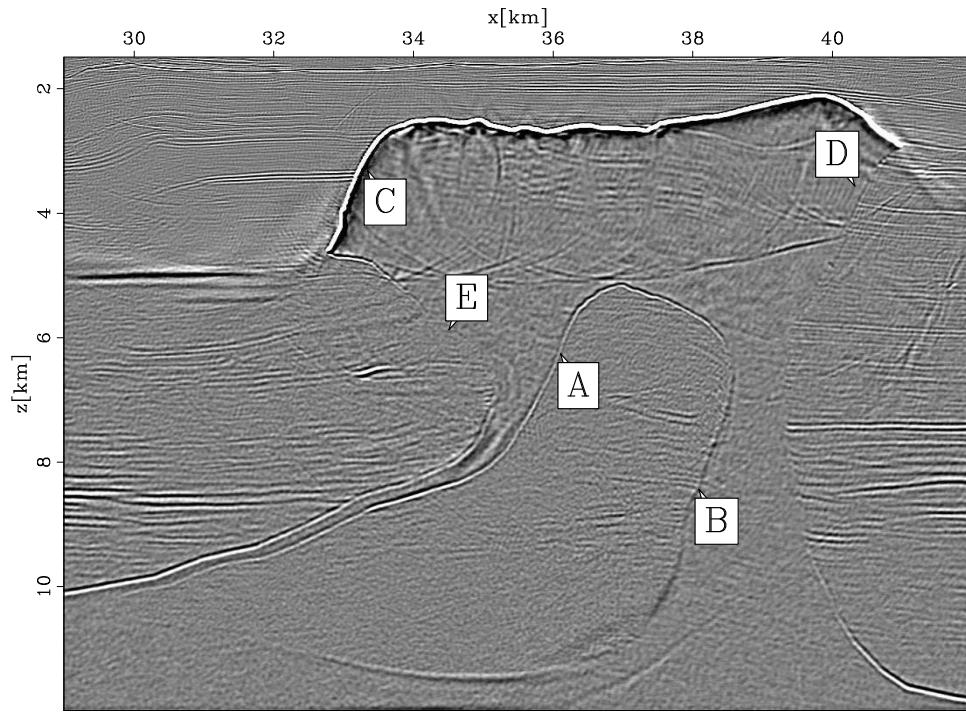


Figure 16: The images of the right salt body obtained by plane-wave migration in tilted coordinates. `guojian2/. bprightsaltilt`

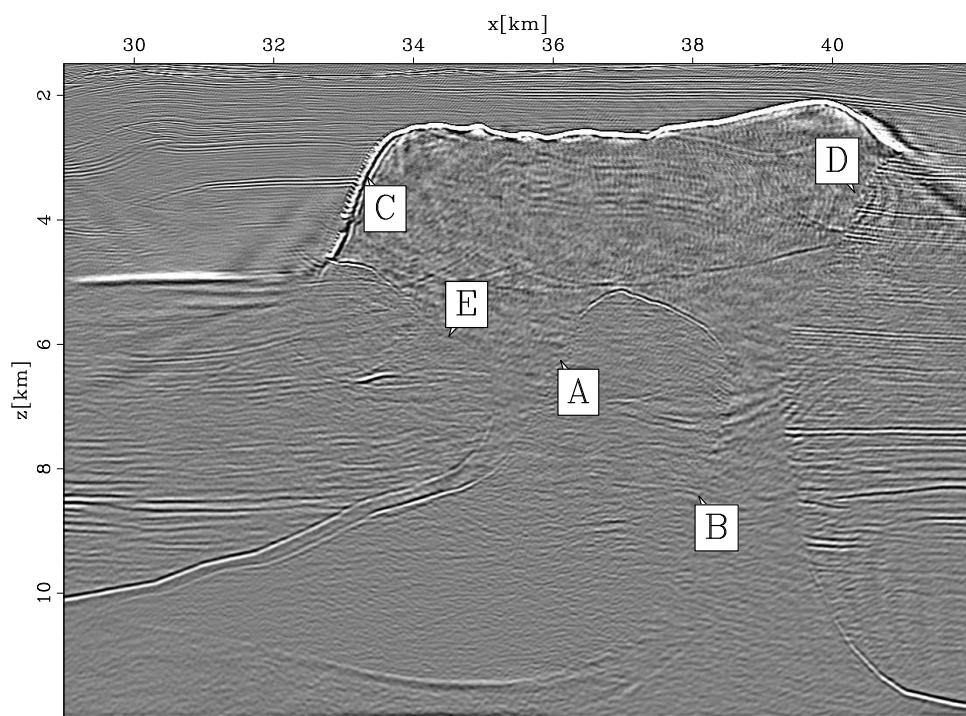


Figure 17: The images of the right salt body obtained by downward continuation migration. `guojian2/. bprightsaltnotilt`

two decades to reduce the sensitivity to the coordinates. Plane-wave migration in tilted coordinates is much cheaper than reverse-time migration, but it can handle waves that illuminate steeply dipping reflectors and overhanging flanks, such as high-angle energy and overturned waves, which are challenging to image with conventional one-way downward continuation migration. Examples show that plane-wave migration in tilted coordinates is a good tool for delineation of complex salt bodies.

ACKNOWLEDGMENTS

We thank John Etgen of BP Upstream Technology Group for helpful discussion, Amerada Hess and BP for the synthetic datasets, and Paul Sava for the exploding reflector dataset.

REFERENCES

- Baysal, E., D. D. Kosloff, and J. W. C. Sherwood, 1983, Reverse time migration: *Geophysics*, **48**, 1514–1524.
- Biondi, B., 2002, Stable wide-angle Fourier finite-difference downward extrapolation of 3-D wavefields: *Geophysics*, **67**, 872–882.
- Biondi, B. and G. Shan, 2002, Prestack imaging of overturned reflections by reverse time migration, *in* Expanded Abstracts, 1284–1287, Soc. of Expl. Geophys., 72nd Ann. Internat. Mtg.
- Claerbout, J. F., 1971, Toward a unified theory of reflector mapping: *Geophysics*, **36**, 467–481.
- , 1985, Imaging the Earth’s interior: Blackwell Scientific Publications.
- de Hoop, M. V., 1996, Generalization of the Bremmer coupling series: *J. Math. Phys.*, 3246–3282.
- Duquet, B., P. Lailly, and A. Ehinger, 2001, 3D plane wave migration of streamer data, *in* 71st Ann. Internat. Mtg, 1033–1036, Soc. of Expl. Geophys.
- Etgen, J., 2002, Waves, beams and dimensions: an illuminating if incoherent view of the future of migration: Presented at the 72nd Ann. Internat. Mtg, Soc. of Expl. Geophys.
- , 2005, How many angles do we really need for delayed-shot migration?, *in* Expanded Abstracts, 1985–1988, Soc. of Expl. Geophys., 74th Ann. Internat. Mtg.
- Gazdag, J., 1978, Wave equation migration with the phase-shift method: *Geophysics*, **43**, 1342–1351.
- Higginbotham, J. H., Y. Shin, and D. V. Sukup, 1985, Directional depth migration (short note): *Geophysics*, **50**, 1784–1796.
- Huang, L. Y. and R. S. Wu, 1996, Prestack depth migration with acoustic screen propagators, *in* 66th Ann. Internat. Mtg, 415–418, Soc. of Expl. Geophys.
- 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 3-D 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., R. Stolt, D. Hanson, and R. Day, 2002, Plane wave source composition: An accurate

- phase encoding scheme for prestack migration, *in* 72nd Ann. Internat. Mtg, 1156–1159, Soc. of Expl. Geophys.
- Nichols, D., 1994, Imaging in complex structures using band-limited green's function, *in* Ph.D. thesis, Stanford University.
- Rietveld, W. E. A., 1995, Controlled illumination of prestack seismic migration, *in* Ph.D. thesis, Delft University of Technology.
- Ristow, D. and T. Ruhl, 1994, Fourier finite-difference migration: *Geophysics*, **59**, 1882–1893.
- Sava, P., 2006, Imaging overturning reflections by Riemannian wavefield extrapolation: *Journal of Seismic Exploration*, **15**, 209–223.
- Sava, P. and S. Fomel, 2005, Riemannian wavefield extrapolation: *Geophysics*, **70**, T45–T56.
- Shragge, J., 2006, Non-orthogonal Riemannian wavefield extrapolation, *in* Expanded Abstracts, 2236–2240, Soc. of Expl. Geophys., 75th Ann. Internat. Mtg.
- Whitmore, N. D., 1983, Iterative depth migration by backward time propagation, *in* 53rd Ann. Internat. Mtg, Session:S10.1, Soc. of Expl. Geophys.
- , 1995, An imaging hierarchy for common angle plane wave seismograms, *in* Ph.D. thesis, University of Tulsa.
- Zhang, G., 1993, System of coupled equations for up-going and down-going waves: *Acta Mathematicae Sinica*, **16**, 251–263.
- Zhang, J. and G. A. McMechan, 1997, Turning wave migration by horizontal extrapolation: *Geophysics*, **62**, 291–297.
- Zhang, Y., J. Sun, C. Notfors, S. Gray, L. Chernis, and J. Young, 2005, Delayed-shot 3D depth migration: *Geophysics*, **70**, E21–E28.

Angle domain common image gathers for steep reflectors

Guojian Shan and Biondo Biondi¹

ABSTRACT

Downward continuation migration cannot provide reliable angle domain common image gathers (CIGs) for steeply dipping reflectors, because it cannot handle most waves that illuminate steep reflectors. Also there is a severe stretch in conventional horizontal subsurface offset at steep reflectors. Both reverse-time migration and plane-wave migration in tilted coordinates solve these two problems and provide robust angle domain CIGs for steeply dipping reflectors. A test on the BP velocity benchmark dataset shows that both migration methods generate robust angle domain CIGs that are comparable. When the migration velocity is not correct, the angle domain CIGs from both migration methods show useful moveout information for velocity estimation.

INTRODUCTION

Velocity estimation plays a key role in seismic imaging. A typical migration velocity analysis method includes three steps: (1) migrations using the background velocity are run to obtain angle domain CIGs; (2) curvatures are estimated from angle domain CIGs by residual moveout analysis; (3) curvature information is inverted to velocity update by back projection.

Assumptions about the subsurface are made during most velocity estimation methods, such as a horizontal stratified earth for NMO or flat reflectors by Toldi (1985). Even recently, for a sophisticated tomography, only reflectors that are not very steep are chosen for velocity estimation and are assumed to be flat in residual moveout analysis (Clapp, 2000).

However, CIGs of steeply dipping reflectors are important in velocity analysis. The angular coverage of the rays illuminating near-flat reflectors is very limited. Most rays travel in a direction that is less than 30° from the vertical direction. Therefore, seismic reflection tomography is a limited angle tomography, an ill-posed problem (Tam and Perez-Mendez, 1981). Since most rays are almost vertical, the vertical resolution in seismic reflection tomography is very limited (Clapp, 2000). In contrast, most waves illuminating steeply dipping reflectors have a part of wave-path that is almost horizontal. Therefore using angle domain CIGs of steep reflectors improves the angle coverage of rays in tomography. As a consequence, this reduces the poor condition and improves the stability of the problem. It also leads to less artifacts caused by low angular coverage and better vertical resolution of the resulting velocity.

Angle domain CIGs of steeply dipping reflectors are also useful for anisotropy parameter estimation. VSP and check shots are usually used to improve the angular coverage

¹e-mail: valencia@sep.stanford.edu

in anisotropy parameter inversion (Bear et al., 2005) in addition to the reflectors picked in conventional reflection tomography. Since angle domain CIGs of steep reflectors also broaden the angular coverage, they help to constrain anisotropy parameter estimation. It is well known that the anisotropy parameter δ is mainly constrained by waves traveling close to the vertical direction but the anisotropy parameter η is mainly constrained by waves traveling close to the horizontal direction. Therefore, CIGs of reflectors that are almost flat is useful for estimating the parameter δ but the estimation for the parameter η estimation needs CIGs of steeply dipping reflectors.

Downward-continuation migration is routinely applied in the industry. However, it is difficult to obtain reliable angle domain CIGs of steeply dipping reflectors by conventional downward continuation migration (Biondi and Shan, 2002; Biondi and Symes, 2004). Downward continuation migration is based on the one-way wave equation, so it can propagate waves traveling almost vertically well but it cannot propagate waves traveling almost horizontally accurately. But waves illuminating steeply dipping reflectors travel almost horizontally or even overturn before or after they bounce. Reverse-time migration can image steeply dipping reflectors and provide robust angle domain CIGs by using the vertical subsurface offset in addition to the horizontal subsurface offset (Biondi and Shan, 2002).

Shan and Biondi (2004) have demonstrated that plane-wave migration in tilted coordinates is an effective tool to image steeply dipping reflectors. In this paper, we discuss how to produce reliable angle domain CIGs using plane-wave migration in tilted coordinates. We use the BP velocity benchmark dataset to compare angle domain CIGs from reverse-time migration and plane-wave migration in tilted coordinates. Before we discuss plane-wave migration in tilted coordinates, we briefly review how to generate angle domain CIGs by downward continuation migration and reverse-time migration.

ANGLE DOMAIN CIGS BY DOWNWARD CONTINUATION MIGRATION

Both shot-profile and source-receiver migrations can generate subsurface offset domain CIGs that are equivalent to each other (Shan and Zhang, 2003). In shot-profile migration, the source wavefield $S(s_x, x, z, \omega)$ and receiver wavefield $R(s_x, x, z, \omega)$ are extrapolated into all depths, where s_x is the source location, x is the horizontal location, z is the vertical location, and ω is the angular frequency. Offset domain CIGs $I(x, z, h_x)$ are formed by cross-correlating the source and receiver wavefields with a horizontal shift h_x as follows (Rickett and Sava, 2002):

$$I(x, z, h_x) = \int \int S^*(s_x, x - h_x, z, \omega) R(s_x, x + h_x, z, \omega) d\omega ds_x \quad (1)$$

where the horizontal shift h_x is called horizontal subsurface offset, and S^* is the conjugate complex of the source wavefield S . And $I(x, z, h_x = 0)$ is the conventional image. Offset domain CIGs $I(x, z, h_x)$ are transformed to angle domain CIGs $I(x, z, \gamma)$ by applying local slant-stacking in the space domain or radial-trace transform in the Fourier domain using the relationship as follows (Sava and Fomel, 2003):

$$\tan \gamma = -\frac{k_{h_x}}{k_z}, \quad (2)$$

where k_{h_x} and k_z are wavenumbers corresponding to h_x and z , respectively.

For near-flat reflectors, angle domain CIGs obtained by downward continuation migration using equations 1 and 2 are reliable. However, there are two issues in downward continuation migration that impose difficulty in obtaining reliable CIGs of steeply dipping reflectors. First, steeply dipping reflectors are mainly illuminated by high angle and overturned waves, but these waves cannot be modeled accurately by downward continuation migration. Second, because of the stretch of the horizontal subsurface offset at steeply dipping reflectors, we cannot obtain reliable angle domain CIGs from horizontal offset domain CIGs. Given an opening angle, the steeper the reflector is, the larger the horizontal subsurface offset is needed to get a reliable angle domain CIG. However, the length of the subsurface offset is limited in shot-profile migration to save the cost. Therefore, we cannot get useful angle domain CIGs of steep reflectors from horizontal offset domain CIGs.

In the next two sections, we discuss how the two issues in downward continuation migration are solved by reverse-time migration and plane-wave migration in tilted coordinates.

ANGLE DOMAIN CIGS BY REVERSE-TIME MIGRATION

Reverse-time migration, based on the two-way wave equation, handles high-angle energy and overturned waves naturally. In downward continuation migration, source and receiver wavefields are extrapolated along the z -axis and the subsurface offset direction (the horizontal direction) is normal to the extrapolation direction (the vertical direction). In contrast, in reverse-time migration the source wavefield $S = S(s_x, x, z, t)$ and the receiver wavefield $R = R(s_x, x, z, t)$ are extrapolated along the time axis, where s_x is the source location, x is the horizontal location, z is the vertical location and t is the travel-time. There is no functional difference between the x -axis and z -axis. Therefore, we can obtain general-direction subsurface offset CIGs in reverse-time migration and conventional horizontal offset and vertical offset are only two special cases (Biondi and Shan, 2002). As with downward continuation migration, in reverse-time migration horizontal offset domain CIGs are formed by cross-correlating source and receiver wavefields with a horizontal shift h_x as follows:

$$I_x(x, z, h_z) = \int \int S(s_x, x - h_x, z, t)R(s_x, x + h_x, z, t)dt ds_x, \quad (3)$$

where the shift h_x is called horizontal subsurface offset. Similarly, vertical offset domain CIGs are formed by cross-correlating source and receiver wavefields with a vertical shift h_z as follows:

$$I_z(x, z, h_z) = \int \int S(s_x, x, z - h_z, t)R(s_x, x, z + h_z, t)dt ds_x, \quad (4)$$

where the shift h_z is called vertical subsurface offset.

As with downward continuation migration, we can apply equation 2 to transform the horizontal offset domain CIGs $I_x(x, z, h_x)$ to angle domain CIGs $I_x(x, z, \gamma)$. Similarly, we can also transform the vertical offset domain CIGs $I_z(x, z, h_z)$ to angle domain CIGs $I_z(x, z, \gamma)$ as follows:

$$\tan \gamma = -\frac{k_{h_z}}{k_x}, \quad (5)$$

where k_{h_z} and k_x are wavenumbers corresponding to h_z and x , respectively. Horizontal CIGs work well for flat reflectors but they are not reliable for steep reflectors, while vertical

CIGs are good for steep reflectors. Both vertical and horizontal CIGs are not robust for an area with complex geology, where reflectors have a full range of dips. For a image point, the subsurface offset that parallels the dip direction of the reflector is called geologic offset. CIGs would be robust if we used geologic offset for each image point. However, it is too expensive to generate geologic offset CIGs directly. Biondi and Symes (2004) demonstrate that the geologic offset h_0 , horizontal offset h_x , and vertical offset h_z can be linked by the following relationships:

$$h_x = \frac{h_0}{\cos(\alpha)}, \quad (6)$$

$$h_z = \frac{h_0}{\sin(\alpha)}, \quad (7)$$

where α is the dip angle of the reflector. The relationships (equations 6 and 7) also show why horizontal CIGs fail at steeply dipping reflectors. Large horizontal subsurface offset is needed to get reliable angle domain CIGs for a steep reflector. For the extreme case that the reflector is vertical, from equation 6 we need infinite horizontal subsurface offset.

Although neither vertical nor horizontal CIGs are robust, robust angle domain CIGs can be constructed by merging them as follows (Biondi and Symes, 2004):

$$I(x, z, \gamma) = \cos^2 \alpha(x, z) I_x(x, z, \gamma) + \sin^2 \alpha(x, z) I_z(x, z, \gamma), \quad (8)$$

where $\alpha(x, z)$ is the dip angle at the location (x, z) . Equation 8 is performed in the Fourier domain (k_x, k_z) , in which the dip angle of the reflector can be calculated accurately.

ANGLE GATHERS BY PLANE-WAVE MIGRATION IN TILTED COORDINATES

Reverse-time migration solves the two issues in downward continuation migration in generating CIGs for steep reflectors, but it is well known that it is expensive to apply reverse-time routinely. Plane-wave migration in tilted coordinates has been demonstrated useful imaging technology for steep reflectors (Shan and Biondi, 2004; Shan et al., 2007). In plane-wave migration in tilted coordinates, the propagation direction of the waves illuminating steeply dipping reflectors is usually close to the extrapolation direction and thus they can be imaged correctly. In this section, we discuss how to generate angle domain CIGs by plane-wave migration in tilted coordinates and show that it can also produce reliable CIGs for steep reflectors. We start with CIGs in the conventional plane-wave migration.

As with shot-profile migration, offset domain CIGs in plane-wave migration are formed as follows:

$$I(x, z, h_x) = \int \int \omega S^*(p_x, x - h_x, z, \omega) R(p_x, x + h_x, z, \omega) d\omega dp_x, \quad (9)$$

where h_x is the horizontal subsurface offset, $S(p_x, x, z, \omega)$ and $R(p_x, x, z, \omega)$ are the source and receiver wavefields corresponding to the ray parameter p_x , respectively. Notice that the imaging condition in equation 9 is the cross-correlation between the source and receiver wavefields weighted with the angular frequency ω , which is also called ρ -filter in Radon transform literature. As with the conventional zero-subsurface-offset image, offset domain

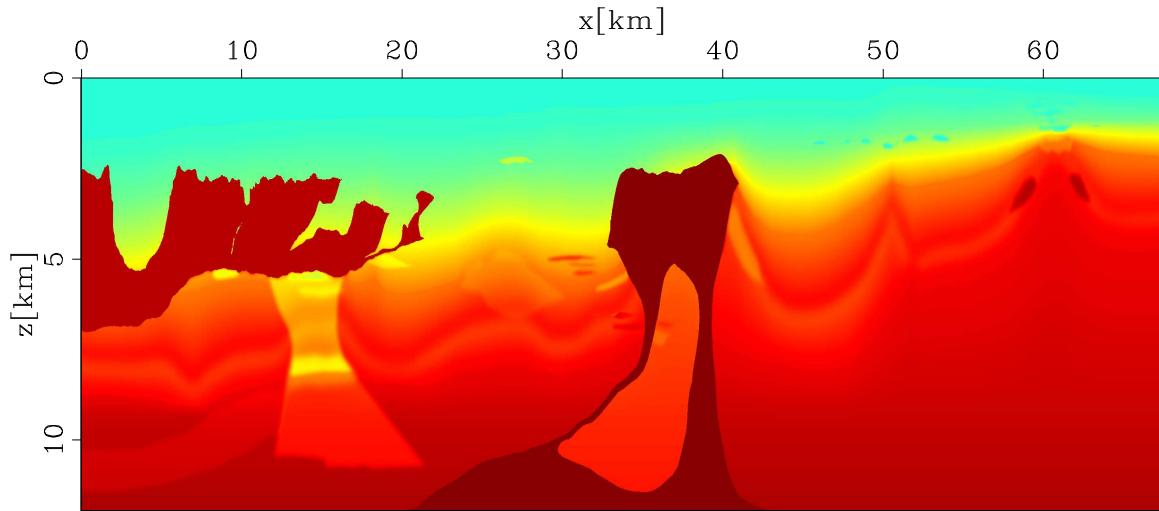


Figure 1: Velocity model of the BP velocity Benchmark. `guojian1/. bpvel`

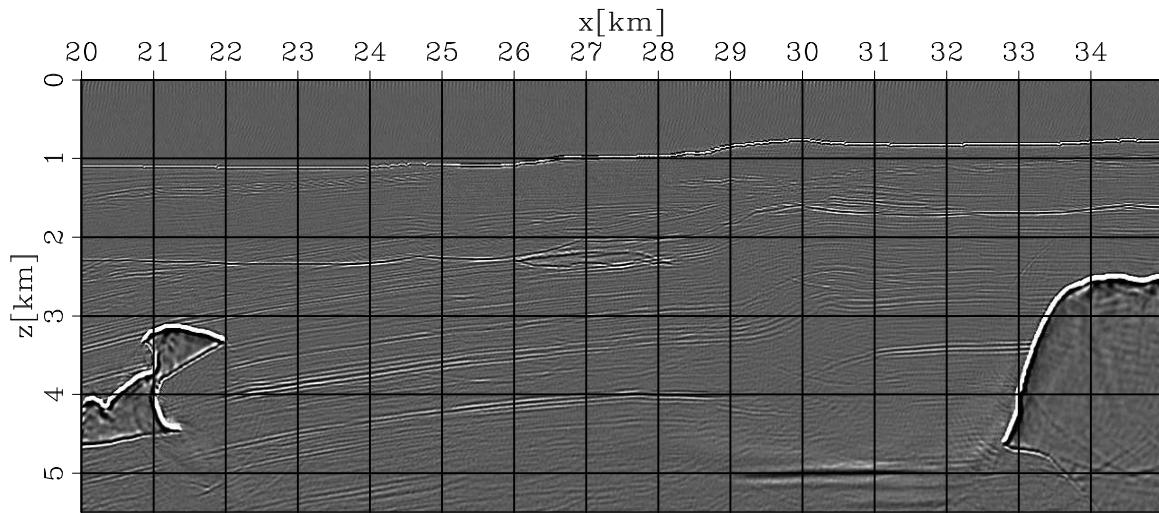


Figure 2: Image obtained by plane-wave migration in tilted coordinates. Both steep salt flank and near-flat sediments are present in this area. `guojian1/. imagetilt`

CIGs defined in equation 9 are equivalent to those obtained by shot-profile migration. Offset domain CIGs are transformed to angle domain CIGs by local slant-stacking (equation 2).

Given a plane-wave source corresponding to the ray parameter p_x , we use the tilted coordinates (x', z') with a tilting angle θ . The subsurface offset domain CIGs for this plane-wave source are formed by:

$$I_{p_x}(x', z', h_{x'}) = \int \omega S^*(p_x, x' - h_{x'}, z', \omega) R(p_x, x' + h_{x'}, z', \omega) d\omega. \quad (10)$$

where the subsurface offset $h_{x'}$ parallels the x' axis. In plane-wave migration in tilted coordinates, the subsurface offset direction is not necessary the geologic dip direction, but is usually closer to the dip direction for steeply dipping reflectors, than the conventional horizontal subsurface offset. As for the transformation in the conventional plane-wave migration, we can transform offset domain CIGs $I_{p_x}(x', z', h_{x'})$ of plane-wave source corresponding to p_x to angle domain CIGs $I_{p_x}(x', z', \gamma)$ in tilted coordinates by applying

$$\tan \gamma = -\frac{k_{h_{x'}}}{k_{z'}}, \quad (11)$$

where $k_{h_{x'}}$ and $k_{z'}$ are wavenumbers corresponding to $h_{x'}$ and z' , respectively. For each angle γ , we rotate the image $I_{p_x}(x', z', \gamma)$ back to vertical Cartesian coordinates. The angle domain CIGs of all possible plane-wave sources are then stacked in vertical Cartesian coordinates.

We can also transform the subsurface offset CIGs obtained by plane-wave migration in tilted coordinates into horizontal offset and vertical offset CIGs, and merge them using equation 8 after transforming them into angle domain CIGs, similarly to reverse-time migration. Equations 6 and 7 are the relationships linking the geologic offset h_0 , horizontal offset h_x and vertical offset h_z . The horizontal and vertical offsets are two special cases and the relationship can be generalized to a general-direction offset. If the angle between the general-direction offset \bar{h} and geologic offset h_0 is β , the relationship between them is

$$\bar{h} = \frac{h_0}{\cos \beta}. \quad (12)$$

The angle β in equation 12 for $\bar{h} = h_x$ is α and for $\bar{h} = h_z$ is $90^\circ - \alpha$. From equation 12, the geologic offset h_0 is the optimal offset to generate angle domain CIGs and the further the offset direction is from the dip direction, the larger the subsurface offset we need given the same opening angle. For the tilted coordinate system (x', z') , the angle between the subsurface offset and geologic offset is $\theta - \alpha$. Therefore, the subsurface offset $h_{x'}$ in tilted coordinates and the geologic offset h_0 can be linked by the following relationship:

$$h_{x'} = \frac{h_0}{\cos(\theta - \alpha)}. \quad (13)$$

From equations 13, 6 and 7, the subsurface offset in tilted coordinates $h_{x'}$, vertical offset h_z and horizontal offset h_x are linked by the following relationship:

$$h_x = \frac{h_{x'} \cos(\theta - \alpha)}{\cos \alpha}, \quad (14)$$

$$h_z = \frac{h_{x'} \cos(\theta - \alpha)}{\sin \alpha}. \quad (15)$$

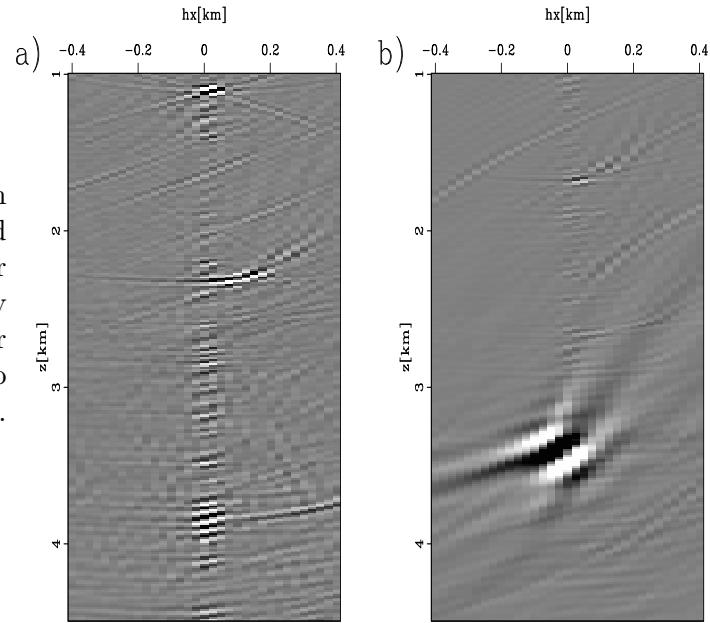


Figure 3: Horizontal offset domain CIGs with the true velocity obtained by reverse-time migration. (a) For relatively flat sediments, the energy focuses well at zero offset; (b) For steep salt flanks, the energy leaks to far offsets and the frequency is low.

guojian1/. hxgathers

By equations 14 and 15, the offset domain CIGs in tilted coordinates $I_{x'}(x', z', h_{x'})$ can be decomposed into horizontal offset CIGs and vertical offset CIGs. Vertical offset domain CIGs and horizontal offset domain CIGs of all possible plane-wave sources are stacked after being rotated back to vertical Cartesian coordinates. Being transformed to angle domain CIGs, they are merged using equation 8, as with reverse-time migration.

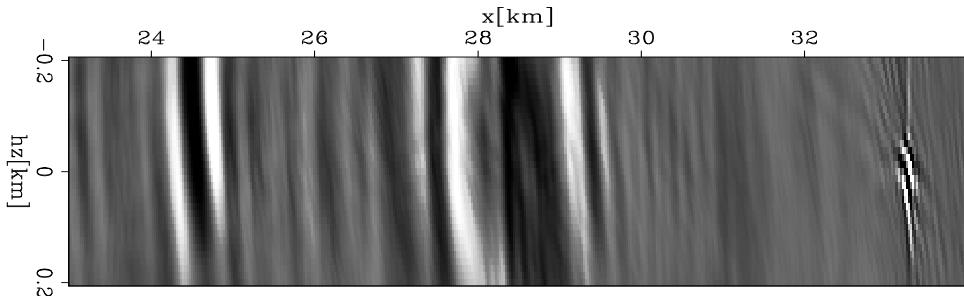


Figure 4: Vertical offset domain CIGs with the true velocity obtained by reverse-time migration. For the steeply dipping salt flank at $x = 33.2$ km, the energy focuses well at zero offset. For the near-flat sediments at $x = 24.5$ km, the energy leaks to far offsets.

guojian1/. hzgather

NUMERICAL EXAMPLES

We apply both reverse-time migration and plane-wave migration in tilted coordinates on the BP velocity benchmark dataset (Billette and Brandsberg-Dahl, 2005). Figure 1 shows the velocity model of the dataset. The maximum offset is 15 km, which is much larger than that in a realistic case. We mainly focus on the area from $x = 20$ km to $x = 35$ km. Figure 2 shows the image of that area obtained by plane-wave migration in tilted coordinates. In this area, there are both steep salt flanks and near-flat sediments. We run migration with

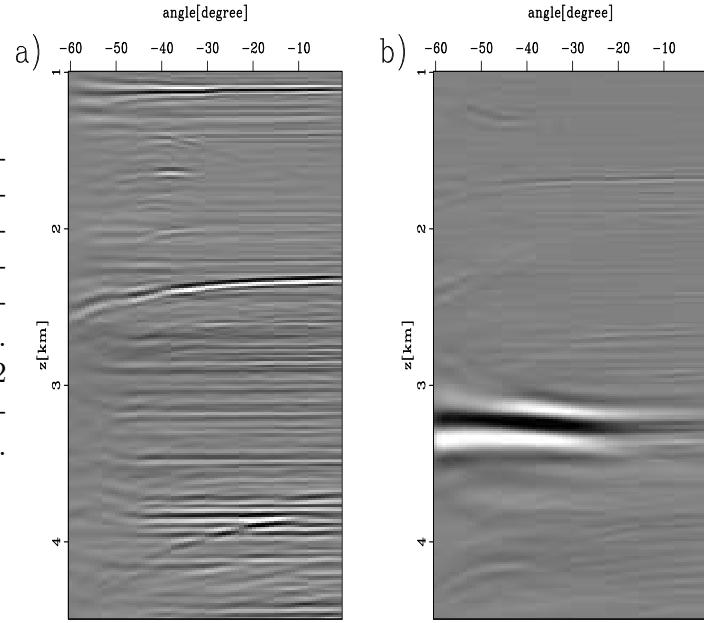


Figure 5: Horizontal angle domain CIGs with the true velocity obtained by reverse-time migration. (a) For the near-flat sediments at $x = 23$ km, the gathers are flat except the multiples. (b) For the steep salt flank at 33.2 km, the gathers are smeared because of the horizontal-offset stretch.

`guojian1/. axgathers`

both the true velocity and the velocity that is 3 percent slower than the true one. We start with showing examples that both horizontal and vertical CIGs of reverse-time migration are not robust where there are a full-range of dip directions. Then we compare angle domain CIGs of plane-wave migration in tilted coordinates with those of reverse-time migration obtained by merging the horizontal and vertical CIGs.

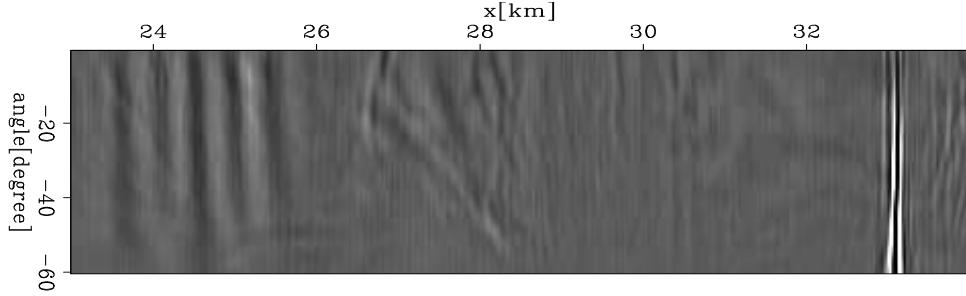


Figure 6: Vertical angle domain CIGs with the true velocity obtained by reverse-time migration. The CIGs of the sediments at $x = 24.5$ km are smeared and the CIGs for the salt flank are at the same horizontal location for all angles. `guojian1/. azgather`

Figure 3 shows the horizontal subsurface offset domain CIGs with the true velocity obtained by reverse-time migration. Figure 3(a) shows the horizontal offset domain CIGs of the near-flat sediments at $x = 23$ km. The energy mostly focuses well at zero offset. Notice that the multiple energy (at $z = 2.3$ km and $z = 3.9$ km) does not focus at zero offset. Figure 3(b) shows the horizontal offset domain CIGs of the steep salt flank at $x = 33.2$ km. The energy leaks to far offsets because of the stretch of the horizontal subsurface offset at steep reflectors. Figure 4 shows the vertical subsurface offset domain CIGs at $z = 3.25$ km. For the steep salt flank at $x = 33.2$ km, the energy focuses well at zero offset while the energy leaks to far offsets for the near-flat sediments at $x = 24.5$ km. As in the theoretical analysis, horizontal offset domain CIGs are good for near-flat reflectors while vertical ones

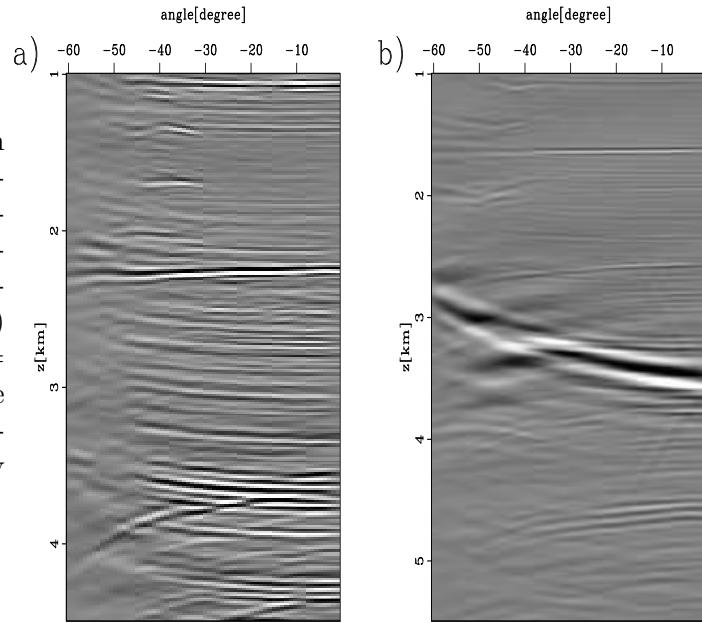


Figure 7: Horizontal angle domain CIGs with the 3 percent slower velocity obtained by reverse-time migration. (a) For the near-flat sediments at $x = 23$ km, the gathers show reasonable curvature. (b) For the steep salt flank at $x = 33.2$ km, the gather is a little smeared. The moveout is not reasonable, given the 3 percent velocity error. `guojian1/. ax97gathers`

are good for steep reflectors.

Figures 5 and 6 show the horizontal and vertical angle domain CIGs with the true velocity obtained by reverse-time migration, respectively. Figure 5(a) shows the horizontal angle domain CIGs of near-flat sediments at $x = 23$ km. The gathers are flat since we use the true velocity in migration. Notice that the gathers of multiples bend down at $z = 2.3$ km and $z = 3.9$ km. Figure 5 (b) shows the horizontal angle domain CIGs of the steep salt flank at $x = 33.2$ km. The horizontal gathers of the salt flank look smeared because of the offset stretch. In contrast, in Figure 6 the vertical angle domain CIGs of the steep salt flank (at $x = 33.2$ km) are at the same horizontal location for all angles but those of the near-flat sediments ($x = 24.5$ km) look smeared.

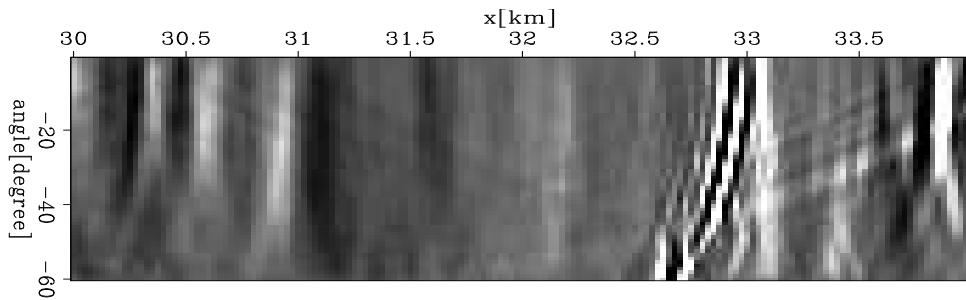


Figure 8: Vertical angle domain CIGs with the 3 percent slower velocity obtained by reverse-time migration. The gather of the salt flank at $x = 33$ km shows reasonable moveout. The gathers of the sediments at $x = 30.5$ km look smeared. `guojian1/. az97gather`

Figures 7 and 8 show a migration with the 3 percent slower velocity in the horizontal and vertical angle domain CIGs, respectively. The horizontal location for Figure 7(a) and (b) are at $x = 23$ km and $x = 33.2$ km, respectively. And the vertical location for Figure 8 is at $z = 3.5$ km. The horizontal angle domain CIGs for the sediments bend up with

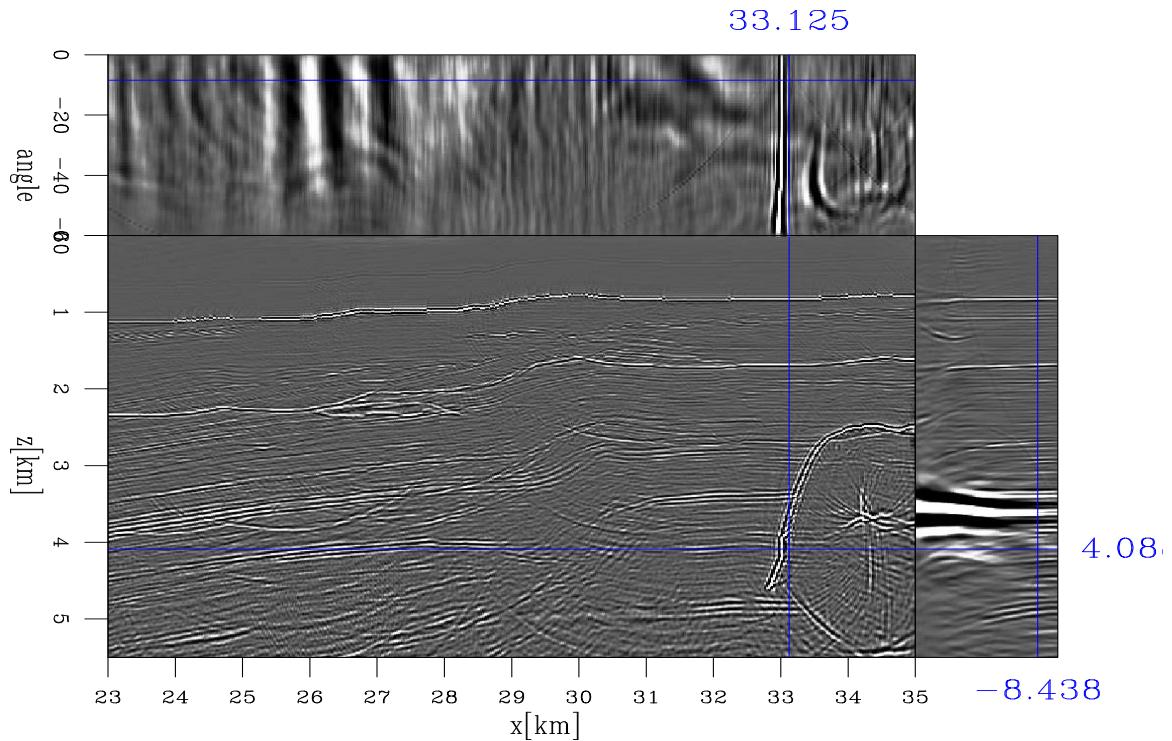


Figure 9: Angle domain CIG cube with the true velocity obtained by reverse-time migration.
`guojian1/.cigtruerm`

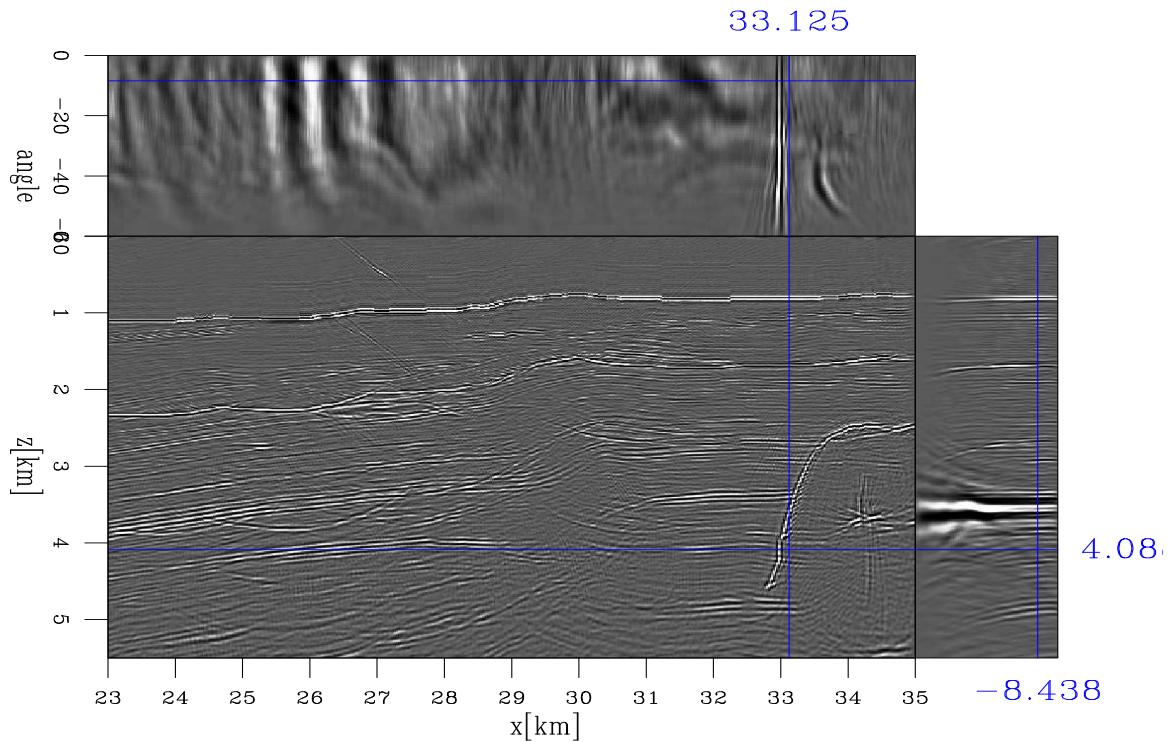


Figure 10: Angle domain CIG cube with the true velocity obtained by plane-wave migration in tilted coordinates.
`guojian1/.cigtruelt`

reasonable moveouts in Figure 7(a), but the horizontal angle domain CIGs for the salt flank look smeared in Figure 7(b). The moveout in the CIGs of the salt flank is too large given the 3 percent velocity error. In contrast, in Figure 8 the moveout in the vertical angle domain CIGs of the salt flank (at $x = 33$ km) is reasonable, but the vertical CIGs of sediments at $x = 30.5$ km look smeared in Figure 8.

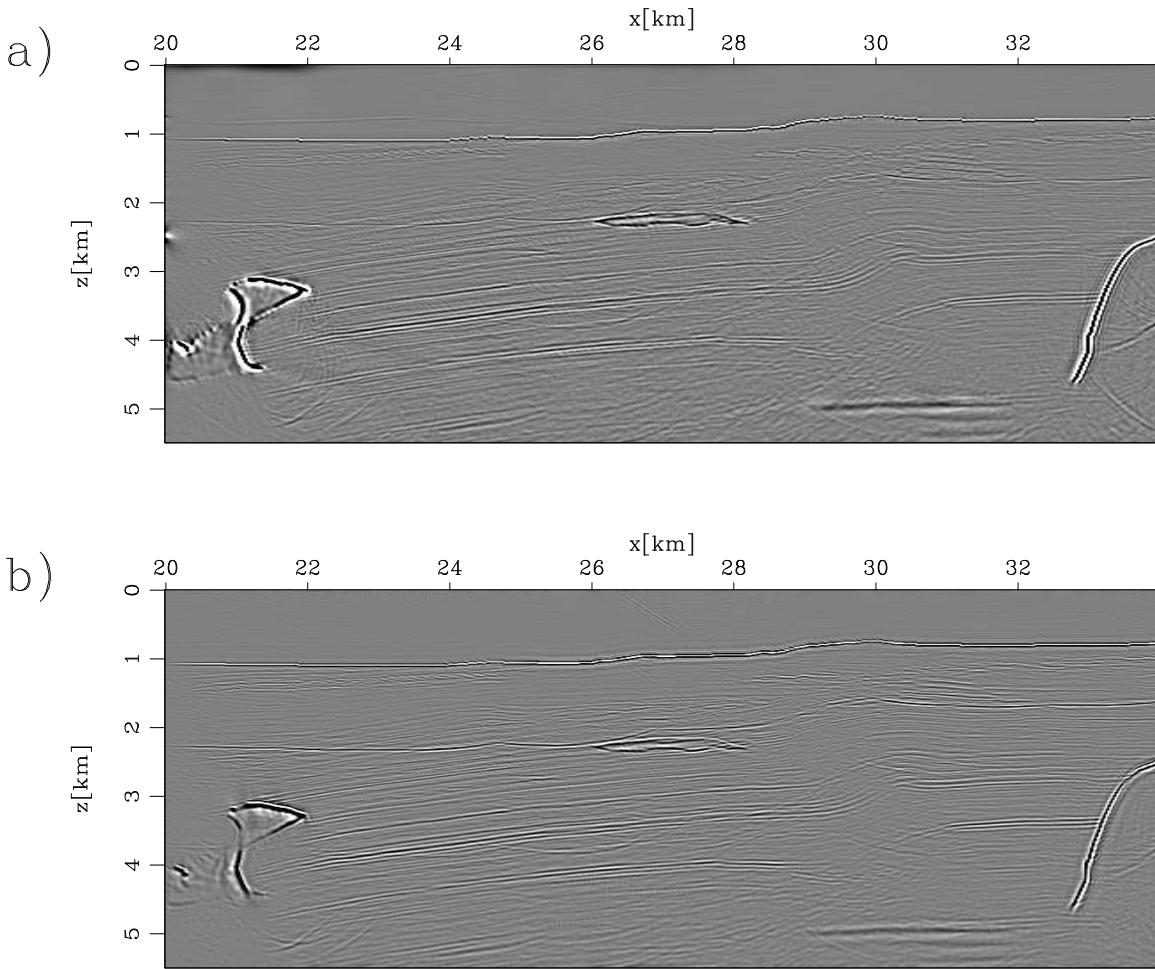


Figure 11: Stack images from angle domain CIGs with the true velocity: (a) reverse-time migration; (b) plane-wave migration in tilted coordinates. `guojian1/.stack`

As with the theoretical analysis, Figures 3 to 8 demonstrate that in reverse-time migration both horizontal and vertical CIGs are not robust in complex area, where there are reflectors with a full range of dip. To obtain reliable CIGs, we merge horizontal and vertical angle domain CIGs by applying equation 8. We can also obtain reliable CIGs by plane-wave migration in tilted coordinates. In the previous section, we discussed two ways to generate angle domain CIGs in plane-wave migration in tilted coordinates. We choose the former one for the following examples. We transform offset domain CIGs into angle domain CIGs in tilted coordinates and then rotate the angle domain CIGs back to vertical Cartesian coordinates. Figures 9 to 12 compare the angle domain CIGs obtained by reverse-time migration and plane-wave migration in tilted coordinates.

After 2-D migration, angle domain CIGs are a 3-D cube $I(x, z, \gamma)$. Conventionally, we

look at vertical sections of the cube $I(x = x_0, z, \gamma)$. However, the image point moves along the direction normal to the apparent geologic dip of the reflector when the velocity is not correct (Biondi and Symes, 2004). Therefore, the CIG of a image point should be viewed in the direction normal to its apparent geologic dip. Otherwise, for a steeply dipping reflectors, if we still look at the CIG cube in a vertical direction, most of the energy in the CIG we see belongs to image points in its neighborhood.

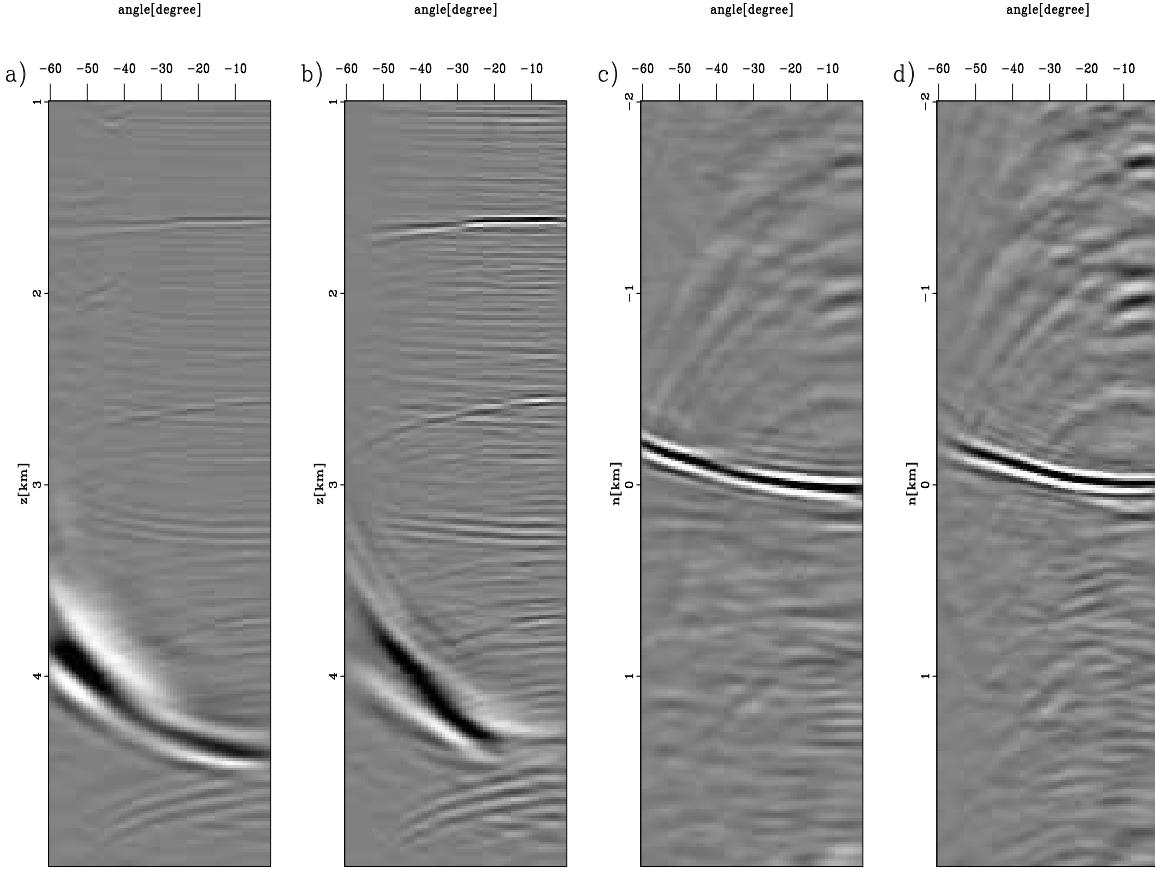


Figure 12: The angle CIGs with the 3 percent slower velocity. (a) The vertical view of angle domain CIGs obtained by reverse-time migration at $x = 32.6$ km. (b) The vertical view of angle domain CIGs obtained by plane-wave migration in tilted coordinates at $x = 32.6$ km. (c) The normal-direction view of angle domain CIGs obtained by reverse-time migration at $x = 32.6$ km, $z = 4.4$ km. (d) The normal-direction view of angle domain CIGs obtained by plane-wave migration in tilted coordinates at $x = 32.6$ km, $z = 4.4$ km.

guojian1/. amerg9770

Figures 9 and 10 respectively show the CIG cubes with the true velocity obtained by reverse-time migration and plane-wave migration in tilted coordinates. In both figures, the top panel shows the CIGs in the horizontal direction and the side panel shows the CIGs in the vertical direction. Reverse-time migration has better large-angle energy compared to plane-wave migration in tilted coordinates, but otherwise they are comparable. The offset of this dataset is unrealistically large and thus the maximum opening angle is unrealistically large. Plane-wave migration in tilted coordinates is still based on the one-way wave equation. Both source and receiver wavefields are extrapolated in the same coordinates, so when opening

angle is very large (more than 60°), the angle difference of source and receiver rays is large and thus one of them cannot be modeled accurately. But there is no angle limitation in reverse-time migration, therefore its large-angle energy are better handled than plane-wave migration in tilted coordinates. Given the realistic offset in real datasets, opening angles are usually smaller than 50° . Therefore, angle domain CIGs of the two migrations are comparable for a real dataset. Notice that in both figures the CIGs of the steep salt flank look smeared in the side panel and those of the sediments look smeared in the top panel. This demonstrates that angle domain CIGs should be viewed in the direction normal to the dip direction. Figure 11 compares the stacks of the angle domain CIGs along the angle axis. Figure 11(a) is obtained by reverse-time migration and Figure 11(b) is obtained by plane-wave migration in tilted coordinates and the images are comparable.

Figure 12 shows the comparison of angle domain CIGs with the velocity that is 3 percent slower than the true one. Figures 12(a) and (b) are the vertical views of the angle domain CIGs obtained by reverse-time migration and plane-wave migration in tilted coordinates, respectively. The horizontal location for them is at $x = 32.6$ km, where there are near-flat sediments at the shallow part and steep salt flanks at $z = 4.4$ km. The events bending down are multiples. As with the CIGs with the true velocity, reverse-time migration has better far-angle energy compared to plane-wave migration in tilted coordinates. In both figures, the moveout of the CIGs of the sediments is reasonable because the viewing direction is almost normal to their apparent geologic dip direction. But the CIGs of the salt flank (at $z = 4.4$ km) look smeared and forked because the viewing direction almost parallels its dip and most far-angle energy in the CIG belongs to the image points in its neighborhood. The curvature of the CIG is not reasonable and the moveout is too large, given the 3 percent velocity error.

Figures 12(c) and (d) are the normal-direction view of the angle domain CIGs obtained by reverse-time migration and plane-wave migration in tilted coordinates, respectively. The location of the event is at $x = 32.6$ km, $z = 4.4$ km, where the salt flank is present. Its apparent geologic dip is about 70° . The vertical axis in both panels is the direction normal to the apparent geologic dip of the reflector. Similarly, reverse-time migration has better far-angle energy, otherwise Figures 12(c) and (d) are comparable. Given the 3 percent velocity error, both the CIGs show a reasonable curvature.

CONCLUSIONS

Conventional horizontal angle domain CIGs are not useful for steeply dipping reflectors because of the offset stretch. Reverse-time migration provides robust angle domain CIGs for both steep and near-flat reflectors by merging horizontal and vertical CIGs. Plane-wave migration in tilted coordinates can also provide reliable CIGs because the propagation direction is closer to extrapolation direction and the subsurface offset direction is closer to the dip direction in tilted coordinates. For both methods, angle domain CIGs of a image point should be viewed in the direction normal to its dip direction. For the BP velocity benchmark dataset with unrealistically large offsets, comparisons show that reverse-time migration has better large angle energy, otherwise the angle domain CIGs from these two methods are comparable. When the velocity is not correct, the CIGs from both migration methods provide useful moveout information.

ACKNOWLEDGMENTS

We thank BP for making the dataset available.

REFERENCES

- Bear, L., T. Dickens, J. Krebs, J. Liu, and P. Traynin, 2005, Integrated velocity model estimation for improved positioning with anisotropic PsDm: *The Leading Edge*, **24**, 622–634.
- Billette, F. and S. Brandsberg-Dahl, 2005, The 2004 BP velocity benchmark: 67th meeting, Expanded Abstracts, B035, Eur. Assn. Geosci. Eng.
- Biondi, B. and G. Shan, 2002, Prestack imaging of overturned reflections by reverse time migration, *in* Expanded Abstracts, 1284–1287, Soc. of Expl. Geophys., 72nd Ann. Internat. Mtg.
- Biondi, B. and W. Symes, 2004, Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging: *Geophysics*, **69**, 1283–1298.
- Clapp, R., 2000, Geologically constrained migration velocity analysis, *in* Ph.D. thesis, Stanford University.
- Rickett, J. E. and P. C. Sava, 2002, Offset and angle-domain common image-point gathers for shot-profile migration: *Geophysics*, **67**, 883–889.
- Sava, P. C. and S. Fomel, 2003, Angle-domain common-image gathers by wavefield continuation methods: *Geophysics*, **68**, 1065–1074.
- Shan, G. and B. Biondi, 2004, Imaging overturned waves by plane-wave migration in tilted coordinates: 74th Ann. Internat. Mtg., Expanded Abstracts, 969–972, Soc. of Expl. Geophys.
- Shan, G., R. Clapp, and B. Biondi, 2007, 3d plane-wave migration in tilted coordinates: 77th Ann. Internat. Mtg., Expanded Abstracts, 2190–2193, Soc. of Expl. Geophys.
- Shan, G. and G. Zhang, 2003, Equivalence between shot-profile and source-receiver migration: SEP-Report, **113**, 121–126.
- Tam, K. and V. Perez-Mendez, 1981, Tomographical imaging with limited-angle input: *J. Opt. Soc. Am.*, **71**, 582–592.
- Toldi, J., 1985, Velocity analysis without picking, *in* Ph.D. thesis, Stanford University.

Phase unwrapping of angle-domain common image gathers

Francesco De Zan and Biondo Biondi¹

ABSTRACT

In this paper we adapt a phase unwrapping algorithm to estimate the depth shift in Angle-Domain Common Image Gathers (ADCIGs). We show how to set up a linear system of equations tailored to the seismic case and how to solve it by minimizing an L^0 measure via iterations of weighted least-squares problems. For this procedure a meaningful choice of initial weights is crucial.

We propose to unwrap jointly several angle gathers and show that this can overcome sampling deficiencies in the angle domain, such as those that come from processing a limited number of subsurface offsets for angle-gather generation.

INTRODUCTION

Migration velocity analysis is a class of techniques used for updating the velocity field, starting from a migrated image. These techniques are based on linking the curvature of image gathers (for instance Angle-Domain Common Image Gathers) to migration-velocity error. When this relation is linearized, it leads to a simple inversion problem. However the linearization of the wave field with the first-order Born approximation comes with an important limitation: it can handle delays only up to a fraction of the wavelength.

This problem has been given a possible solution in Sava and Biondi (2004). A viable alternative is to transform the ADCIGs to the Fourier domain and do phase unwrapping there, which is roughly equivalent to determining the delay in the original domain. This is suggested in Sava and Biondi (2003) concerning the Rytov approximation. Synthetic Aperture Radar Interferometry and Magnetic Resonance Imaging literature offers many examples of unwrapping techniques. Although they share common principles, each one must be carefully tuned to the specific application.

In this paper we adapt a phase-unwrapping algorithm to unwrap ADCIGs. We describe how to formulate the unwrapping problem, solve it, and test it on a simple synthetic case. We show that by unwrapping jointly several gathers we can overcome some sampling limitations in the angle domain.

PHASE UNWRAPPING

When a signal is delayed, the phases of its Fourier spectral components are rotated proportionally. However, due to the periodic nature of Fourier components, the observable phases are always limited to the interval $[-\pi, \pi]$; i.e. there is no record of the number of entire cycles that may have intervened. This phenomenon is usually referred to as phase ambiguity,

¹**e-mail:** francesco.dezan@polimi.it

because different delays can correspond to the same observed phase shift. Phase unwrapping is the problem of recovering the number of 2π cycles that unambiguously reconstructs the original delay.

Phase unwrapping can be approached in various ways. In this work we follow the recipe presented in Ghiglia and Romero (1996) and Ghiglia and Romero (1994), where the unwrapped phase is found as the solution of a linear system.

The general principle is that even though the unwrapped phases are usually outside the interval $[-\pi, \pi]$, differences in unwrapped phases of “neighboring” points are often included in that interval, so that they can be recovered also from the wrapped values, which are available. Thus we write a number of equations that describe differences in the unwrapped phases and rely on the solution of the system to integrate those differences.

Of course some of the original equations are wrong (they assume the phase difference to be within the interval $[-\pi, \pi]$, when in fact it is not) and conflict with others. The algorithm that solves the system will eventually have to make a decision and discard some equations, favoring others.

First we have to define a graph that represents the equations we will use. Then we describe the algorithm for the solution of the system.

GRAPH AND LINEAR SYSTEM

In our domain, the signal is a function of angle (α), vertical wavenumber (kz) and midpoint inline position (x). Each equation we include in our system connects two points, so that each equation corresponds to a link and the entire system to a graph. A simple cartesian grid was used in the angle- kz plane. Each point is connected to its four neighbors, so for example the point $A(\alpha, kz, x)$ is connected to $A(\alpha, kz \pm 1, x)$ and to $A(\alpha \pm 1, kz, x)$. Points at the boundary of the domain have fewer connections.

To increase the robustness of the unwrapping procedure we do not consider each gather independently but connect several gathers in the inline direction, presuming continuity along that axis too. So $A(\alpha, kz, x)$ is also connected to $A(\alpha, kz, x \pm 1)$, raising to six the number of equations in which a given point typically appears.

An example of the basic equation is the following:

$$\phi(\alpha, kz, z) - \phi(\alpha - 1, kz, z) = [\varphi(\alpha, kz, z) - \varphi(\alpha - 1, kz, z)]_{2\pi} \quad (1)$$

where the other cases are straightforward. The expression $[\cdot]_{2\pi}$ represents the wrapping operator, or the remainder after integer division by 2π ; ϕ are the unwrapped values and φ their wrapped, observed counterparts.

The system is not complete without some boundary equations that serve as a phase reference. We set to zero the zero-angle phases of a reference gather for all the considered wavenumbers.

The whole system can be written in matrix form:

$$\mathbf{G}\Phi = \mathbf{d}, \quad (2)$$

where \mathbf{G} is the graph incidence matrix plus border equations, Φ is the unknown vector of unwrapped phases and \mathbf{d} is a function of the observed phases (the wrapped differences). \mathbf{G} is a very sparse matrix with typically two non-zero entries per row.

L^0 SOLUTION AND WEIGHTED ITERATIONS

The solution of the above system of equations (2) can be found by minimizing a chosen indicator. Given the particular nature of the unwrapping problem, the L^0 measure is considered a good choice. The point is that we are not looking for a smooth solution that tries to accomodate all equations (like the L^2 norm does); we instead want the algorithm to make hard choices between alternatives and to produce a solution that satisfies, with no approximation, the highest possible number of equations. Ghiglia and Romero (1996) describe a way to minimize the L^0 measure via successive steps that are computed solving weighted least squares problems. Ghiglia and Romero's algorithm is more general and provides a way to minimize any L^p measure, with p in $[0, 2]$. An application of the L^1 -norm is found in Lomask (2006).

The following is the outline of the suggested algorithm, (setting $p = 0$ for our specific case):

- Set up the initial weights, \mathbf{W}_0 .
- Set $i = 0$.
- Until i has reached the maximum number of iterations, repeat the following steps:
 1. Solve (to convergence) the Weighted Least Square (WLS) system:

$$\mathbf{G}^T \mathbf{W}_i \mathbf{G} \Phi_i = \mathbf{G}^T \mathbf{W}_i \mathbf{d}. \quad (3)$$

2. Compute new weights according to the formula

$$W_{i+1}(n) = \frac{\epsilon_0}{\epsilon_0 + |\mathbf{g}(n)\Phi_i - d(n)|^{2-p}} W_i(n). \quad (4)$$

3. Increase i by 1.
- End.

\mathbf{W}_i is a diagonal matrix with elements $W_i(n)$, the weights for each equation. The vector $\mathbf{g}(n)$ is the n^{th} row of \mathbf{G} , so that $\mathbf{g}(n)\Phi_i$ is a scalar and ϵ_0 an adequately small value. For efficiency reasons the WLS step is implemented by preconditioned conjugate gradient.

With this iterative mechanism and this particular choice of weights, each equation which is not satisfied at a given iteration is almost ignored for the next iteration, provided that more trusted equations exist that involve the same points.

Thus the choice of the initial weights is critical to yielding good results. We preliminarily used the amplitude information as a measure for the phase reliability: each equation was given a weight proportional to the harmonic average between the amplitudes of the two points involved.

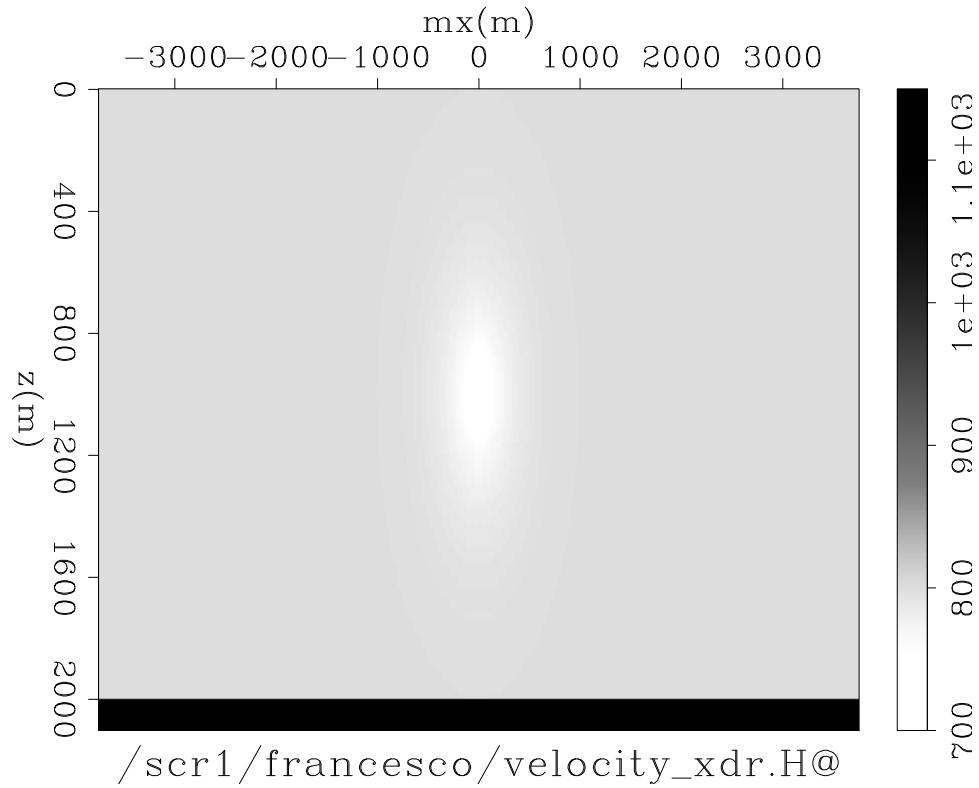


Figure 1: The velocity used for modeling the seismic data.

EXAMPLE

For a first test we create a model with a negative Gaussian anomaly in an constant velocity background (see Fig.1) and migrate (incorrectly) the modeled data using a constant velocity model. After migration we apply an offset-to-angle transformation using 33 offsets. The result is seen in Figure 2. As expected, the angle gathers show some deviation from being flat. This curvature can ideally be used to correct the migration velocity and improve the focusing. Notice the jump at near angles because of the insufficient angle sampling, a consequence of the number of processed offsets.

We pick 33 gathers equally spaced in the inline direction, from a position where the presence of the anomaly is unfelt to directly under the anomaly. After windowing, we tranform the z axis so that for each gather we have now a kz -angle panel instead of the original z -angle panel. Applying the described unwrapping procedure, we obtain the result shown in Figure 3, which refers to the gather right under the anomaly. The left image is the original wrapped phase, referred to the 0 angle for visualization convenience. The right image is the corresponding unwrapped phase. Please note that the wrapped phase field is not devoid of ambiguity, i.e. the integration path is not irrelevant. However the algorithm is able to cut the phase at approximately the right position, at low angles for high frequencies. This is possible because of the choice of initial weights and the linking of several gathers together.

To confirm the result, we apply the same algorithm to the case where we have computed

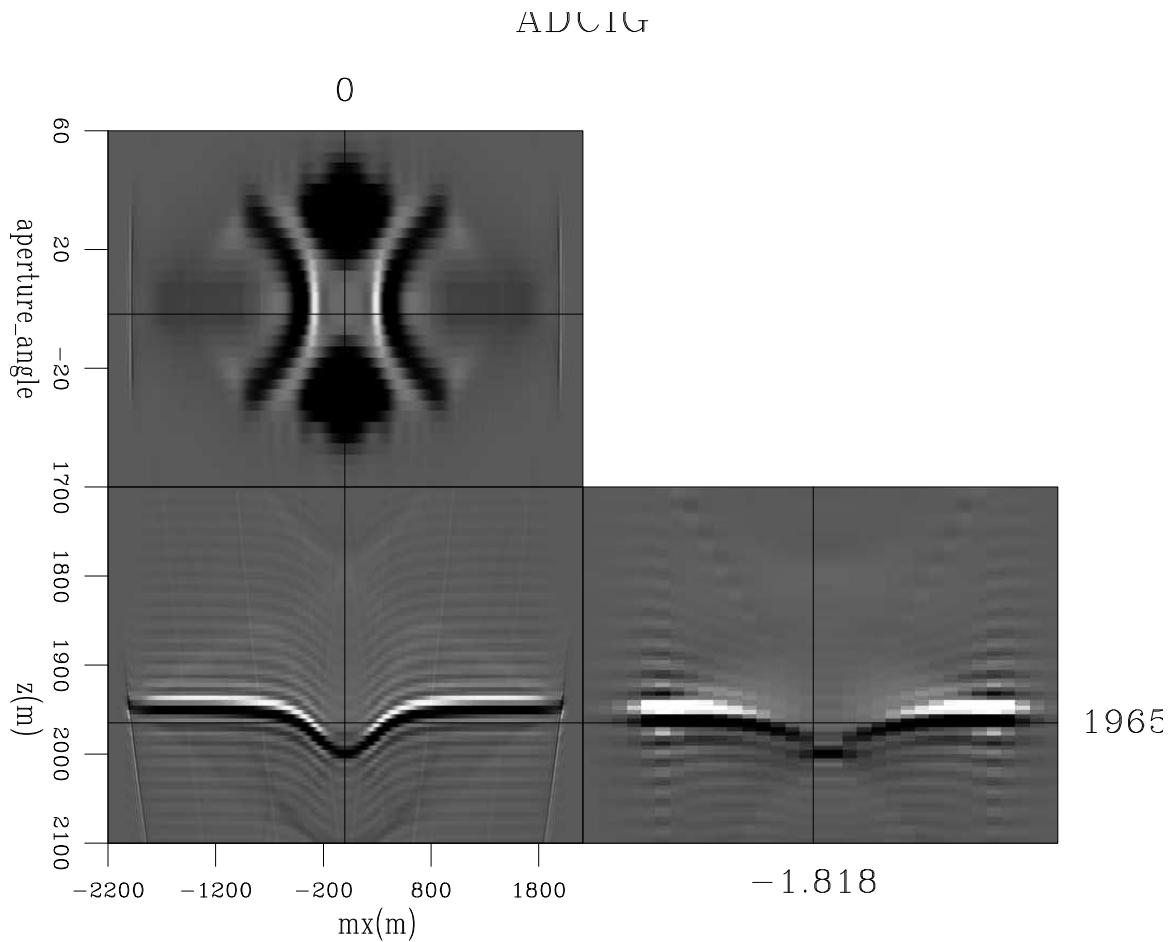


Figure 2: An Angle-Domain Common Image Gather computed using 33 offsets.
francesco1/. gath-1

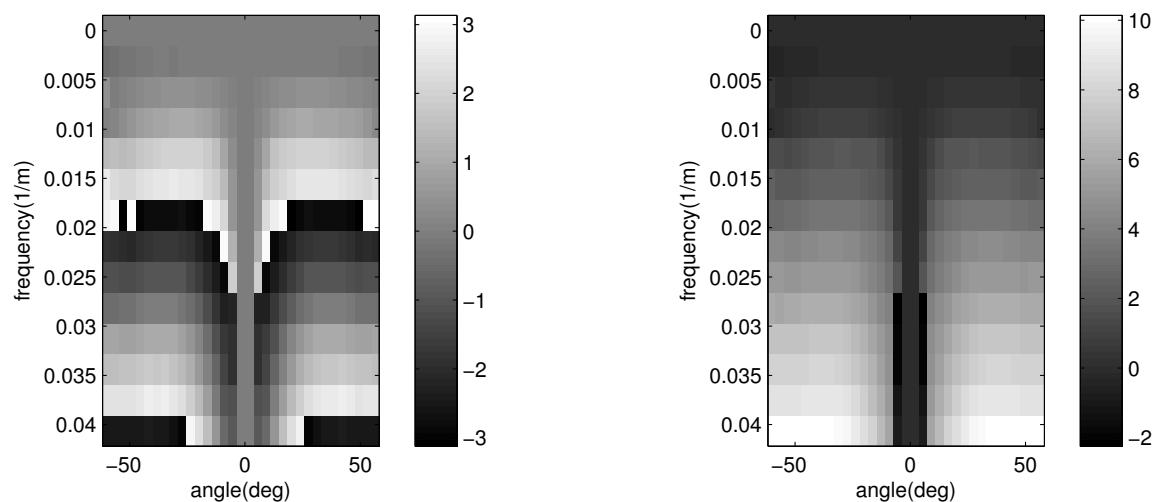


Figure 3: Wrapped (left) and unwrapped (right) phase for an ADCIG computed using 33 offsets. Phase measured in radians. francesco1/. unwrapping-33

a larger number of offsets, 65 instead of 33. This increases the resolution in the angle domain (see Fig. 4), and the discontinuity disappears from the z -angle domain. The same happens in the kz -angle domain (see Fig. 5, left), where we no longer see a jump in the wrapped phase. The unwrapped phase is comparable to the previous one.

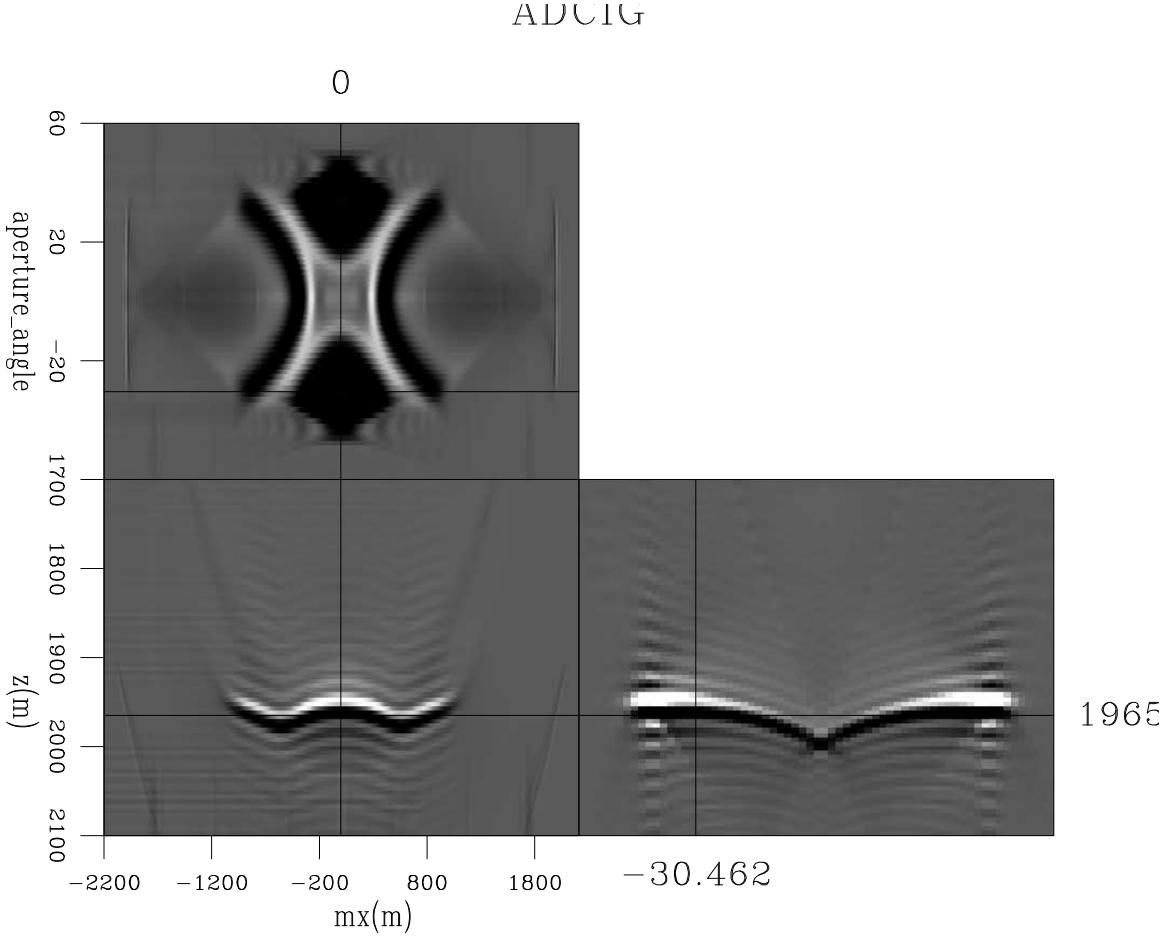


Figure 4: An ADCIG computed using 65 offsets. [francesco1/.gath-2]

When dispersion effects can be ignored, it is possible to derive from the unwrapped phases a single number representing the delay for a given gather and a given angle. We interpolate lines into reliable unwrapped phase values, again using amplitude as a reliability criterion. The slopes of the lines correspond to the z -domain delays. Figures 6 and 7 display these delays in terms of samples for the two cases, with 33 and 65 offsets. For visualization purposes we subtracted the average delay for each gather, so that the effect of the anomaly is more clearly visible. A mask is used because midpoints have different angular coverage.

Phase unwrapping makes it possible to treat different wavenumbers independently, i.e. to take advantage of the information carried by the dispersion. Even in this simple example we can actually see some dispersion effects. Figure 8 shows the delay predicted by the single wavenumber for all gather-angle pairs, after subtraction of the “average” delay. Lower wavenumbers have a higher dispersion, but higher ones are more prone to unwrapping problems.

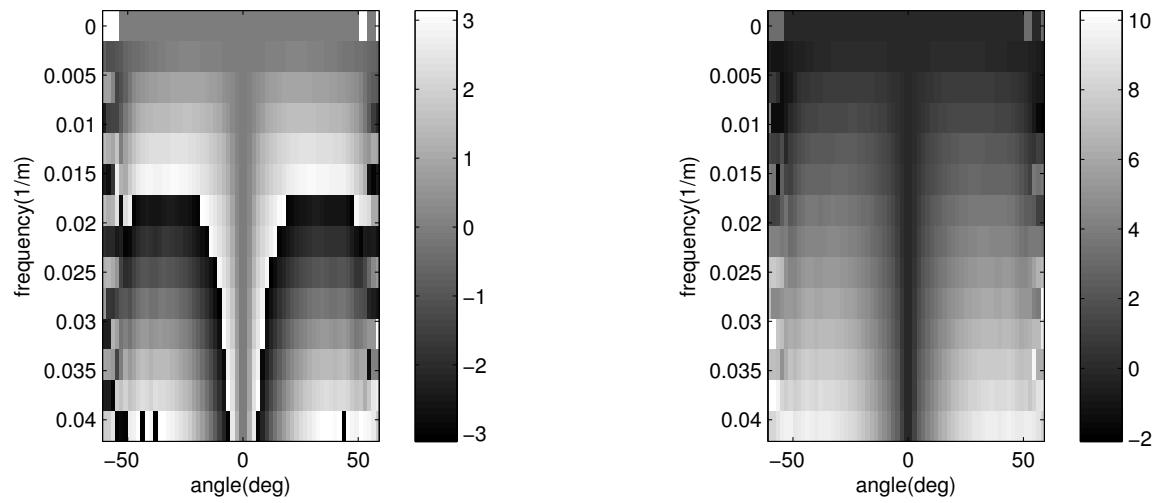


Figure 5: Wrapped (left) and unwrapped (right) phase for an ADCIG computed using 65 offsets. francesco1/. unwrapping-65

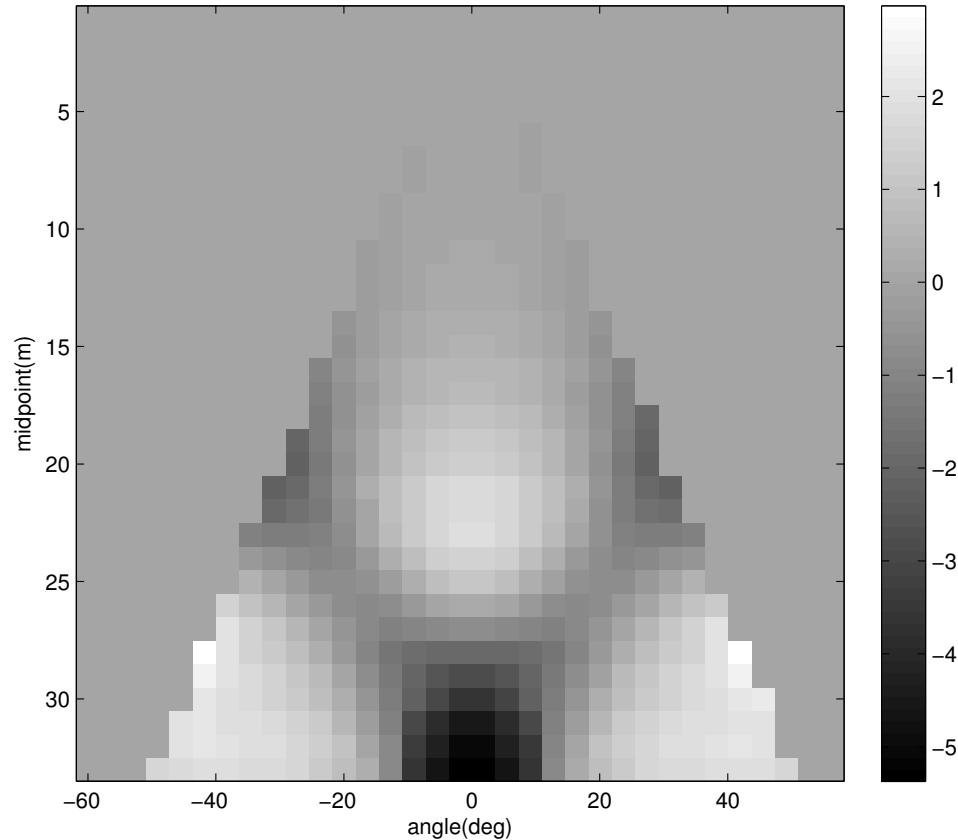


Figure 6: The delay (in samples) for a number of ADCIGs as a function of aperture angle. Gathers were computed using 33 offsets. francesco1/. delay-1

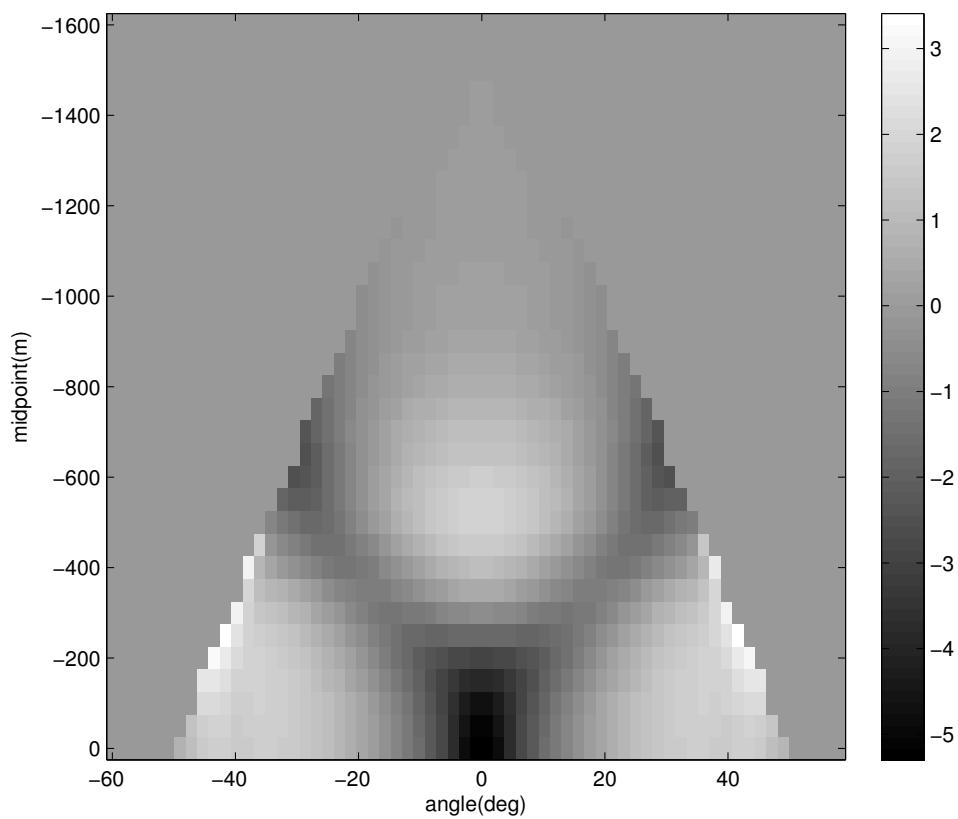


Figure 7: The delay (in samples) for a number of ADCIGs as a function of aperture angle. Gathers were computed using 65 offsets. francesco1/. delay-2

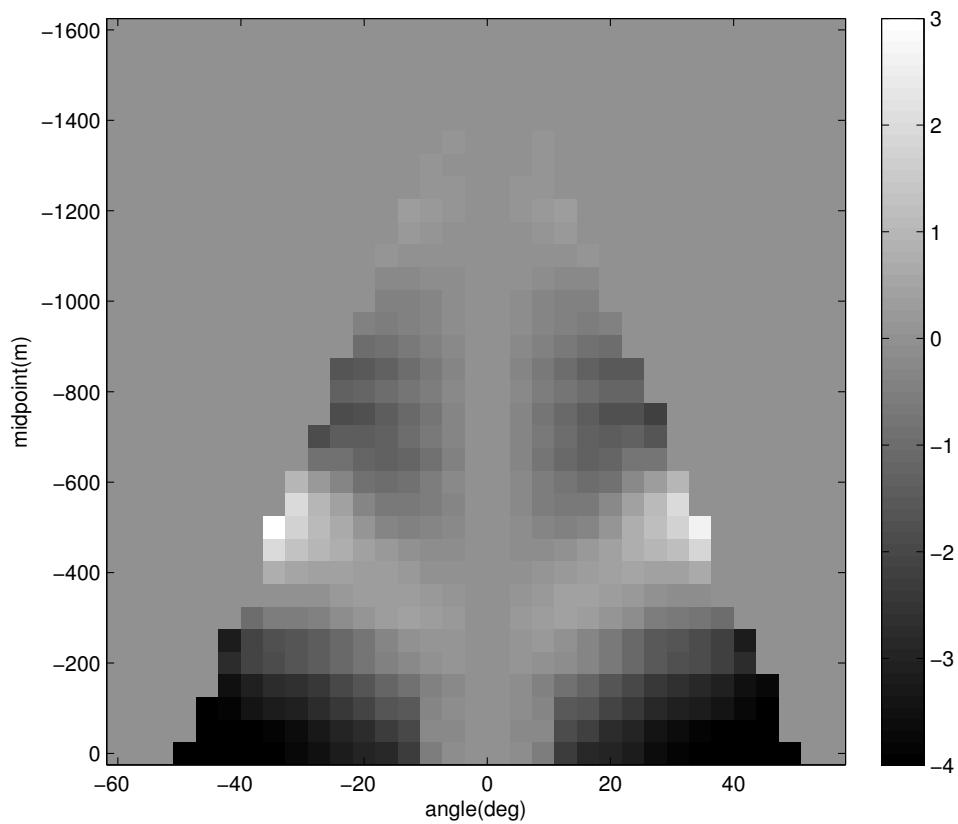


Figure 8: An example of the residual delay (converted to samples) for a given kz , after subtracting the delay identified using all wavenumbers. francesco1/. dispersion2

CONCLUSIONS

Phase unwrapping in the kz -angle domain can be used to evaluate the delay of angle gathers, a preliminary step towards velocity analysis. Simple numerical test indicates that some limitations that come from angle sampling or illumination can be overcome by considering jointly a number of gathers from the same horizon. The application of this unwrapping technique may require more investigation about image windowing (which should ideally follow the still-unknown gather curvature) and gather picking to ensure phase continuity.

REFERENCES

- Ghiglia, D. C. and L. A. Romero, 1994, Robust two-dimensional weighted and unweighted phase unwrapping that uses fast transforms and iterative methods: *J. Opt. Soc. Am. A*, **11**, 107–117.
- , 1996, Minimum l^p -norm two-dimensional phase unwrapping: *J. Opt. Soc. Am. A*, **13**, 1999–2013.
- Lomask, J., 2006, Seismic volumetric flattening and segmentation: Presented at the SEP–126, Stanford Exploration Project.
- Sava, P. and B. Biondi, 2003, Wave-equation MVA: Born Rytov and beyond: SEP-Report, **114**, 83–94.
- , 2004, Wave-equation migration velocity analysis. I. Theory: *Geophysical Prospecting*, **52**, 593–606.

INTRODUCTION

Angle-domain common image gathers (ADCIGs), created from downward-continuation or reverse time migration, can provide useful lithological and velocity information (Prucha et al., 1999). In geologically complex areas, poor illumination causes undesirable kinematic effects and amplitude variations along the angle axis (Prucha et al., 2000; Valenciano, 2006).

The subsurface-offset-to-angle transformation consists of a radial trace transform in the Fourier space with some regularization in the angle direction (Sava and Fomel, 2000) or slant-stack in the physical space plus an additional transformation from offset ray-parameter to reflection angle (Prucha et al., 1999). The regularization, to some extent, can diminish the amplitude variation caused by poor illumination. The more accurate solution to the illumination problem, however, is achieved by computing a regularized least-squares inverse image (Clapp, 2005) rather than the simply the adjoint (migration). The inverse image problem can be solved either by computing the Hessian implicitly (Clapp, 2005) or explicitly Valenciano (2006), preferentially, in the reflection-angle domain or, without any physically meaningful regularization direction, in the subsurface-offset domain.

In the reflection-angle domain, the inverse image problem which explicitly computes the Hessian can be performed according to two different strategies. First, by computing the angle-domain Hessian. Valenciano and Biondi (2006) proposed to obtain the angle-domain Hessian by applying the slant-stack technique to compute ADCIGs on the subsurface-offset Hessian. They noticed that the resulting angle-domain Hessian for a model with a Gaussian velocity anomaly lacked the resolution to determine which angles were more illuminated. Recently, Fomel (2003) introduced the theoretical framework of the oriented wave equation, under which computing the angle-domain Hessian could be promising. In the other approach, the angle-domain Hessian can be evaluated by chaining the offset-to-angle operator and the subsurface-offset Hessian (Valenciano and Biondi, 2005). Valenciano (2007), in this report, shows good results obtained by applying this strategy in Sigsbee dataset.

Here, I propose a general framework to map any information computed in the subsurface-offset domain to the angle-domain. The proposed approach relies on the asymptotic nature of the slant-stack transformation from subsurface-offset to angle domain. I first show the validity of the stationary-phase assumption for the offset-to-angle transformation, then describe a weighted transformation from subsurface-offset to reflection-angle domain, and finally illustrate the technique with the transformation of the diagonal of the Hessian in the subsurface-offset domain to the angle domain, yielding amplitude factors to compensate for illumination problems in ADCIGs. Additionally, I show the transformation of some off-diagonal terms which, at present, does not have a direct application in the amplitude correction problem.

PHASE BEHAVIOR OF THE OFFSET-TO-ANGLE TRANSFORMATION

The offset-to-angle transformation can be expressed by the integration of the subsurface-offset domain common image gathers (SODCIGs), $P(z, h)$, along a certain slanted path, according to the equation

$$Q(z, \gamma) = \int_A \varrho[P(z, h)]dh|_{z=\zeta(\gamma, h)}, \quad (1)$$

where $Q(z, \gamma)$ is the output ADCIG, z is depth, γ is the aperture angle, h is the subsurface offset, ϱ is the *rho*-filter which aims to yield the correct phase of the output ADCIG (?), A is the domain of integration that defines the range of subsurface offsets to be summed, and $\zeta(\gamma, h)$ is the slanted path given by

$$\zeta(\gamma, h) = z_0 + h \tan \gamma, \quad (2)$$

where z_0 is the depth coordinate at zero subsurface offset. A single reflector in a SODCIG can be represented by

$$P(z, h) = A(h)f(z - z_r(h)), \quad (3)$$

where $A(h)$ is an amplitude term whose value depends on the reflection coefficient, illumination and focusing, f is the depth domain representation of the seismic pulse, and z_r is the reflector depth. The fact that A and z_r are functions of h accommodates the focusing of reflector amplitudes at nonzero-subsurface offsets because of inaccuracies in migration velocity and problems in illumination. A SODCIG containing several reflectors can be described by the superposition of individual reflectors, each described by equation 3.

Equation 1, Fourier transformed to the k_z domain after inserting equation 3, reads

$$\hat{Q}(k_z, \gamma) = \sqrt{\frac{-ik_z}{2\pi}} F(k_z) \int_{-h}^h A(h) e^{-ik_z \Phi(\gamma, h)} dh, \quad (4)$$

where $\Phi(\gamma, h) = \zeta(\gamma, h) - z_r(h)$ is the phase function. Assuming that $A(h)$ is not itself an oscillating function, and considering the high- k_z regime, the argument of the integral in equation 4 rapidly oscillates, yielding negligible amplitudes for integration over a full period, except for the case where the phase function, $\Phi(\gamma, h)$, remains stationary. This condition is achieved in the vicinity of a point — the stationary point — in the SODCIG with a certain subsurface offset, h^* , where $\zeta(\gamma, h)$ is tangent to $z_r(h)$, or

$$\zeta(\gamma, h) = z_r(h)$$

$$\frac{\partial \zeta(\gamma, h)}{\partial h} = \frac{\partial z_r(h)}{\partial h},$$

estimated in $h = h^*$.

Equation 4 can be evaluated by the stationary-phase method. According to Bleistein (1984), under the assumption of a single stationary point in which the second derivative does not vanish, integrals like

$$I(\lambda) = \int_A f(t) e^{i\lambda\phi(t)} dt, \quad (5)$$

where $f(t)$ is a smooth and compact function, can be asymptotically approximated by

$$I(\lambda) \sim e^{i(\lambda\phi(c) + sgn(\phi''(c))\frac{\pi}{4})} f(c) \sqrt{\frac{2\pi}{\lambda |\phi''(c)|}}, \quad (6)$$

if $\lambda \rightarrow \infty$. The term $sgn(\phi''(c))$ corresponds to the signal of the second derivative of the phase function, $\phi(t)$, evaluated at the stationary point, c .

It turns out that the stationary phase formula of equation 4 is given by

$$\hat{Q}(k_z, \gamma) \sim \frac{A(h^*)}{\sqrt{|\Phi''(h^*)|}} F(k_z) e^{-ik_z \Phi(\gamma, h^*)}. \quad (7)$$

Finally, the inverse Fourier transform of equation 7 gives

$$Q(z, \gamma) \sim \frac{A(h^*)}{\sqrt{|\Phi''(h^*)|}} F(z - \Phi(\gamma, h^*)). \quad (8)$$

Equation 8 shows that the main contribution to the amplitudes in the ADCIG comes from the vicinity of the stationary point. The second derivative of the phase function with respect to h is basically the second derivative of z_r , as $\zeta(\gamma, h)$ is a straight line. If z_r is a straight event in the SODCIG, meaning that just a very small range of angles has been illuminated (Tang, 2007), there will be as many stationary points as subsurface offsets. In this situation, the integration interval is divided in such a way that each new interval contains only one stationary point, and the final result is the sum of all individual stationary point contributions. The other special case is when all the energy is focused at zero subsurface-offset, indicating good illumination for all reflection angles and migration with the correct velocity. It is a generalization of the previous case and is solved in the same way for various illumination angles.

WEIGHTED OFFSET-TO-ANGLE TRANSFORMATION

Bleistein (1987) describes a strategy to estimate parameters from the subsurface using different two images migrated with slightly different weights. Tygel et al. (1993) applied the same ideas to what they called a multiple-weight diffraction stack to obtain the stationary point location that in turn, along with source and receiver position, specifies the reflection ray. For instance, to estimate reflector dips one can compute two different migrated images, M_a and M_1 , using two distinct migration-weighting functions, say the migration angle (M_a) and simply a constant value of one (M_1). For M_a , the resulting amplitudes are weighted by the migration angles around the stationary point. In this region the migration operator and reflectors are tangent. Consequently the local average of the migration angle is an estimate of the reflector dip. So, the division M_a/M_1 results in an estimate of the dip of the reflectors.

The phase behavior of the offset-to-angle transformation shows that the main contribution for the image in the angle domain comes from the vicinity of the stationary point. Therefore, the use of the weighted stacking strategy (Bleistein, 1987) to map quantities computed in the subsurface-offset domain to the angle domain is straightforward. The mapping of certain attributes can be useful, for instance, to balance amplitudes in the angle domain.

In the following, the aim of the weighted offset-to-angle transformation is to compute weights to be applied on ADCIGs in such a way that amplitude variations due to illumination are attenuated. The weighted offset-to-angle transformation is represented by the computation of ADCIGs from SODCIGs previously multiplied by some parameter — in the present case, the subsurface-offset Hessian diagonals, $H(z, h)$ — defined in the subsurface-offset domain. The amplitudes of the resulting ADCIGs, according to the stationary phase results in the previous section, can be represented by

$$\tilde{Q}(z, \gamma) \sim H(z, h^*) \frac{A(h^*)}{\sqrt{|\Phi''(h^*)|}} F(z - \Phi(\gamma, h^*)). \quad (9)$$

where $H(z, h^*)$ are the averaged values of the subsurface-offset Hessian diagonals in the vicinity of the stationary point. The ADCIGs, $\tilde{Q}(z, \gamma)$, are to be divided by the non-weighted transformed results from equation 8, $Q(z, \gamma)$, using a regularization term, ϵ , in the denominator to avoid division by small numbers. Finally, a median filter in the (z, x) -plane (represented by $\langle \rangle_{(z,x)}$) for every angle section is applied to remove spurious amplitudes, thus providing an estimate of subsurface-offset parameter in the angle domain. The angle-transformed Hessian diagonals, $H(z, \gamma)$ are computed according to equation 10:

$$H(z, \gamma) \sim \langle \frac{\tilde{Q}Q}{Q^2 + \epsilon} \rangle_{(z,x)}. \quad (10)$$

The general formula of the subsurface-offset Hessian in the prestack-inversion problem is

$$\mathbf{H}(\mathbf{x}, \mathbf{h}; \mathbf{x}', \mathbf{h}') = \sum_{\omega} \sum_{\mathbf{x}_s} \mathbf{G}_s^*(\mathbf{x} + \mathbf{h}, \mathbf{x}_s; \omega) \mathbf{G}_s(\mathbf{x}' + \mathbf{h}', \mathbf{x}_s; \omega) \sum_{\mathbf{x}_r} \mathbf{G}_r^*(\mathbf{x} - \mathbf{h}, \mathbf{x}_r; \omega) \mathbf{G}_r(\mathbf{x}' - \mathbf{h}', \mathbf{x}_r; \omega) \quad (11)$$

where G_s denotes Green's function from the source point, \mathbf{x}_s , to the image point, \mathbf{x} , and G_r the Green's function from the image point to the receiver point, \mathbf{x}_r ; \mathbf{h} is the subsurface-offset; the prime indicates points in the image space in the vicinity of the image point and different subsurface-offsets, and the * stands for the conjugate transpose of the Green's functions.

The main diagonal of the Hessian, which is the Laplacian of the cost function related to the model parameters, contains the autocorrelation of the Green's functions and, generally, carries most of the information about illumination. Sometimes, a good and cheap solution is just to approximate the Hessian by its main diagonal and apply its inverse to the migrated image. However, this procedure does not correct for kinematic errors of the migrated image and, depending on the complexity of the illumination pattern, only the least-squares inverse image may be able to provide reasonable results (Clapp, 2005).

Equation 12 shows the structure of the subsurface-offset domain Hessian used in the examples.

$$\mathbf{H}(\mathbf{x}, \mathbf{h}; \mathbf{h}') = \sum_{\omega} \sum_{\mathbf{x}_s} \mathbf{G}_s^*(\mathbf{x} + \mathbf{h}, \mathbf{x}_s; \omega) \mathbf{G}_s(\mathbf{x} + \mathbf{h}', \mathbf{x}_s; \omega) \sum_{\mathbf{x}_r} \mathbf{G}_r^*(\mathbf{x} - \mathbf{h}, \mathbf{x}_r; \omega) \mathbf{G}_r(\mathbf{x} - \mathbf{h}', \mathbf{x}_r; \omega) \quad (12)$$

In this case, the diagonals just represent the cross-correlation between Green's functions computed for a specific image point but shifted by different subsurface-offsets. The off-diagonal terms of the subsurface-offset Hessian are an expression of how much illumination for a specific subsurface-offset is conditioned by the illumination in another subsurface-offset. For the ideal case of infinite cable length, infinite frequency bandwidth and constant velocity, the subsurface-offset Hessian is the identity operator meaning that subsurface-offsets are linearly independent. In this extreme situation, all the energy in a SODCIG will be concentrated at zero-subsurface offset. Therefore, in general, the main diagonal represents an estimate of how much illumination for a specific subsurface-offset is not conditioned by the illumination in another subsurface-offset. Consequently, it is straightforward to consider the main diagonal as the natural candidate to be transformed to angle domain.

Although, in principle, any diagonal of the subsurface-offset Hessian can be transformed to the angle domain by the proposed approach, at present I have conceived a direct application only for the transformed main diagonal, which is to use it as a weight to balance

the amplitudes of the ADCIGs. In the next section I show examples of the angle-domain transformed subsurface-offset Hessian diagonals, as well as the comparison of migrated images before and after the amplitude compensation with the transformed main diagonal, for a small portion of the Sigsbee dataset.

EXAMPLES

To test this methodology I applied it on the well known Sigsbee synthetic dataset. This dataset presents illumination problems due to an irregular salt body shape, which results in unbalanced amplitude patterns in the seismic section (Figure 1). The small rectangle in Figure 1 highlights the target area. The off-end acquisition geometry consists of 348 receivers, 75 ft apart, resulting in 26025 ft maximum offset. As source coordinates are smaller than receiver coordinates, the source-receiver offsets are positive. Therefore, the energy is mainly distributed at positive reflection angles. Figures 2 and 3 show two different

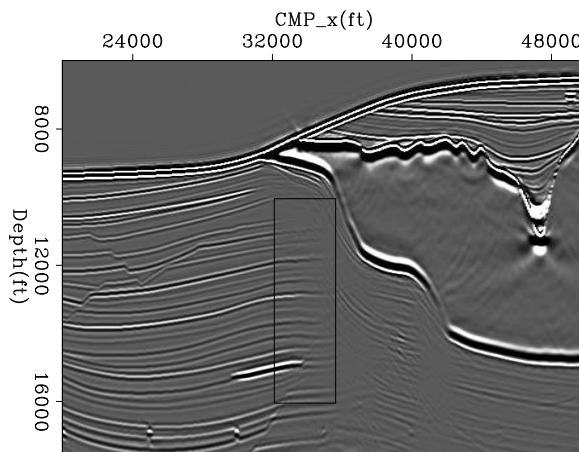


Figure 1: Shot profile migration of part of Sigsbee dataset — zero-subsurface offset. The small box highlights the target area. [claudio1/. Sigsbee](#)

SODCIGs and their respective subsurface-offset Hessian main diagonal, located at CMP coordinates 33200 ft and 35500 ft, respectively. The illumination problem gets more severe as we approach the dipping salt flank. In this work, all figures related to illumination show high-illumination values in dark gray and low illumination values in light gray. Both SODCIGs clearly exhibit the effects of poor illumination represented by horizontal and dipping ($\sim 40^\circ$ — 50°) straight events. The energy smeared along these directions will be mapped to the reflection-angle domain according the dips observed in the subsurface-offset domain. In the SODCIG at CMP position 35500 ft, events curving upward are the expression of multiples.

Transformation of the Main diagonal

Figures 4 and 5 show that, for ADCIGs at CMP coordinates 33200 ft and 35500 ft, the energy is mainly focused at reflection angles around 0° and 40° — 50° . Additionally, these figures show the original reflection-angle gather (a), the main diagonal of the Hessian transformed to the reflection-angle domain (b), and the amplitude-compensated ADCIG (c).

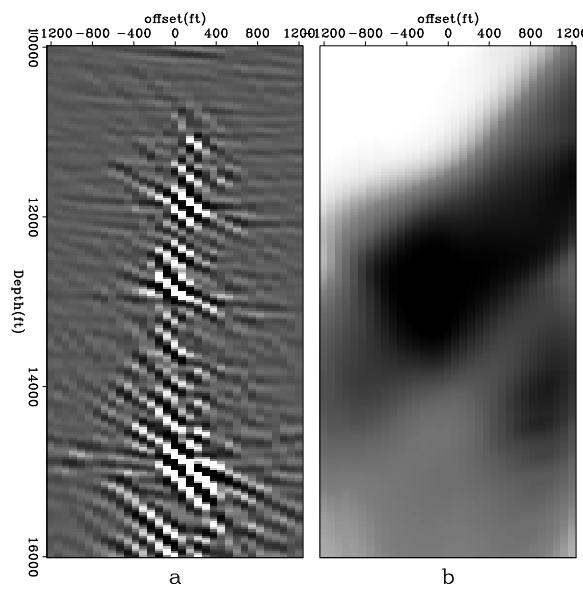


Figure 2: SODCIG and diagonal of the subsurface-offset Hessian at CMP coordinate 33200 ft. Note the effects of poor illumination represented by horizontal and dipping straight events in the SODCIG. `claudio1/. Ojoin16`

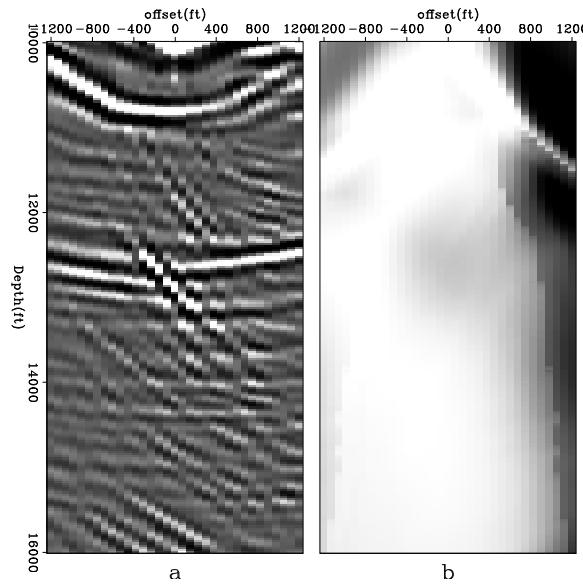


Figure 3: SODCIG and diagonal of the subsurface-offset Hessian at CMP coordinate 33200 ft. Upward curved events correspond to multiples. Note the effects of poor illumination represented by horizontal and dipping straight events in the SODCIG. `claudio1/. Ojoin50`

Notice how the amplitudes are better distributed along the reflection-angle axis after compensation by the inverse of the diagonal of the Hessian. However, as only the diagonal of the Hessian is being used, the kinematic artifacts remain.

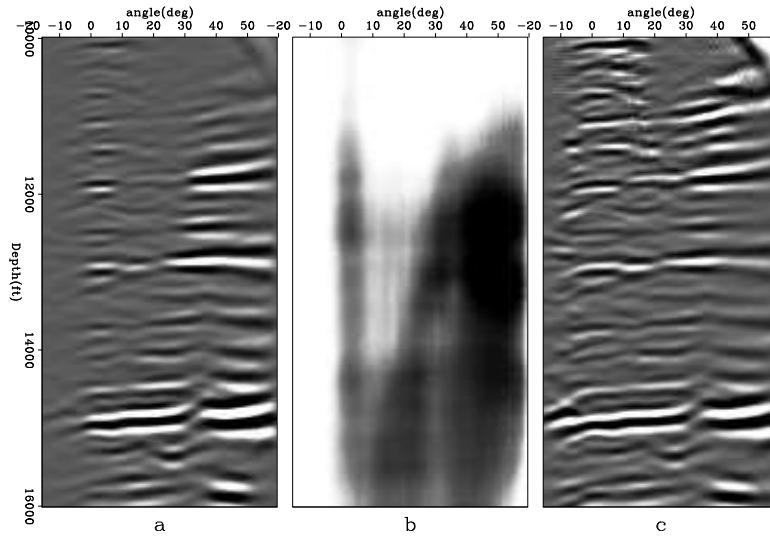


Figure 4: ADCIGs and diagonal of the transformed angle-domain Hessian at CMP coordinate 33200 ft. Before amplitude compensation (a); diagonal of the Hessian in the angle domain (b); and after amplitude compensation (c). Note the improved amplitude balance in the angle direction. claudio1/.join16

The proposed approach seems to be dependent on the amplitude strength of the events in the ADCIGs. However, as shown in the next example, it yields useful information about illumination. Figures 6, 7 and 8 show angle sections of the original angle data (a), the main diagonal of the Hessian in the angle domain (b) and the amplitude-balanced angle data (c). Again, the amplitude compensation proved to be effective. However, notice how for the zero-angle section the illumination computed in the angle domain is low at the right part of the section, in spite of the high amplitudes of the internal multiples. This confirms, to some extent, that the proposed approach can yield reliable information about illumination despite the presence of high-amplitude events not predicted in the computation of the Green's functions.

Figure 9 shows the stacked section along the angle axis, before (a) and after (b) the amplitude compensation by the inverse of the diagonal of the Hessian in the angle domain. The dimming of the amplitudes at the right portion of the section is almost eliminated. Unfortunately, however, the amplitudes of internal multiples are also increased.

Transformation of the off-diagonal terms

In the final example, I show, for the zero-angle section, off-diagonal terms after the transformation to the angle domain. As previously mentioned, it is still not clear the way this results can be used to correct for illumination problems. According to the initial interpretation of the results shown in Figure 10 it is clear that off-diagonal terms start gaining importance as we approach the flank of the salt body at the right of the section. All the

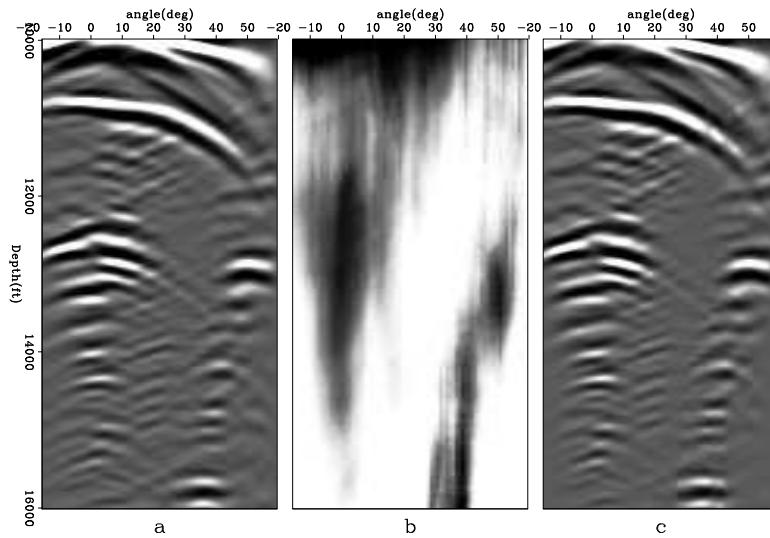


Figure 5: ADCIGs and diagonal of the transformed angle-domain Hessian at CMP coordinate 35500 ft. Before amplitude compensation (a); diagonal of the Hessian in the angle domain (b); and after amplitude compensation (c). `claudio1/.join50`

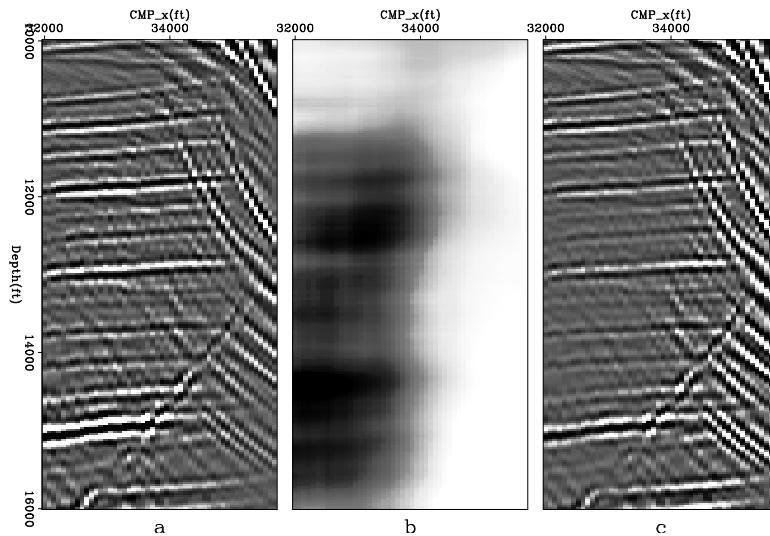


Figure 6: Zero-angle section. Before amplitude compensation (a); diagonal of the Hessian in the angle domain (b); and after amplitude compensation (c). Notice the low illumination in the right part of the section, near the flank of the salt body. `claudio1/.join00`

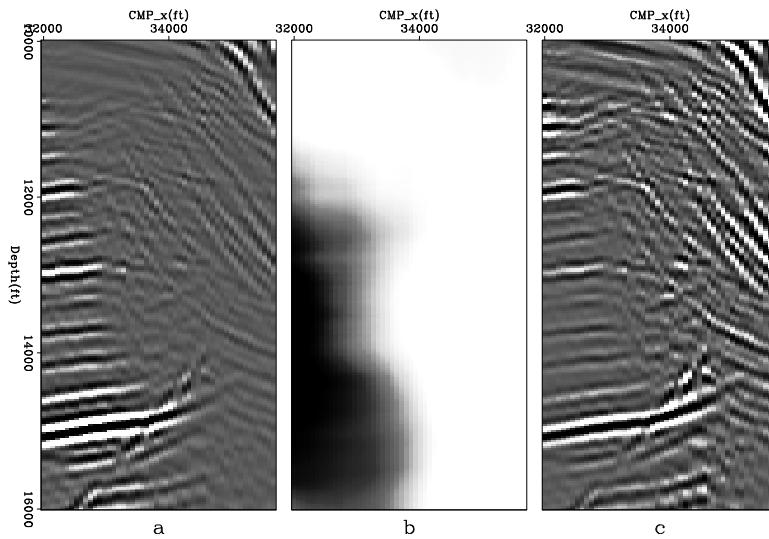


Figure 7: 15° -angle section. Before amplitude compensation (a); diagonal of the Hessian in the angle domain (b); and after amplitude compensation (c). Notice the improved amplitude balance along the CMP direction. `claudio1/. join15`

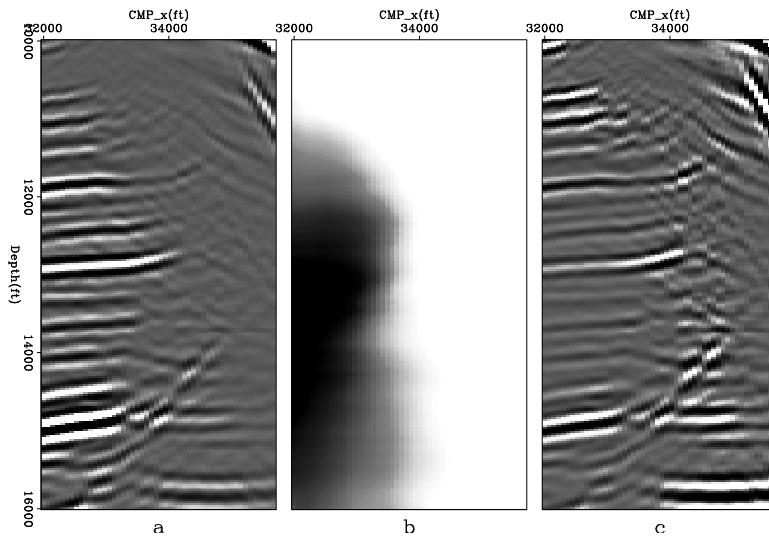


Figure 8: 30° -angle section. Before amplitude compensation (a); diagonal of the Hessian in the angle domain (b); and after amplitude compensation (c). `claudio1/. join30`

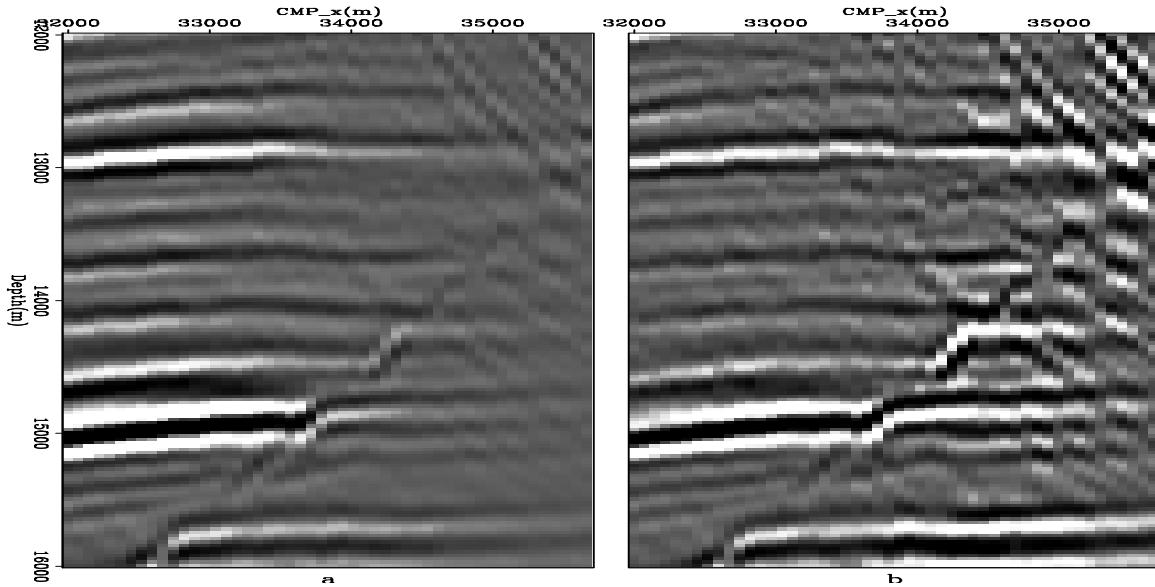


Figure 9: Stack along the angle axis. Before amplitude compensation (a) and after (b).
 claudio1/. Stk

transformed diagonals were scaled to the same value. Actually, the rms value of the main diagonal is approximately 20 and 180 times higher than the rms value of the 5th and 15th off-diagonal, respectively.

CONCLUSIONS

I showed how to estimate angle-domain parameters from the subsurface-offset domain using what I call weighted offset-to-angle, particularly subsurface-offset Hessian diagonals. The proposed approach provides useful information, which can be confirmed by the amplitude compensation results. The transformation of off-diagonal terms of the subsurface-offset Hessian indicates that the results are not strongly dependent on the amplitude distribution in the ADCIGs. However, it is still not clear how to use these transformed off-diagonal terms in inversion schemes.

ACKNOWLEDGEMENTS

I would like to thank Alejandro Valenciano for providing me the subsurface-offset domain Hessian and common-image gathers and for the explanations and fruitful discussions.

REFERENCES

- Bleistein, N., 1984, Mathematical methods for wave phenomena: Academic Press Inc.
- , 1987, On the imaging of reflectors in the earth: Geophysics, **52**, 931–942.
- Clapp, M. L., 2005, Imaging under salt: illumination compensation by regularized inversion: PhD thesis, Stanford University.

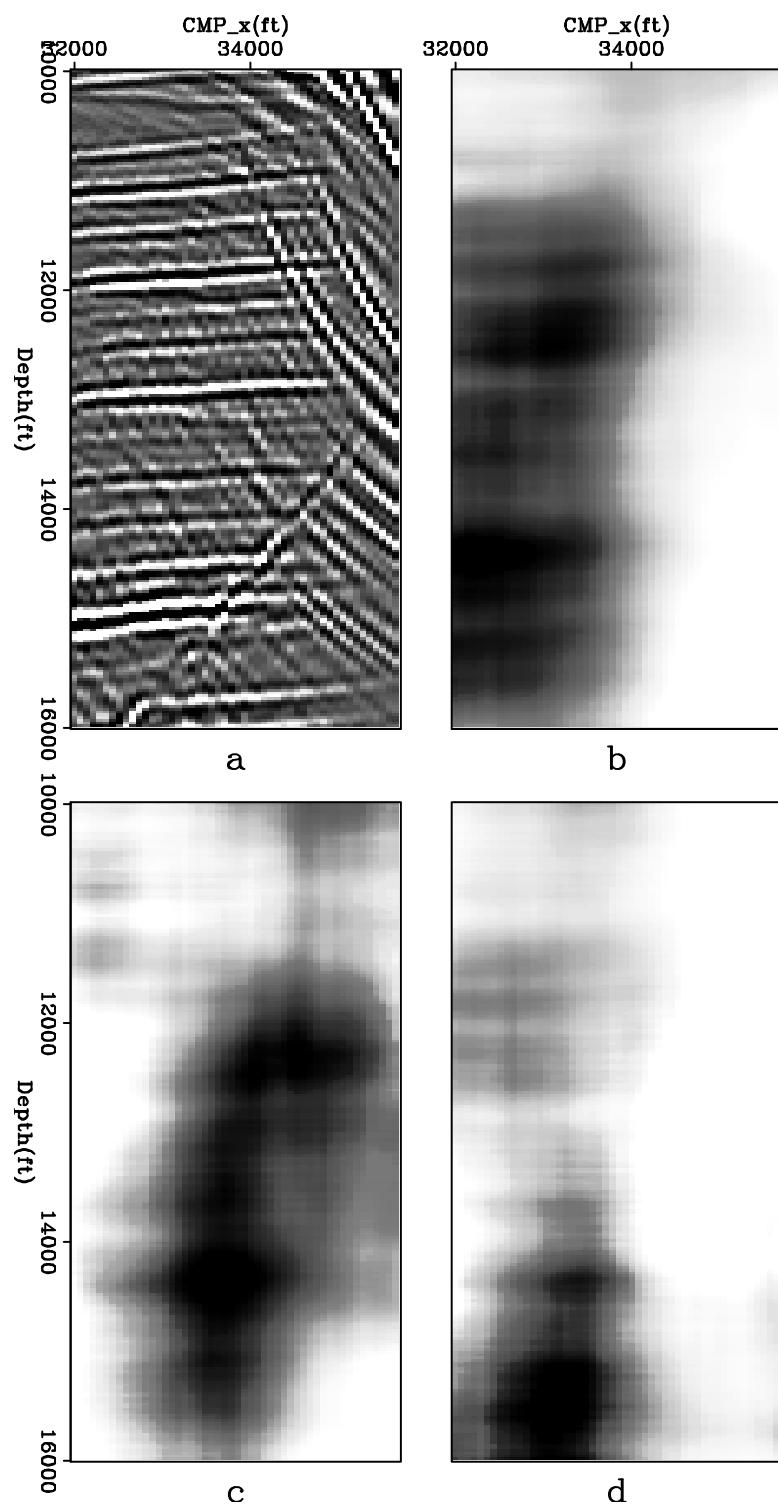


Figure 10: Zero-angle section (a); main diagonal (b); 5th subsurface-offset domain off-diagonal transformed to angle (c); and 15th subsurface-offset domain off-diagonal transformed to angle (d). claudio1/.jDoff01

- Fomel, S., 2003, Angle-domain seismic imaging and the oriented wave equation: 73rd Ann. Internat. Mtg, Soc. Expl. Geophys., Expanded Abstracts, 893–896.
- Prucha, M. L., B. L. Biondi, and W. W. Symes, 1999, Angle-domain common image gathers by wave-equation migration: SEP-Report, **100**, 101–112.
- Prucha, M. L., R. G. Clapp, and B. Biondi, 2000, Seismic image regularization in the reflection angle domain: SEP-Report, **103**, 109–119.
- Sava, P. and S. Fomel, 2000, Angle-gathers by Fourier Transform: SEP-Report, **103**, 119–130.
- Tang, Y., 2007, Selective stacking in the reflection-angle and azimuth domain: SEP-Report, **129**, 159–178.
- Tygel, M., J. Schleicher, P. Hubral, and C. Hanitzsch, 1993, Multiple weights in diffraction stack migration: Geophysics, **59**, 1820–1830.
- Valenciano, A. A., 2006, Target-oriented wave-equation inversion with regularization in the subsurface offset domain: SEP-Report, **124**, 85–94.
- , 2007, Target-oriented wave-equation inversion: regularization in the reflection angle.: SEP-Report, **131**.
- Valenciano, A. A. and B. Biondi, 2005, Wave-equation angle-domain hessian: SEP-Report, **123**, 75–82.
- , 2006, Wave-equation inversion prestack hessian: SEP-125, 201–209.

Ignoring density in waveform inversion

Roland Gunther and Biondo Biondi¹

ABSTRACT

We study the effectiveness of velocity-only time-domain waveform inversion for inverting synthetic data modeled with both velocity and density contrasts. We present a detailed review of the Born approximation for the constant-density acoustic wave equation and its application to the inversion of velocity models for seismic reflection data. We create synthetic models with both constant and variable density and compare the effectiveness of velocity-only waveform inversion in each case. Results from this simple test suggest that density contrasts can hamper the reconstruction of velocity perturbations.

INTRODUCTION

Velocity models for processing seismic reflection data are usually derived from travelttime tomography or other methods that depend on detection of moveout in picked reflection events. Picking is time consuming and prone to error and makes use of only a subset of the information available in a dataset. Waveform inversion provides an alternative approach for deriving velocity models. As an automatic algorithm, waveform inversion is less dependent on human input. The goal of the inversion is to match both data phase and amplitude, so it is theoretically possible to recover subtle local variations that are too small to lead to measurable moveouts in gathers.

Though the method is conceptually appealing, several barriers have prevented waveform inversion from becoming viable for real data: it is only able to recover anomalies that are either very small in magnitude or that have similar spatial wavelengths as the seismic data; it is computationally expensive, especially if based on time-domain modeling; and when the physics of wave-propagation is simplified to reduce cost and complexity, the inversion may not converge to a useful solution. In this report we investigate the last issue.

Early formulations (Lailly, 1984; Tarantola, 1984) describe the method as simultaneous inversions for the source function, the density field, and the bulk modulus field. Woodward (1990) and Luo and Schuster (1991) choose to invert only for a velocity field. Due to the limited geometries of seismic reflection surveys, there is an ambiguity between velocity and density: a velocity anomaly, a density anomaly, or a combination of the two can all create reflections, and near-vertical-incidence waves do not contain much information to distinguish between these cases.

Here we first present a simplified formulation of waveform inversion, based on Tarantola (1984), and show the results of inverting a small synthetic dataset modeled with the same physics—the constant-density acoustic wave equation—as used in the inversion. Then we

¹e-mail: rgunther@stanford.edu

show results from a velocity-only inversion of a dataset modeled with both velocity and density contrasts.

REVIEW OF WAVEFORM INVERSION

Our implementation is based on the constant-density acoustic wave equation

$$\left(\nabla^2 - \frac{1}{v(\vec{r})^2} \frac{\partial^2}{\partial t^2} \right) \Psi(\vec{r}, t) = 0, \quad (1)$$

where ψ is a pressure-field solution, \vec{r} are the model coordinates (x and z for the two-dimensional case), and t is time. Though the implementation uses time-domain finite differences, it is more convenient to express the equation in frequency ω and slowness σ :

$$\left(\nabla^2 + \omega^2 \sigma(\vec{r})^2 \right) \Psi(\vec{r}, t) = 0. \quad (2)$$

The Born approximation

Given a solution $\Psi(\vec{r}, t)$ to equation (2), is it possible to recover $\sigma(\vec{r})$? The Born approximation, named after physicist Max Born, was first developed for scattering theory in quantum mechanics. Applied to seismology, the first-order approximation provides a linear, and thus invertible, relationship between a small change in the slowness model and a resulting small change in the wavefield. We split the model into a background slowness $\sigma_0(\vec{r})$ and a small slowness perturbation $\Delta\sigma(\vec{r})$, where

$$\sigma(\vec{r}) = \sigma_0(\vec{r}) + \Delta\sigma(\vec{r}). \quad (3)$$

The wavefield depends on slowness squared, so here we bring in the first approximation, which is not yet the Born approximation:

$$\sigma(\vec{r})^2 \approx \sigma_0(\vec{r})^2 + 2\sigma_0(\vec{r})\Delta\sigma(\vec{r}). \quad (4)$$

To achieve an approximate relation linear with $\Delta\sigma$, we first substitute (4) into the wave equation:

$$\left(\nabla^2 + \omega^2 \sigma(\vec{r})^2 \right) \Psi(\vec{r}, t) \approx \left(\nabla^2 + \omega^2 \sigma_0(\vec{r})^2 + 2\omega^2 \sigma_0(\vec{r})\Delta\sigma(\vec{r}) \right) \Psi(\vec{r}, t). \quad (5)$$

This approximation is then divided into halves, with only one side depending on $\Delta\sigma$:

$$(\nabla^2 + \omega^2 \sigma_0(\vec{r})^2) \Psi(\vec{r}, t) \approx -2\omega^2 \sigma_0(\vec{r}) \Delta\sigma(\vec{r}) \Psi(\vec{r}, t). \quad (6)$$

Much as the slowness field was split into two parts, the wavefield now is divided into a background wavefield Ψ_0 and a scattered wavefield $\Delta\Psi$ such that

$$\Psi(\vec{r}, \omega) = \Psi_0(\vec{r}, \omega) + \Delta\Psi(\vec{r}, \omega), \quad (7)$$

where, by definition, Ψ_0 is the solution for the background wavefield, or

$$(\nabla^2 + \omega^2 \sigma_0(\vec{r})^2) \Psi_0(\vec{r}, \omega) = 0. \quad (8)$$

Substituting the divided wavefield (7) into the approximate wave equation (6), and using the fact that the background wavefield is an exact solution for the background velocity, we can write

$$\left(\nabla^2 + \omega^2\sigma_0(\vec{r})^2\right)\Delta\Psi(\vec{r}, \omega) \approx -2\omega^2\sigma_0(\vec{r})\Delta\sigma(\vec{r})\Psi(\vec{r}, t). \quad (9)$$

At this point, we have an implicit relation between a small change $\Delta\sigma$ in the model and the resulting scattered wavefield $\Delta\Psi$. Ideally, we would like to have an explicit expression for $\Delta\sigma$ as a function of the background and scattered wavefields. Such an expression cannot be written directly; instead, we can find an expression for $\Delta\Psi$ as a function of $\Delta\sigma$. This expression is an integral over potential scatterers convolved with the Green's function $G_0(\vec{r}, \omega; \vec{r}')$, the response at point \vec{r}' and frequency ω for a point source at point \vec{r} . The subscript indicates that the Green's function is defined for the background wavefield. We build up the integral expression by starting with the formal definition of the Green's function, which is the solution of the wave equation with a delta-function source:

$$\left(\nabla^2 + \omega^2\sigma_0(\vec{r})^2\right)G_0(\vec{r}, \omega; \vec{r}') = \delta(\vec{r} - \vec{r}'). \quad (10)$$

Both sides of this definition are multiplied by $-2\omega^2\sigma_0\Delta\sigma\Psi$ and integrated with respect to \vec{r}' :

$$-\int d\vec{r}' 2\omega^2\sigma_0(\vec{r}')\Delta\sigma(\vec{r}')\Psi(\vec{r}', \omega) \left[(\nabla^2 + \omega^2\sigma_0(\vec{r})^2)G_0(\vec{r}, \omega; \vec{r}') = \delta(\vec{r} - \vec{r}') \right]. \quad (11)$$

The Laplacian operator is taken with respect to \vec{r} , not \vec{r}' , so the left side of the expression can be simplified by moving the integral inside the operator; on the right side, the delta function sifts the original function out of the integral, leaving

$$\left(\nabla^2 + \omega^2\sigma_0(\vec{r})^2\right) - \int d\vec{r}' 2\omega^2\sigma_0(\vec{r}')\Delta\sigma(\vec{r}')G_0(\vec{r}, \omega; \vec{r}')\Psi(\vec{r}', \omega) = -2\omega^2\sigma_0(\vec{r})\Delta\sigma(\vec{r})\Psi(\vec{r}, \omega). \quad (12)$$

Comparing with (9), we can see that the integral represents a solution for $\Delta\Psi$, allowing us to write

$$\Delta\Psi(\vec{r}, \omega) \approx -\int d\vec{r}' 2\omega^2\sigma_0(\vec{r}')\Delta\sigma(\vec{r}')G_0(\vec{r}, \omega; \vec{r}')\Psi(\vec{r}', \omega). \quad (13)$$

Unfortunately, the scattered wavefield is still a function of the entire—and unknown—wavefield Ψ . The first-order Born approximation asserts that when the scattered wavefield is small compared to the background wavefield, the interaction between scattering points can be ignored. This is equivalent to replacing Ψ with Ψ_0 on the right-hand side, leaving

$$\Delta\Psi(\vec{r}, \omega) \approx -\int d\vec{r}' 2\omega^2\sigma_0(\vec{r}')\Delta\sigma(\vec{r}')G_0(\vec{r}, \omega; \vec{r}')\Psi_0(\vec{r}', \omega). \quad (14)$$

This approximation now provides a linear relationship between a small change in the model and the resulting small wavefield perturbation.

Application to seismic inversion

Application of the Born approximation as expressed in (14) requires knowledge of the residual wavefield everywhere in the image space. Unfortunately, the full wavefield, and thus the residual wavefield, is only known at the receivers. For simplicity we assume that receivers are located at all x-locations on the surface, or that the receiver wavefield is unaliased and

can be perfectly recovered. We define $w_n(\vec{r}, \omega, s)$ as the background wavefield for shot s at the n^{th} iteration. This wavefield is computed by forward modeling the shot field through the n^{th} slowness model. The data residual $\Delta d_n(x, \omega, s)$ is computed by selecting the background wavefield at $z = 0$ and subtracting from the recorded data $d(x, \omega, s)$, or

$$\Delta d_n(x, \omega, s) = d(x, \omega, s) - |w_n(\vec{r}, \omega, s)|_{z=0}. \quad (15)$$

The objective of the inversion is to minimize the l^2 norm of Δd .

Our implementation uses the linear forward operator to compute a step length at each iteration. Substituting w_n for the background wavefield and selecting only the scattered field at the receivers, the frequency domain expression for the linear forward operator becomes

$$\Delta d_n(x, \omega, s) = \left| - \int d\vec{r}' 2\omega^2 \sigma_0(\vec{r}') \Delta \sigma(\vec{r}') G_n(\vec{r}, \omega; \vec{r}') w_n(\vec{r}', \omega, s) \right|_{z=0}. \quad (16)$$

Since we use time-domain finite-difference modeling, it is useful to express the operator in the time domain. The $-\omega^2$ factor is applied as a second time derivative to the w_n wavefield and the multiplication of w_n , and G_n becomes a convolution along the time dimension, yielding

$$\Delta d_n(x, t, s) = \left| \int d\vec{r}' 2\sigma_0(\vec{r}') \Delta \sigma(\vec{r}') G_n(\vec{r}, t; \vec{r}') * \ddot{w}_n(\vec{r}', t, s) \right|_{z=0}. \quad (17)$$

The forward operator is implemented in two steps. First, the background wavefield w_n is computed by propagating the source field forward in time. Next, the background wavefield is scaled by $-2\sigma_n \Delta \sigma$ and used as a new source field that is also propagated forward in time.

The gradient direction $\Delta \sigma$ for each step of the inversion is computed using the adjoint of the forward operator. The independent variables used in the forward operator are \vec{r} , \vec{r}' , ω , and s . The forward operator integrates over \vec{r}' and selects data at $z = 0$, so the adjoint is expressed by integrating over the remaining variables and injecting data, expressed here as multiplying with a delta function, at $z = 0$:

$$\Delta \sigma_n(\vec{r}') = - \iiint ds d\omega d\vec{r} 2\omega^2 \sigma_n(\vec{r}') w_n^*(\vec{r}', \omega, s) G_n^*(\vec{r}, \omega; \vec{r}') \delta(z) \Delta d_n(x, \omega, s) \quad (18)$$

This integral represents reverse-time migration of the data residual. We show a simplified expression by defining a new wavefield res_n that represents the propagation of the data residual. The time axis of the Green's function is reversed due to the complex conjugate in the frequency domain:

$$res_n(\vec{r}', t, s) = \int d\vec{r} G_n(\vec{r}, -t; \vec{r}') * \delta(z) \Delta d_n(x, t, s). \quad (19)$$

In practice, the integral is computed by forward propagating the time-reversed data residual. Due to Green's function reciprocity, integration over \vec{r} is equivalent to the integration over \vec{r}' in (17). The wavefield res_n is then substituted into the time-domain expression for the adjoint operator where integration over frequencies is exchanged for integration over time, and the time axis of the background wavefield is reversed:

$$\Delta \sigma_n(\vec{r}') = 2\sigma_n(\vec{r}') \iint ds dt \ddot{w}_n(\vec{r}', -t, s) \cdot res_n(\vec{r}', t, s). \quad (20)$$

With both the forward and adjoint linear seismic modeling expressions defined, we have all of the building blocks needed to invert for σ . We use a non-linear variation of conjugate

gradients following Claerbout (2004). The method differs from linear conjugate gradients in that for each iteration the operators, which depend on w_n , change and the data residual Δd_n is recomputed.

APPLICATION TO MODELS WITH AND WITHOUT DENSITY

To test effects of density on waveform inversion, we construct two earth models, one with reflectors simulated by velocity spikes and one with reflectors simulated by density spikes. We add a Gaussian anomaly to both velocity models to test whether our constant-density implementation of waveform inversion can recover long-wavelength velocity perturbations.

Figure 1 shows the slowness field (a) for the constant-density model. Ten horizontal stripes with +1% change in slowness act as reflectors that generate events in the data. Though the goal of waveform inversion is to invert long-wavelength velocity perturbations, the inversion also needs to recover high-frequency perturbations in order to match the data. We also add a +1% Gaussian anomaly to the model. We provide the inversion with a constant-slowness initial model (b) that matches the background velocity of the actual model. After 185 iterations, the inversion (c) recovers both the reflectors and the Gaussian anomaly. Vertical slices through the middle of the slowness model and inversion result (d) show that the inversion comes close to correctly estimating the magnitude of the slowness spikes, especially in the region unaffected by the anomaly. The inversion under-estimates the magnitude of the slowness anomaly, and it smears the anomaly vertically. These two effects tend to counteract each other since a slowness perturbation with a large spatial extent but small magnitude can introduce similar delays as a spatially small perturbation with a large magnitude.

Figure 2 shows the slowness field (a) and the density field (b) for the variable-density model. In this example we introduce the same +1% Gaussian anomaly to the velocity model, but we simulate reflectors with density contrasts instead of slowness contrasts. Since slowness and density changes create reflections of opposite polarity, we add -1% spikes to the density model. After 300 iterations, the inversion introduces horizontal stripes into the velocity model to account for the density reflectors and partially recovers the Gaussian anomaly (c). Overall, the inversion result is noisier, and the norm of the data residual is larger than for the previous example. Vertical slices through the model and inversion result (d) show that the inversion introduces negative changes at the top and bottom of the anomaly, which is entirely positive. This example illustrates that data effects due to density can inhibit the ability of waveform inversion to recover velocity anomalies.

CONCLUSION

As with travel-time based velocity analysis methods, the primary purpose for waveform inversion is to find a velocity model for imaging. However, waveform inversion needs to match data phase and amplitude, not just travel-times, and the examples in this report show that inverting for just velocity can be dangerous when density variations are significant. A possible solution is the joint inversion of density and velocity as proposed by Tarantola (1984). Further research is needed to determine whether seismic data typically contain the information necessary to constrain both fields.

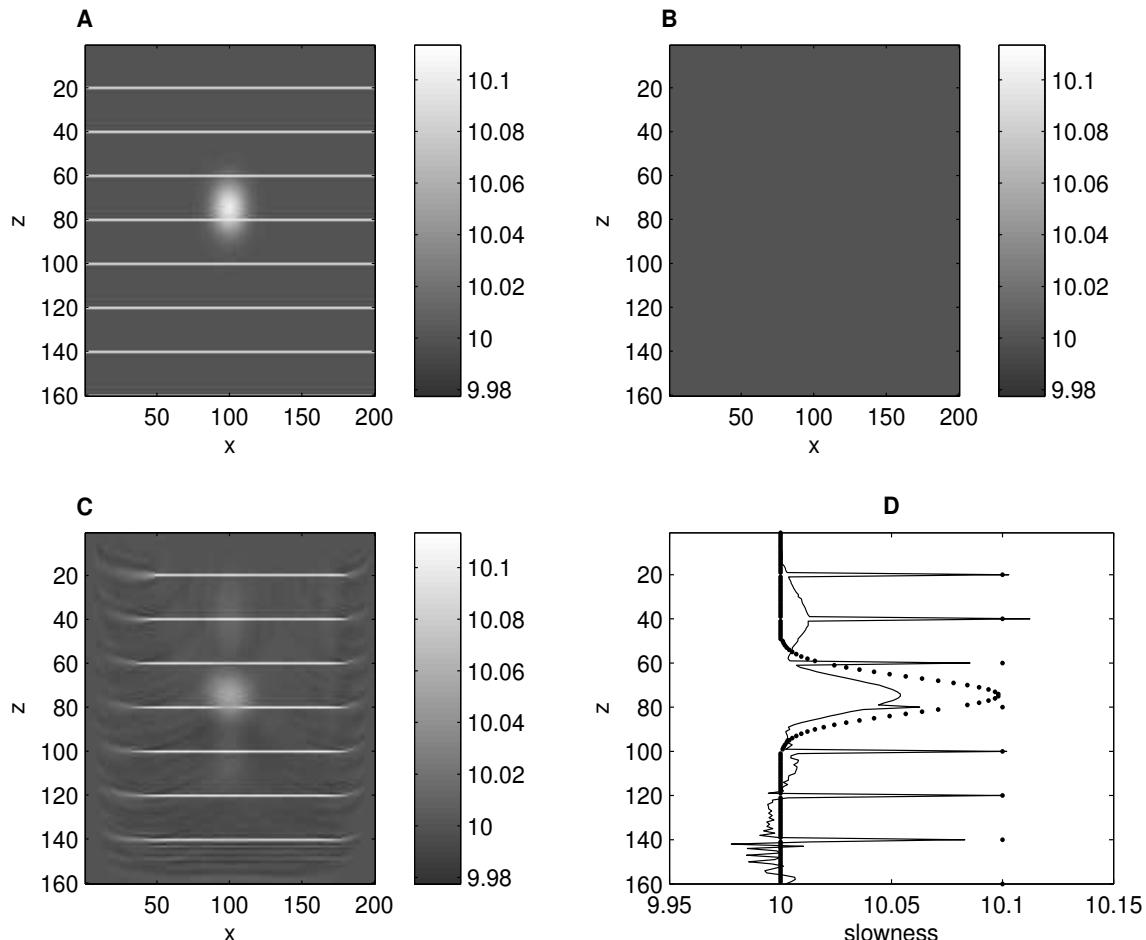


Figure 1: Numerical experiment for a constant-density earth: (a) slowness model used to compute the data; (b) starting slowness model; (c) inversion result after 185 iterations; and (d) slices through the model (dotted line) and inversion result (solid line). rgunther1/.fig-vel

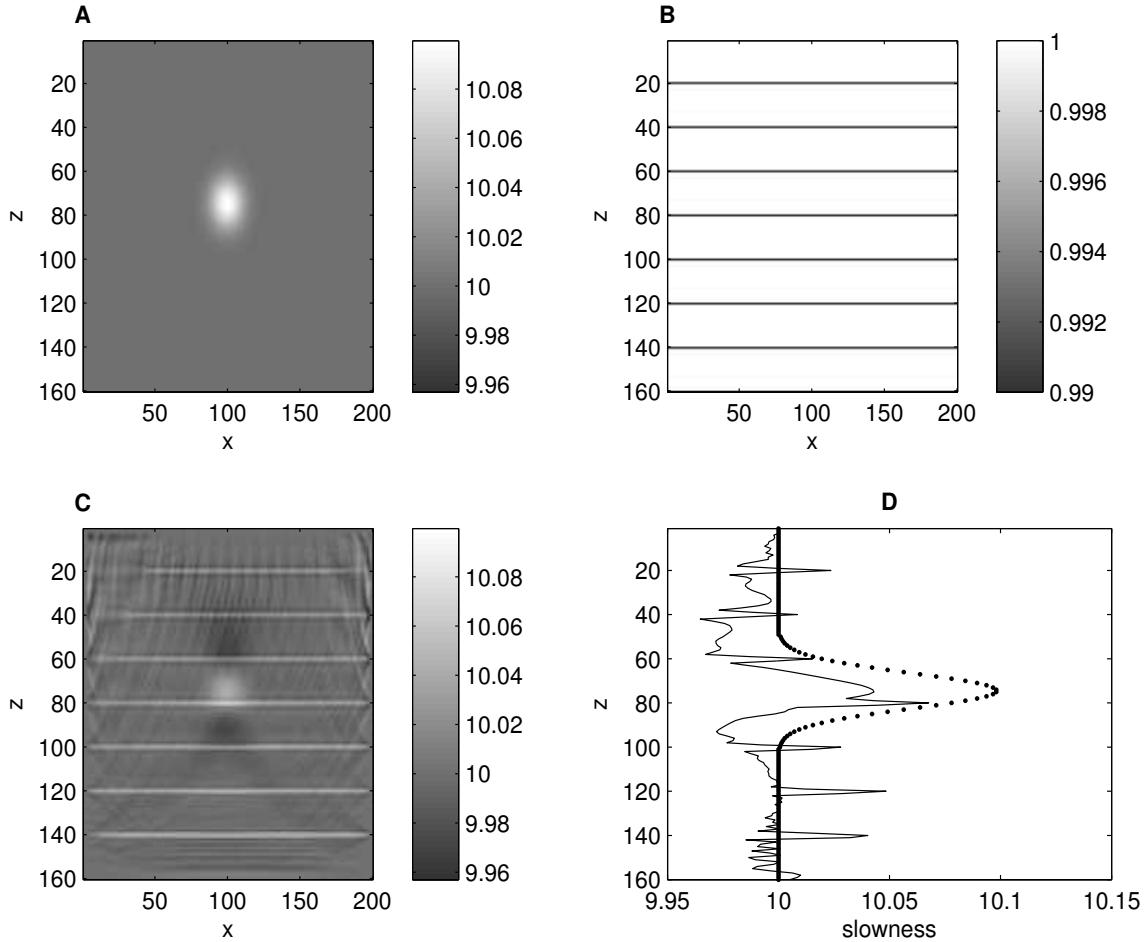


Figure 2: Numerical experiment for a variable-density earth. Data is modeled for a constant-background slowness field (a) containing a Gaussian anomaly but no reflectors. The reflectors are instead embedded in the density field (b). After 300 iterations, the inversion (c) attempts to fit the reflection with velocity spikes but still manages to recover some of the anomaly. (d) Slices through the model (dotted line) and inversion result (solid line) show that the anomaly is recovered less effectively than in the previous example.

```
rgunther1/.fig-den
```

REFERENCES

- Claerbout, J., 2004, Image estimation by example: Geophysical soundings image construction: Stanford Exploration Project.
- Lailly, P., 1984, The seismic inversion problem as a sequence of before stack migrations: Conference on Inverse Scattering: Philadelphia, SIAM, **49**, 1259–1266.
- Luo, Y. and G. T. Schuster, 1991, Wave-equation travel-time inversion: *Geophysics*, **56**, 645–653.
- Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: *Geophysics*, **49**, 1259–1266.
- Woodward, M., 1990, Wave equation tomography: **62**.

Exact seismic velocities for TI media and extended Thomsen formulas for stronger anisotropies

James G. Berryman¹

ABSTRACT

I explore a different type of approximation to the exact anisotropic wave velocities as a function of incidence angle in transversely isotropic (TI) media. This formulation extends the Thomsen weak anisotropy approach to stronger deviations from isotropy without significantly affecting the simplicity of the equations. One easily recognized improvement is that the extreme value of the quasi-SV-wave speed $v_{sv}(\theta)$ is located near the correct incidence angle $\theta = \theta_{ex}$, rather than always being at the position $\theta = 45^\circ$, which universally holds for Thomsen's approximation — although $\theta_{ex} \equiv 45^\circ$ is actually never correct for *any* TI anisotropic medium. Also, the magnitudes of all the wave speeds are typically (although there may be some exceptions depending on the actual angular location of the extreme value) more closely approximated for all values of the incidence angle. Furthermore, the value of a special angle θ_m (which is close to the location of the extreme and also required by the new formulas) can be deduced from the same data that are normally used in the weak anisotropy data analysis. All the main technical results presented are independent of the physical source of the anisotropy. To illustrate the use of the results obtained, two examples are presented based on systems having vertical fractures. The first set of model fractures has their axes of symmetry randomly oriented in the horizontal plane. Such a system is then isotropic in the horizontal plane and, thus, exhibits vertical transverse isotropic (VTI) symmetry. The second set of fractures also has its axes of symmetry in the horizontal plane, but (it is assumed) these axes are aligned so that the system exhibits horizontal transverse isotropic (HTI) symmetry. Both types of systems, as well as any other TI medium (whether due to fractures or layering or other physical causes) are more accurately treated with the new wave speed formulation.

INTRODUCTION

Thomsen's weak anisotropy formulation (Thomsen, 1986) was originally designed for media having vertical transversely isotropic (VTI) symmetry, but clearly applies equally well to any other TI media (for example HTI) with only very minor technical changes related to how the orientation of the axis of symmetry is labelled in Cartesian coordinates. This formulation is also independent of the natural mechanism producing the anisotropy, whether it be due to layering, or horizontal fractures, or randomly oriented vertical fractures, or some other source. So the method has wide applicability for use in exploration problems. However, when the approximate results of the Thomsen's original formulation are compared to known exact results for the same VTI media, it is easy to see that there are some deficiencies. In

¹e-mail: rgunther@stanford.edu

particular, for VTI media, the vertically polarized (SV) shear wave will always have a peak (or possibly a trough, for some fairly rare types of anisotropic media) somewhere in the range $0 \leq \theta \leq \pi/2 = 90^\circ$. Thomsen's weak anisotropy formulation always puts this extreme point (either minimum or maximum) exactly at $\theta = \pi/4 = 45^\circ$. However, as I show here, the $\theta = 45^\circ$ angular location never actually occurs for any interesting degree of VTI anisotropy; instead $\theta \rightarrow 45^\circ$ (by which I mean the extreme point approaches but never reaches 45°) for extremely weak anisotropy — *e.g.*, very low horizontal crack density is one example of this. In an effort to determine whether it might be possible to improve on Thomsen's approximation, I have found that a relatively small modification of Thomsen's formulas places the extreme v_{sv} point at nearly the right angular location, and also typically (though not universally) improves the overall fit of both $v_{sv}(\theta)$ and $v_p(\theta)$ to the exact VTI curves. The ultimate cost of this improvement is negligible since the data required to estimate the location of the extreme point are exactly the same as the data used to determine Thomsen's other parameters for weak anisotropy. The method can also be used with only minor technical modifications for media having horizontal transversely isotropic (HTI) symmetry, such as reservoirs having aligned vertical fractures. The paper focuses on the general theory and uses other recent work relating fracture influence parameters (Sayers and Kachanov, 1991; Berryman and Grechka, 2006) to provide some useful examples of the applicability of the new method. Other choices of the various possible applications of the new method will appear in later publications.

The main result of the paper — from which all the subsequent results follow — is a new, more compact, and more intuitive way of writing the quantity $\zeta(\theta)$ [appearing here in equation 12]. This quantity has its extreme value at almost the same location as that of the quasi-SV-wave phase velocity, and this angular location is very easy to determine.

The following section reviews the standard results for wave speeds in a VTI medium, and also presents the Thomsen weak anisotropy results. The next section presents the analysis leading to the extended (*i.e.*, improving on Thomsen) anisotropy formulation, which allows the wave speed formulas to reflect more accurately the correct behavior near the extremes (greatest excursions from the values at normal incidence and near horizontal incidence). Then, the next section shows how to determine the value of θ_m (the incidence angle that determines where the extreme SV-wave behavior occurs) from the same data already used in Thomsen's formulas. Furthermore, normal moveout corrections are recomputed for the new formulation, and it is found that the results are identical to those for Thomsen formulation; thus, no new corrections are needed near normal incidence. Finally, to illustrate the results, models of VTI and HTI reservoirs having vertical fractures are computed using the new wave speed formulation and compared to prior results. Appendix A computes the quasi-SV-wave speed at $\theta = \theta_m$ exactly, and also at two levels of approximation in order to have values to check against the corresponding results in the main text. Appendix B discusses how to get HTI results simply and directly from VTI results, both for the exact wave speeds and for the new approximate wave speed formulas. The final section of the main text presents an overview and suggests some possible applications of the results.

THOMSEN'S WEAK ANISOTROPY FORMULATION FOR SEISMIC WAVES

Thomsen's weak anisotropy formulation (Thomsen, 1986), being a collection of approximations designed specifically for use in velocity analysis for exploration geophysics, is clearly not exact. Approximations incorporated into the formulas become most apparent for angles θ greater than about 15° from the vertical, especially for compressional and vertically polarized shear wave velocities $v_p(\theta)$ and $v_{sv}(\theta)$, respectively. For VTI media, angle θ is measured from the \hat{z} -vector pointing directly into the earth.

For reference purposes, I include here the exact velocity formulas for; quasi-P, quasi-SV, and SH seismic waves at all angles in a VTI elastic medium. These results are available in many places (Postma, 1955; Musgrave, 1959, 2003; Rüger, 2002; Thomsen, 2002), but were taken directly from Berryman (1979) with only some minor changes of notation; specifically, the a,b,c,f,l,m notation for stiffnesses has been translated to the Voigt c_{ij} stiffness notation wherein $a \rightarrow c_{11}$, $b \rightarrow c_{12}$, $c \rightarrow c_{33}$, $f \rightarrow c_{13}$, $l \rightarrow c_{44}$, and $m \rightarrow c_{66}$. The results are:

$$v_p^2(\theta) = \frac{1}{2\rho} \left\{ \left[(c_{11} + c_{44}) \sin^2 \theta + (c_{33} + c_{44}) \cos^2 \theta \right] + R(\theta) \right\} \quad (1)$$

and

$$v_{sv}^2(\theta) = \frac{1}{2\rho} \left\{ \left[(c_{11} + c_{44}) \sin^2 \theta + (c_{33} + c_{44}) \cos^2 \theta \right] - R(\theta) \right\}, \quad (2)$$

where

$$R(\theta) = \sqrt{[(c_{11} - c_{44}) \sin^2 \theta - (c_{33} - c_{44}) \cos^2 \theta]^2 + 4(c_{13} + c_{44})^2 \sin^2 \theta \cos^2 \theta} \quad (3)$$

and, finally,

$$v_{sh}^2(\theta) = \frac{1}{\rho} \left[c_{44} + (c_{66} - c_{44}) \sin^2 \theta \right]. \quad (4)$$

I have purposely written equations 1 and 2 in this way to emphasize the fact that $v_p^2(\theta)$ and $v_{sv}^2(\theta)$ are closely related since they are actually the two solutions of a quadratic equation having the form:

$$(v^2)^2 - (v_p^2 + v_{sv}^2)v^2 + v_p^2 v_{sv}^2 = 0. \quad (5)$$

Any approximations made to one of these two wave speeds should therefore always be reflected in the other for this reason. In particular, any approximation to the square root in R should be made consistently for both v_p and v_{sv} .

For VTI symmetry, the stiffness matrix c_{ij} is defined for $i,j = 1, \dots, 6$ by

$$c_{ij} = \begin{pmatrix} c_{11} & c_{12} & c_{13} & & & \\ c_{12} & c_{11} & c_{13} & & & \\ c_{13} & c_{13} & c_{33} & & & \\ & & & c_{44} & & \\ & & & & c_{44} & \\ & & & & & c_{66} \end{pmatrix}, \quad (6)$$

where $c_{12} = c_{11} - 2c_{66}$. In an isotropic system (which is a more restrictive case than our current interests), $c_{12} = c_{13} = \lambda$, $c_{44} = c_{66} = \mu$, and $c_{11} = c_{33} = \lambda + 2\mu$, where λ and μ are the usual Lamé constants. The definition in equation 6 makes use of the Voigt notation,

i.e., 6×6 matrix instead of 4th order tensor, wherein Voigt single indices $i, j = 1, 2, 3, 4, 5, 6$ correspond to the pairs of tensor indices $11, 22, 33, 23, 31, 12$, respectively. And it relates stress σ_{ij} to strain ϵ_{ij} via $\sigma_{23} = c_{44}\epsilon_{23}$, $\sigma_{31} = c_{44}\epsilon_{31}$, $\sigma_{12} = c_{66}\epsilon_{12}$, and $\sigma_{ii} = \sum_j c_{ij}\epsilon_{jj}$ (no summation over repeated indices is assumed here) for $i, j = 1, 2, 3$. For VTI symmetry, we typically take $x_3 = z$ (the vertical) as the axis of symmetry. But, for HTI symmetry, we may choose index direction x_3 to be some other physical direction (such as horizontal directions x or y , or some linear combination thereof); having done this, equations 2–4 apply strictly only in the vertical plane perpendicular to the fracture plane, while a small amount of vector analysis is then required to obtain the velocity values at all azimuthal angles $\phi \neq \pi/2$ away from the fracture plane.

Expressions for phase velocities in Thomsen's weak anisotropy limit can be found in many places, including Thomsen (1986, 2002) and Rüger (2002). The pertinent expressions for phase velocities in VTI media as a function of angle θ , measured as previously mentioned from the vertical direction, are

$$v_p(\theta) \simeq v_p(0) \left(1 + \epsilon \sin^2 \theta - (\epsilon - \delta) \sin^2 \theta \cos^2 \theta \right), \quad (7)$$

$$v_{sv}(\theta) \simeq v_s(0) \left(1 + \left[v_p^2(0)/v_s^2(0) \right] (\epsilon - \delta) \sin^2 \theta \cos^2 \theta \right), \quad (8)$$

and

$$v_{sh}(\theta) \simeq v_s(0) \left(1 + \gamma \sin^2 \theta \right). \quad (9)$$

In our present context, $v_s(0) = \sqrt{c_{44}/\rho_0}$, and $v_p(0) = \sqrt{c_{33}/\rho_0}$, where c_{33} , c_{44} , and ρ_0 are two stiffnesses of the cracked medium and the mass density of the isotropic host elastic medium. [For the specific physical examples that follow involving models of fractured reservoirs, I assume that the cracks contain insufficient volume to affect the overall mass density significantly.] The three Thomsen (1986) seismic parameters appearing in equations 7–9 for weak anisotropy with VTI symmetry are $\gamma = (c_{66} - c_{44})/2c_{44}$, $\epsilon = (c_{11} - c_{33})/2c_{33}$, and

$$\delta = \frac{(c_{13} + c_{44})^2 - (c_{33} - c_{44})^2}{2c_{33}(c_{33} - c_{44})} = \left(\frac{c_{33} + c_{13}}{2c_{33}} \right) \left(\frac{c_{13} + 2c_{44} - c_{33}}{c_{33} - c_{44}} \right). \quad (10)$$

Parameter γ is a measure of the shear wave anisotropy and birefringence. Parameter ϵ is a measure of the quasi-P wave anisotropy. Parameter δ controls the complexity of the shape of the wave fronts for quasi-P and quasi-SV waves; *e.g.*, when $\delta = \epsilon$ the wave fronts are elliptical in shape, whereas for all TI anisotropic systems having $\epsilon - \delta \neq 0$, the wave front will deviate from being elliptical, and it is in such cases that ray arrival triplications may occur.

All three of these parameters γ , ϵ , δ can play important roles in the velocities given by equations 7–9 when the anisotropy is large, as would be the case in fractured reservoirs when the crack densities are high enough. If crack densities are very low, then the SV shear wave will actually have no dependence on angle of wave propagation. Note that the so-called anellipticity parameter (Dellinger *et al.*, 1993; Fomel, 2004; Tsvankin, 2005, p. 253), $A = \epsilon - \delta$, vanishes when $\epsilon \equiv \delta$ — which (as will be shown) does happen to a very good approximation for low crack densities. Then, the results are anisotropic but have the special (elliptical) shape to the wave front mentioned previously.

For each of these phase velocities, the derivation of Thomsen's approximation has included a step that removes the square on the left-hand side of equations 1, 2, and 4 —

obtained by expanding a square root of the right hand side. This step introduces a factor of $\frac{1}{2}$ multiplying the $\sin^2 \theta$ terms on the right hand side, and — for example — immediately explains how equation 8 is obtained from equation 4. The other two equations for $v_p(\theta)$ and $v_{sv}(\theta)$, *i.e.*, equations 7 and 8, involve additional approximations. More of the details about the nature of these approximations are elucidated by first obtaining an alternative approximate formulation.

EXTENDED APPROXIMATIONS FOR ANISOTROPIC WAVE SPEEDS

The biggest and most obvious problem with Thomsen's approximations to the wave speeds generally occurs in $v_{sv}(\theta)$. The key issue is that Thomsen's approximation for $v_{sv}(\theta)$ is completely symmetric around $\theta = \pi/4 = 45^\circ$, while unfortunately this is generally not true of the actual wave speeds $v_{sv}(\theta)$. This error may seem innocuous in itself since it is not immediately clear whether it affects the results for small angles of incidence ($< 15^\circ$) or not, but it can in fact lead to large over- or under-estimates of wave speeds in the neighborhood of both the extreme value located at $\theta = \theta_{ex}$ and also at $\theta = 45^\circ \neq \theta_{ex}$. To improve this situation while still making use of a practical approximation to the wave speed, I reconsider an approach originally proposed in Berryman (1979). In particular, notice that the square root formula for $R(\theta)$ can be conveniently, and exactly, rewritten as:

$$R(\theta) = [(c_{11} - c_{44}) \sin^2 \theta + (c_{33} - c_{44}) \cos^2 \theta] \sqrt{1 - \zeta(\theta)}, \quad (11)$$

where

$$\zeta(\theta) \equiv 4 \frac{[(c_{11} - c_{44})(c_{33} - c_{44}) - (c_{13} + c_{44})^2] \sin^2 \theta \cos^2 \theta}{[(c_{11} - c_{44}) \sin^2 \theta + (c_{33} - c_{44}) \cos^2 \theta]^2}. \quad (12)$$

To simplify this expression, first notice that ζ has an absolute maximum value, which occurs when θ takes the value θ_m determined by

$$\tan^2 \theta_m = \frac{c_{33} - c_{44}}{c_{11} - c_{44}}. \quad (13)$$

The extreme value of ζ is given by

$$\zeta_m = 1 - \frac{(c_{13} + c_{44})^2}{(c_{11} - c_{44})(c_{33} - c_{44})} = \frac{2(\epsilon - \delta)c_{33}}{c_{11} - c_{44}} = \frac{2(\epsilon - \delta)v_p^2(0)}{v_p^2(0)(1 + 2\epsilon) - v_s^2(0)}, \quad (14)$$

where the second and third expressions relate ζ_m to the difference between the Thomsen parameters ϵ and δ , and to $v_p(0)$ and $v_s(0)$. Then, $\zeta(\theta)$ can be rewritten as

$$\zeta(\theta) = \frac{2\zeta_m}{1 + \chi(\theta)}, \quad (15)$$

where

$$\chi(\theta) = \frac{1}{2} \left[\frac{\tan^2 \theta}{\tan^2 \theta_m} + \frac{\tan^2 \theta_m}{\tan^2 \theta} \right]. \quad (16)$$

For realistic systems, it is always true that $\zeta(\theta) \leq 1$. [For example, in the fractured reservoir examples presented later in the paper, the largest observed value of $\zeta_m \simeq 0.29$.

Also, note $\zeta_m \geq 0$ for all layered media since $\epsilon - \delta \geq 0$ for layered elastic media (Postma, 1955; Backus, 1962; Berryman, 1979).] So, we can expand the square root in equation 11, keeping only its first order Taylor series correction, which is

$$\sqrt{1 - \zeta(\theta)} \simeq 1 - \frac{\zeta(\theta)}{2} = 1 - \frac{\zeta_m}{1 + \chi(\theta)}. \quad (17)$$

Results for $v_p(\theta)$ and $v_{sv}(\theta)$ then become:

$$v_p^2(\theta) \simeq \frac{1}{\rho} \left\{ [c_{11} \sin^2 \theta + c_{33} \cos^2 \theta] - \frac{\zeta_m [(c_{11} - c_{44}) \sin^2 \theta + (c_{33} - c_{44}) \cos^2 \theta]}{2[1 + \chi(\theta)]} \right\} \quad (18)$$

and

$$v_{sv}^2(\theta) \simeq \frac{1}{\rho} \left\{ c_{44} + \frac{\zeta_m [(c_{11} - c_{44}) \sin^2 \theta + (c_{33} - c_{44}) \cos^2 \theta]}{2[1 + \chi(\theta)]} \right\}. \quad (19)$$

Note that the only approximation made in arriving at equations 18 and 19 again was the approximation of the square root via equation 17.

Clearly, the analysis is not really restricted in any way to using just the first order Taylor approximation in equation 17. For example, other authors (Fowler, 2003; Pederson *et al.*, 2007) have explored rational approximations to such square roots at length. These approaches can certainly be useful in many applications as they provide higher order approximations (not necessarily just first and second order Taylor contributions), while avoiding the computational complexity of the square root operation. Nevertheless, such efforts are beyond our current scope and so will not be discussed further here.

Compact form for $\zeta(\theta)$

More progress can be made by first noting that the quantity $\frac{1}{2}[1 + \chi(\theta)]$ may be written as a perfect square:

$$\frac{1}{2}[1 + \chi(\theta)] = \frac{1}{4} \left(\frac{\tan \theta}{\tan \theta_m} + \frac{\tan \theta_m}{\tan \theta} \right)^2 = \frac{(\tan^2 \theta + \tan^2 \theta_m)^2}{4 \tan^2 \theta \tan^2 \theta_m}. \quad (20)$$

This expression may be simplified using trigonometric identities in the following way. First multiply both the numerator and denominator of equation 20 by $\cos^4 \theta \cos^4 \theta_m$. The denominator of the result is then proportional to $\sin^2 2\theta \sin^2 2\theta_m$, which is a useful form that I will keep. The numerator however is now proportional to the square of

$$\cos^2 \theta \cos^2 \theta_m (\tan^2 \theta + \tan^2 \theta_m) = \sin^2 \theta \cos^2 \theta_m + \sin^2 \theta_m \cos^2 \theta = \frac{1}{2} (1 - \cos 2\theta \cos 2\theta_m), \quad (21)$$

which is another useful form I want to keep. Combining equations 20 and 21, the final result for $\zeta(\theta)$ is therefore

$$\zeta(\theta) = \frac{\zeta_m \sin^2 2\theta_m \sin^2 2\theta}{[1 - \cos 2\theta_m \cos 2\theta]^2}. \quad (22)$$

Equation 22 is the main technical result of this paper, and it is exact. No approximations were made in arriving at equation 22. [Remark: The only approximations made to the wave speeds anywhere in this paper involve Taylor expansions of square roots. So the first

approximations made here, of the form $\sqrt{1 - \zeta(\theta)} \simeq 1 - \zeta(\theta)/2$, do not depend directly on a weak anisotropy assumption, but only on the smallness of ζ_m compared to unity. However, the second ones, *i.e.*, those removing the squares in the formulas for the velocities, do depend directly on a type of weak anisotropy assumption — similar in spirit to Thomsen's (1986) approximations.]

Combining equation 22 with definition 12, it can also be shown that

$$\begin{aligned} [(c_{11} - c_{44}) \sin^2 \theta + (c_{33} - c_{44}) \cos^2 \theta]^2 &= (c_{11} - c_{44})(c_{33} - c_{44}) \frac{4\zeta_m \sin^2 \theta \cos^2 \theta}{\zeta(\theta)} \\ &= (c_{11} - c_{44})(c_{33} - c_{44}) \frac{[1 - \cos 2\theta_m \cos 2\theta]^2}{\sin^2 2\theta_m} \\ &= (c_{11} - c_{44})^2 \tan^2 \theta_m \frac{[1 - \cos 2\theta_m \cos 2\theta]^2}{4 \sin^2 \theta_m \cos^2 \theta_m} \\ &= (c_{11} - c_{44})^2 \frac{[1 - \cos 2\theta_m \cos 2\theta]^2}{4 \cos^4 \theta_m}. \end{aligned}$$

So it follows that

$$\sin^2 \theta + \tan^2 \theta_m \cos^2 \theta = \frac{[1 - \cos 2\theta_m \cos 2\theta]}{2 \cos^2 \theta_m}, \quad (23)$$

which is another useful identity that can be checked directly.

Then, making use of the identity $\sin^2 2\theta_m / \cos^2 \theta_m = 4 \sin^2 \theta_m$, the speed of the quasi-SV-wave is given by

$$\rho v_{sv}^2(\theta) \simeq c_{44} + (c_{11} - c_{44})\zeta_m \frac{2 \sin^2 \theta_m \sin^2 \theta \cos^2 \theta}{[1 - \cos 2\theta_m \cos 2\theta]}. \quad (24)$$

Similarly, the speed of the quasi-P-wave is given (also consistent with equation 24) by

$$\rho v_p^2 \simeq c_{33} + (c_{11} - c_{33}) \sin^2 \theta - (c_{11} - c_{44})\zeta_m \frac{2 \sin^2 \theta_m \sin^2 \theta \cos^2 \theta}{[1 - \cos 2\theta_m \cos 2\theta]}. \quad (25)$$

Again, the only approximation made in these two expressions is the one due to expanding the square root in equation 17.

A tedious but straightforward calculation based on equations 2, 11, and 23 shows that the extreme value of $v_{sv}(\theta)$ — although not exactly at $\theta = \theta_m$ — nevertheless occurs very close to this angle. This calculation is however more technical than others presented here, so it will not be shown explicitly, but the results are confirmed later in the graphical examples. A similar result (but not identical) holds for the extended Thomsen formulas that follow.

Extended Thomsen formulas

A more direct comparison with Thomsen's approximations uses equations 24 and 25 to arrive at approximate formulas for $v_{sv}(\theta)$ and $v_p(\theta)$ analogous to Thomsen's. The resulting expressions, which may be called “extended Thomsen formulas,” are given by

$$v_p(\theta)/v_p(0) \simeq 1 + \epsilon \sin^2 \theta - (\epsilon - \delta) \frac{2 \sin^2 \theta_m \sin^2 \theta \cos^2 \theta}{[1 - \cos 2\theta_m \cos 2\theta]} \quad (26)$$

and

$$v_{sv}(\theta)/v_s(0) \simeq 1 + \left[v_p^2(0)/v_s^2(0) \right] (\epsilon - \delta) \frac{2 \sin^2 \theta_m \sin^2 \theta \cos^2 \theta}{[1 - \cos 2\theta_m \cos 2\theta]}. \quad (27)$$

Equations 26 and 27 are the two main *approximate* results of this paper. So far only two approximations have been made, and both of these came from expanding a square root in a Taylor series, and retaining only the first nontrivial term.

Comparing equations 26 and 27 to equations 6 and 7, the differences are found to lie in a factor of the form:

$$\frac{2 \sin^2 \theta_m}{[1 - \cos 2\theta_m \cos 2\theta]} \rightarrow \frac{1}{2 \cos^2 \theta_m} \quad \text{as} \quad \theta \rightarrow \theta_m, \quad (28)$$

which depends explicitly on the angle θ_m determined by $\tan^2 \theta_m = (c_{33} - c_{44})/(c_{11} - c_{44})$, and also on θ itself. As indicated, the expression goes to $1/2 \cos^2 \theta_m$ in the limit of $\theta \rightarrow \theta_m$, which is also in agreement with the results for $v_{sv}(\theta_m)$ in Appendix A. But, since $\sin^2 \theta_m = \tan^2 \theta_m / (1 + \tan^2 \theta_m)$ and $\cos 2\theta_m = (1 - \tan^2 \theta_m) / (1 + \tan^2 \theta_m)$, useful identities are

$$\sin^2 \theta_m = \frac{c_{33} - c_{44}}{c_{11} + c_{33} - 2c_{44}} = 1 - \cos^2 \theta_m \quad (29)$$

and

$$\cos 2\theta_m = \frac{c_{11} - c_{33}}{c_{11} + c_{33} - 2c_{44}} = 1 - 2 \sin^2 \theta_m. \quad (30)$$

These results can therefore be used, after deducing some of the elastic constants from field data at near offsets, in order to extend the validity of the equations to greater angles and farther offsets. Inversion of such data is however beyond this paper's scope.

To make the formulas 26 and 27 look as much as possible like Thomsen's formulas — and thereby arrive at a somewhat different understanding of equations 7 and 8, first eliminate θ_m by arbitrarily setting it equal to some value such as $\theta_m = 45^\circ$, in which case $2 \sin^2 \theta_m = 1$ and $\cos 2\theta_m = 0$. Then, the θ dependence in the denominators goes away, and Thomsen's formulas 7 and 8 are recovered exactly. The particular choice $\theta_m = 45^\circ$ is however completely unnecessary as shall be shown, and furthermore is never valid for any anisotropic medium having $c_{11} \neq c_{33}$.

DEDUCING θ_M FROM SEISMIC DATA

In the extended formulas for seismic data, the key quantity needed is clearly the value of the angle θ_m . However, this value is quite easily determined since

$$\tan^2 \theta_m = \frac{c_{33} - c_{44}}{c_{11} - c_{44}} = \frac{v_p^2(0) - v_s^2(0)}{(c_{11}/\rho) - v_s^2(0)} \quad (31)$$

and

$$\epsilon = \frac{c_{11} - c_{33}}{2c_{33}} = \frac{c_{11}/\rho - v_p^2(0)}{2v_p^2(0)}. \quad (32)$$

Therefore,

$$\tan^2 \theta_m = \frac{v_p^2(0) - v_s^2(0)}{(1 + 2\epsilon)v_p^2(0) - v_s^2(0)}. \quad (33)$$

Thus, θ_m is completely determined by the same data used in the standard analysis of reflection seismic data, which determines the various small angle wave speeds and the Thomsen weak anisotropy parameters.

The pertinent fixed factors for use in the formulas are given by

$$\sin^2 \theta_m = \frac{v_p^2(0) - v_s^2(0)}{2[(1 + \epsilon)v_p^2(0) - v_s^2(0)]} \quad (34)$$

and

$$\cos 2\theta_m = \frac{\epsilon v_p^2(0)}{(1 + \epsilon)v_p^2(0) - v_s^2(0)}. \quad (35)$$

Finally, equation 14 also shows how to determine the extreme value $\zeta_m = \zeta(\theta_m)$ using the same data. Examples of such computations are presented in TABLE 1 for various anisotropic rock types. Data for these examples comes from Thomsen's TABLE 1, and no other information is required.

TABLE 1. Examples of ζ_m — *i.e.*, the extreme value $\zeta(\theta_m)$ — and its angular location θ_m for various rocks and minerals. The data for ϵ , δ , $v_p(0)$, and $v_s(0)$ are all taken from Table 1 of Thomsen (1986).

Sample	ϵ	δ	$v_p(0)$ (m/s)	$v_s(0)$ (m/s)	ζ_m	θ_m
Cotton Valley shale	0.135	0.205	4721.	2890.	-0.1564	39.89°
Mesaverde sandstone	0.081	0.057	3688.	2774.	0.0805	40.48°
Muscovite crystal	1.12	-0.235	4420.	2091.	0.8985	26.90°
Pierre shale	0.015	0.060	2202.	969.	-0.1076	44.48°
Taylor sandstone	0.110	-0.035	3368.	1829.	0.3135	41.12°
Wills Point shale	0.215	0.315	1058.	387.	-0.1543	39.27°

NORMAL MOVEOUT CORRECTIONS

The altered forms of $v_p(\theta)$ and $v_{sv}(\theta)$ in equations 26 and 27 suggest that it might also be necessary to alter the normal moveout (NMO) corrections to the velocities (Tsvankin, 2005, p. 113). It is easy to see that these corrections are now given by

$$V_{NMO,p} = v_p(0)\sqrt{1 + 2\delta}, \quad (36)$$

for the quasi-P-wave, and,

$$V_{NMO,sv} = v_s(0)\sqrt{1 + 2\sigma}, \quad (37)$$

for the quasi-SV-wave, where

$$\sigma = \left[v_p^2(0)/v_s^2(0) \right] (\epsilon - \delta). \quad (38)$$

These corrections to the NMO velocities are exactly the same as those for Thomsen's weak anisotropy approximation because the factor that is pertinent, and that might have potential to alter these expressions is given, in the small angle limit $\theta \rightarrow 0$, by

$$\frac{2 \sin^2 \theta_m}{1 - \cos 2\theta_m} \equiv 1, \quad (39)$$

which holds for any value of θ_m (including both $\theta_m \rightarrow 0$ and $\theta_m = 45^0$). Since Thomsen's formulas accurately approximate all three wave speeds in this limit by design, the present formulas share this accuracy (and in some cases — *i.e.*, larger offsets — improves upon it). Therefore, no changes are needed in short offset (small θ) data processing.

The NMO correction for the SH-wave clearly does not change either, since it does not depend on this new factor.

RESERVOIRS WITH VERTICALLY ORIENTED FRACTURES

To provide some pertinent examples of results for the types of anisotropic media most interesting in oil and gas reservoirs, two distinct types of reservoirs having vertical fractures will now be considered. The first case studied will have vertical fractures that are not preferentially aligned, so the reservoir symmetry is vertical transverse isotropy (VTI). The second case will also have vertical fractures but these will be preferentially aligned, so the reservoir symmetry will be horizontal transverse isotropy (HTI) and, therefore, exhibit azimuthal (angle ϕ dependent) anisotropy.

These two reservoir fracture models will be built up using results from recent numerical experiments by Grechka and Kachanov (2006a,b). Those results were analyzed by Berryman and Grechka (2006) in light of the crack influence parameter formalism of Kachanov (1980) and Sayers and Kachanov (1991). The significance of two crack influence parameters — η_i , for $i = 1, 2$ — for the case of aligned horizontal cracks for lower crack densities $\rho_c = na^3$ (where $n = N/V$ is the number density of cracks — N being the total number per volume V — and for penny-shaped cracks a is the radius of the circular penny crack-face while b/a is called the aspect ratio) is:

$$\Delta S_{ij}^{(1H)} = \rho_c \begin{pmatrix} 0 & 0 & \eta_1 & & \\ 0 & 0 & \eta_1 & & \\ \eta_1 & \eta_1 & 2(\eta_1 + \eta_2) & & \\ & & & 2\eta_2 & \\ & & & & 2\eta_2 \\ & & & & 0 \end{pmatrix}. \quad (40)$$

For smaller values of crack density ρ_c , equation 40 shows how the presence of penny-shaped cracks increases the compliance of the reservoir. [Note that η_1 is usually small and most often negligible, while η_2 is positive and nonnegligible.] Typical values of crack density ρ_c for reservoirs are $\rho_c \leq 0.1$. The matrix $\Delta S_{ij}^{(1H)}$ is the lowest order compliance correction matrix and should be added to the isotropic compliance matrix

$$\Delta S_{ij}^{(0)} = \begin{pmatrix} 1/E & -\nu/E & -\nu/E & & \\ -\nu/E & 1/E & -\nu/E & & \\ -\nu/E & -\nu/E & 1/E & & \\ & & & 1/G & \\ & & & & 1/G \\ & & & & & 1/G \end{pmatrix}, \quad (41)$$

where $\nu = \lambda/2(\lambda + \mu)$ is Poisson's ratio, $G = \mu$ is the shear modulus, and $E = 2(1 + \nu)G$ is Young's modulus of the (assumed) isotropic background medium. Summing equations 41 and 40 produces the compliance matrix for a horizontally cracked, VTI elastic medium. This combined matrix is then used to compute the behavior of a simple HTI reservoir with aligned vertical cracks using the methods described in Appendix B.

For vertical fractures with randomly oriented axes of symmetry, the resulting VTI medium has a low crack density correction matrix of the form

$$\Delta S_{ij}^{(1V)} = \rho_c \begin{pmatrix} (\eta_1 + \eta_2) & \eta_1 & \eta_1/2 \\ \eta_1 & (\eta_1 + \eta_2) & \eta_1/2 \\ \eta_1/2 & \eta_1/2 & 0 \\ & & \eta_2 \\ & & \eta_2 \\ & & 2\eta_2 \end{pmatrix}, \quad (42)$$

in which the η 's have the same values as those in equation 40 if the only difference between the cracks in equations 42 and 40 is their orientation. Note that $2\Delta S_{ij}^{(1V)} + \Delta S_{ij}^{(1H)}$ is an isotropic correction matrix for a system having crack density $3\rho_c$. Summing equations 41 and 42 produces the compliance matrix for a vertically cracked VTI elastic medium, in which the crack normals are randomly and/or uniformly distributed in the horizontal plane.

Higher order corrections (*i.e.*, second order in powers of ρ_c) in the Sayers and Kachanov (1991) formulation with the three crack influence parameters η_i , for $i = 3, 4, 5$, take the form (again using the Voigt matrix notation):

$$\Delta S_{ij}^{(2H)} = \rho_c^2 \begin{pmatrix} 0 & 0 & \eta_4 \\ 0 & 0 & \eta_4 \\ \eta_4 & \eta_4 & 2(\eta_3 + \eta_4 + \eta_5) \\ & & 2\eta_5 \\ & & 2\eta_5 \\ & & 0 \end{pmatrix} \quad (43)$$

for horizontal fractures — *i.e.*, to be combined with equation 40. Similarly,

$$\Delta S_{ij}^{(2V)} = \rho_c^2 \begin{pmatrix} (\eta_3 + \eta_4 + \eta_5) & \eta_4 & \eta_4/2 \\ \eta_4 & (\eta_3 + \eta_4 + \eta_5) & \eta_4/2 \\ \eta_4/2 & \eta_4/2 & 0 \\ & & \eta_5 \\ & & \eta_5 \\ & & 2(\eta_3 + \eta_5) \end{pmatrix} \quad (44)$$

for the random vertical fractures producing VTI symmetry – to be combined with equation 42.

Examples of values of all five of these crack influence parameters have been obtained based on the numerical studies of Grechka and Kachanov (2006a,b) by Berryman and

Grechka (2006). The two models considered have very different Poisson's ratios for the isotropic background media: (a) $\nu_0 = 0.00$ and (b) $\nu_0 = 0.4375$. We will call these two models, respectively, the first model and the second model. The first model has background stiffness matrix values $c_{11} = c_{22} = c_{33} = 13.75$ GPa, $c_{12} = c_{13} = c_{23} = 0.00$ GPa, and $c_{44} = c_{55} = c_{66} = 6.875$ GPa. Bulk modulus for this model is therefore $K_0 = 4.583$ GPa and shear modulus is $G_0 = 6.875$ GPa. The purpose of this model is to provide as much contrast as possible with the following model, while still retaining a physically pertinent value of Poisson's ratio (for which values typically lie in the range $0.0 \leq \nu_0 \leq 0.5$). The second model has stiffness matrix values $c_{11} = c_{22} = c_{33} = 19.80$ GPa, $c_{12} = c_{13} = c_{23} = 15.40$ GPa, and $c_{44} = c_{55} = c_{66} = 2.20$ GPa. Bulk modulus for this model is therefore $K_0 = 16.86$ GPa and shear modulus is $G_0 = 2.20$ GPa. The second model may be seen to correspond to a background material having compressional wave speed $V_p = 3$ km/s, shear wave speed $V_s = 1$ km/s, and mass density $\rho_m = 2200.0$ kg/m³, and is therefore more typical of a sandstone reservoir. Detailed discussion of the method used to obtain the crack influence parameters is given by Berryman and Grechka (2006), and will not be repeated here. Results are listed in TABLE 2.

In all the following plots, the exact curves (as computed for the model c_{ij} 's) are plotted first in black; then the Thomsen approximation is plotted in red; and finally the new approximation is plotted in blue. Thus, in those examples where red curves appear to be missing, this happens because the blue curves lie right on top of the red ones (to graphical accuracy). This overlay effect is expected whenever θ_m approaches 45°, which can happen at low crack densities since the background medium has been taken to be isotropic.

TABLE 2. Values of five fracture-influence parameters for the two models considered, from Berryman and Grechka (2006).

<i>Fracture-influence Parameters</i>	<i>First Model</i> $\nu_0 = 0.00$	<i>Second Model</i> $\nu_0 = 0.4375$
η_1 (GPa ⁻¹)	0.0000	-0.0192
η_2 (GPa ⁻¹)	0.1941	0.3994
η_3 (GPa ⁻¹)	-0.3666	-1.3750
η_4 (GPa ⁻¹)	0.0000	0.0000
η_5 (GPa ⁻¹)	0.0917	0.5500

TABLE 3. Values of ζ_m [the extreme value of $\zeta(\theta)$] for the four models considered. The model fracture density is ρ_c . The model Poisson ratio for the background medium is ν_0 .

<i>Model</i>	ζ_m	ζ_m	ζ_m
	for $\rho_c = 0.05$	for $\rho_c = 0.10$	for $\rho_c = 0.20$
HTI, $\nu_0 = 0.00$	0.0277	0.0973	0.2943
VTI, $\nu_0 = 0.00$	0.0148	0.0558	0.1965
HTI, $\nu_0 = 0.4375$	0.0102	0.0441	0.1595
	for $\rho_c = 0.025$	for $\rho_c = 0.050$	for $\rho_c = 0.100$
VTI, $\nu_0 = 0.4375$	0.0011	0.0051	0.0210

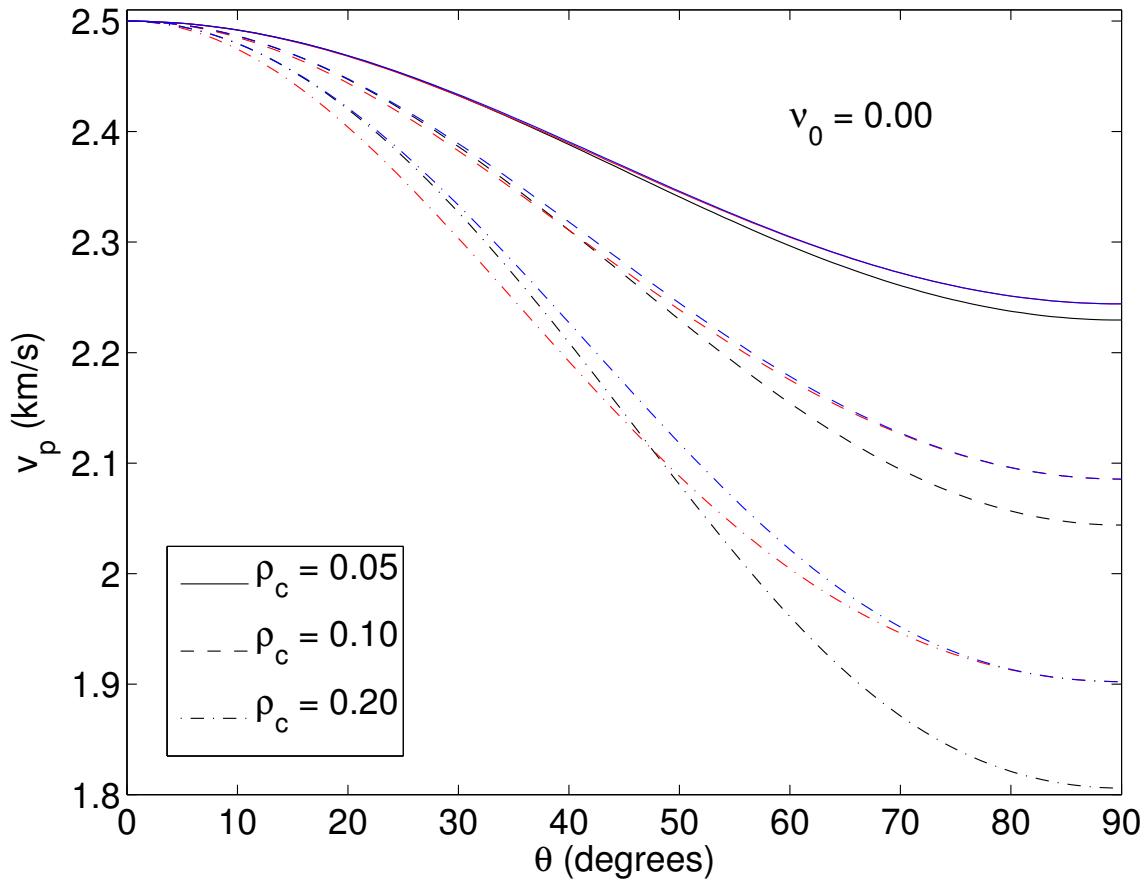


Figure 1: For randomly aligned vertical fractures and VTI symmetry: examples of anisotropic quasi-P compressional wave speed (v_p) for Poisson's ratio of the host medium $v_0 = 0.00$. Velocity curves in black are exact for the fracture model discussed in the text. The Thomsen weak anisotropy velocity curves for the same fracture model are then overlaid in red. Finally, the new curves for the extended Thomsen approximation valid for stronger anisotropies are overlaid in blue. If any of these curves is not visible, it is because one or possibly two other curves are covering them. jim1/.FIG1

For reference purposes, the computed values of ζ_m are also presented in TABLE 3.

VTI Symmetry

Figures 1–6 present results for the case of vertical fractures having an isotropic distribution of normals (symmetry axes) in the horizontal plane. The resulting medium has VTI symmetry.

A first observation is that the low crack density results for $v_{sv}(\theta)$ are nearly constant, showing that $\epsilon - \delta \simeq 0$. When this happens for $v_{sv}(\theta)$, it is also true that $v_p(\theta)$ is approximately elliptical. Of course, the exact results for $v_{sh}(\theta)$ are always elliptical, but the Thomsen and new approximate results are only approximately elliptical.

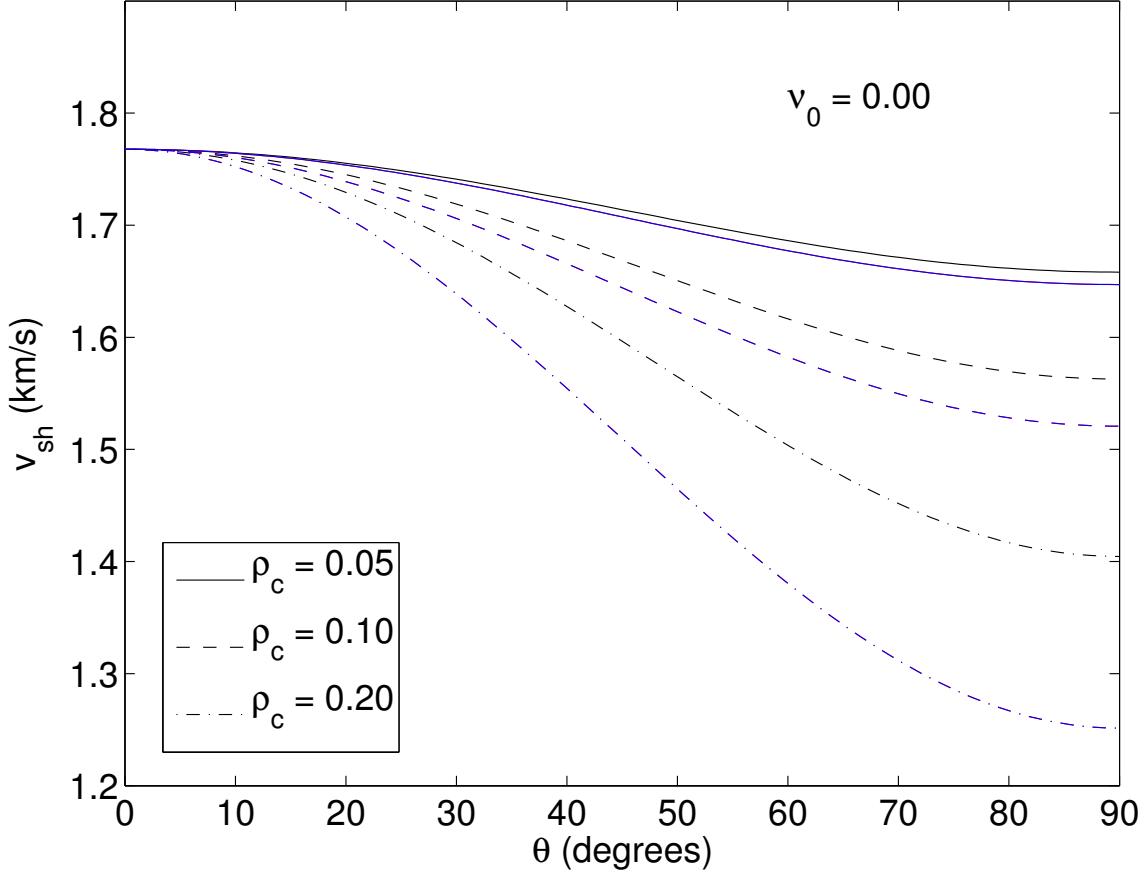


Figure 2: Same as Figure 1 for SH shear wave speed (v_{sh}). jim1/.FIG2

Secondly, all three velocity models (exact, Thomsen, and the new approximation) give very similar results for all cases shown when $\nu_0 = 0.4375$. There are however some significant differences among the results for $\nu_0 = 0.00$, especially for $v_{sv}(\theta)$ and $v_p(\theta)$ – the largest deviations from the exact curves being those for Thomsen's approximations (red curves) in both cases.

HTI Symmetry

Figures 7–12 present results for vertical fractures having their normals (axes of symmetry) aligned in some direction (say $x_3 = x$). The fracture models considered are the same and use the same data as for the preceding (VTI) case.

Thomsen's approximation and the new one are virtually identical here in $v_{sh}(\theta)$ for both $\nu_0 = 0.00$ and $\nu_0 = 0.4375$. For $v_{sv}(\theta)$, Thomsen's approximation is higher than the exact result, while the new approximation is lower.

Results for $v_p(\theta)$ in both Thomsen's and the new approximation are comparable to the exact results for θ 's up to about 45° – 50° , but are not identical to each other or to the exact

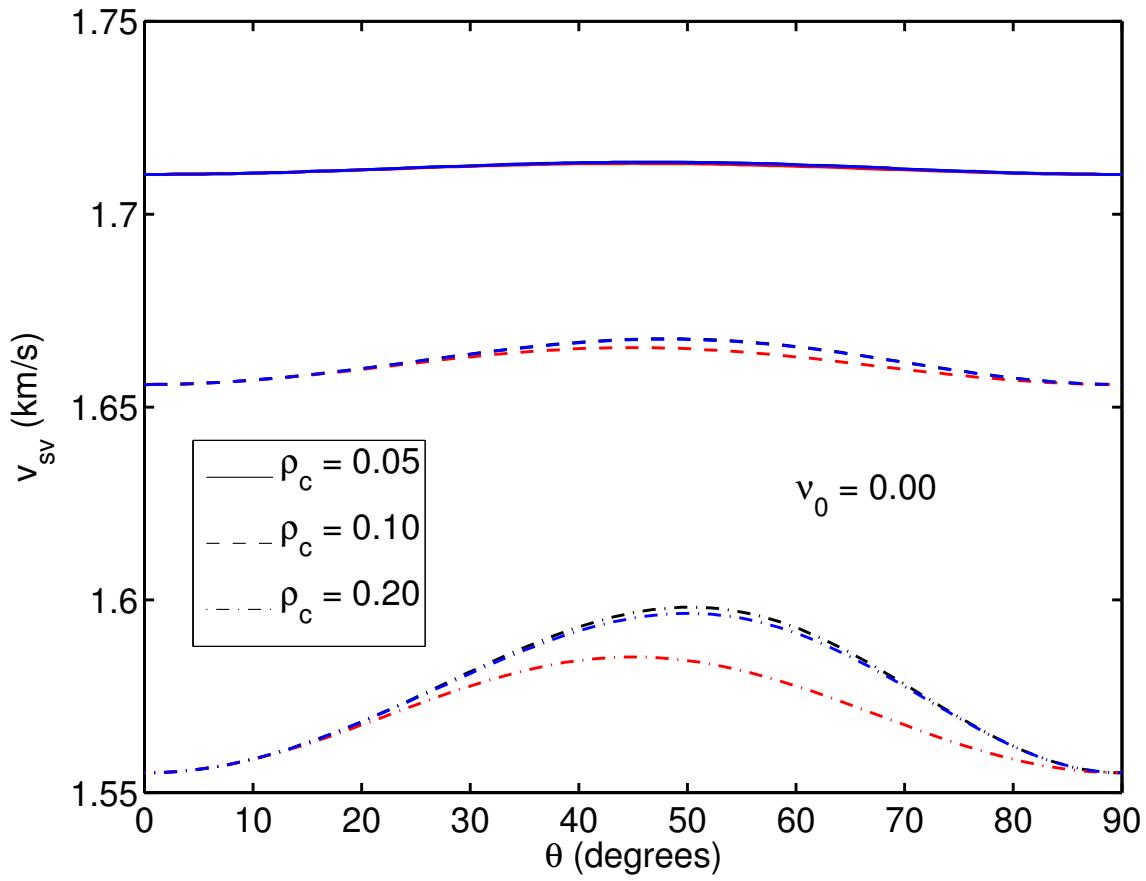


Figure 3: Same as Figure 1 for quasi-SV shear wave speed (v_{sv}). [jim1/. FIG3]

result. For $\nu_0 = 0.4375$, agreement among the three curves is good for $v_p(\theta)$, but not so good for $v_{sv}(\theta)$ or $v_{sh}(\theta)$.

SUMMARY AND CONCLUSIONS

The main technical result of the paper is equation 22, showing directly how $\zeta(\theta)$ is related to θ_m and ζ_m . The most significant applications of this result are summarized in equations 26 and 27. These formulas generalize (*i.e.*, extend the validity of) Thomsen's weak anisotropy approach to wider ranges of angles, and stronger anisotropies. These formulas have the clear advantage that they require no more data analysis than Thomsen's formulas for weak anisotropy, but they give more accurate predictions of the wave speeds for longer offsets. In particular, the new formulas are designed to give the peak (or possibly the trough — if the difference $\epsilon - \delta$ happens to be negative) of the quasi-SV-wave in the right location, (*i.e.*, the correct angle $\theta = \theta_m$ with the vertical), even though the magnitude of these velocities may still be a bit off. For quasi-SV waves, this error made in the velocity magnitude is always less than that made using the original Thomsen formulas. For quasi-P waves the results are somewhat mixed because the errors introduced by poor approximations to $\zeta(\theta)$ can have either sign, positive or negative, depending on what the actual value of θ_m happens

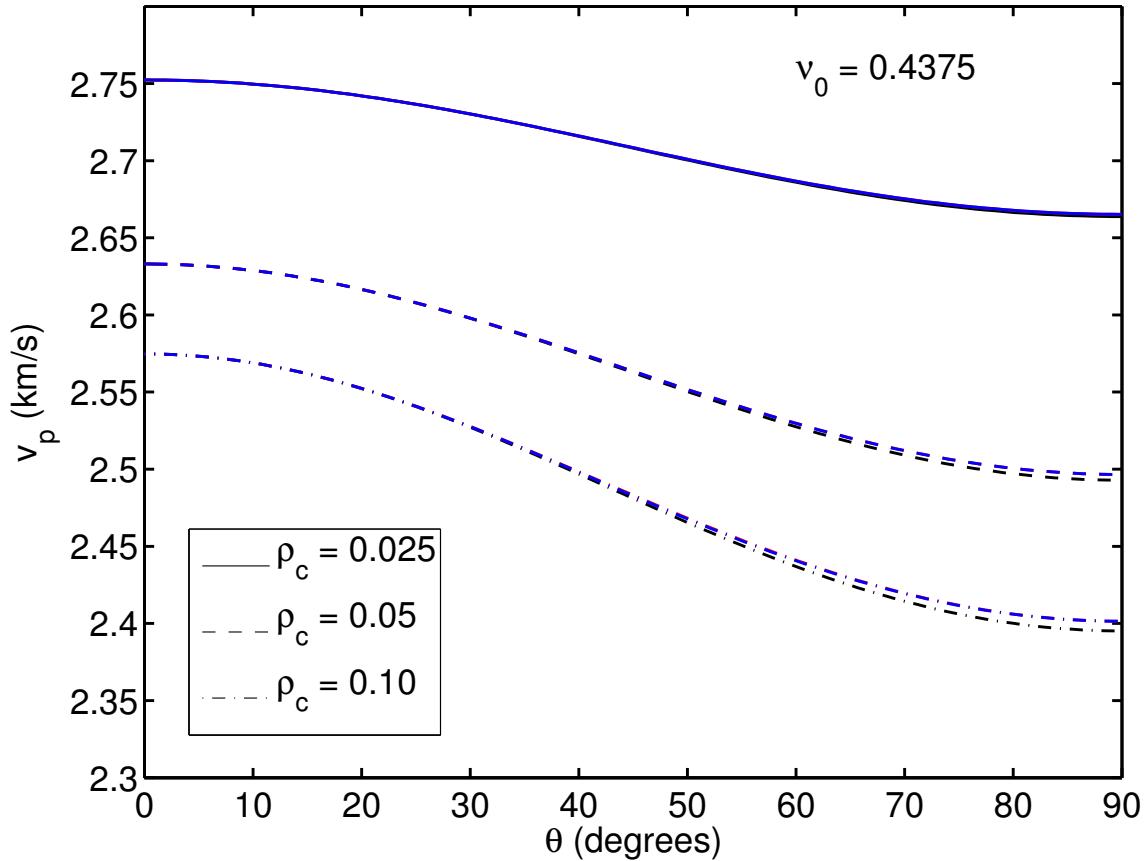


Figure 4: Same as Figure 1, for a different background medium having Poisson's ratio $\nu_0 = 0.4375$. jim1/. FIG4

to be. This fact shows that Thomsen's approximation will sometimes give better results at smaller θ than the new formulas, but other times they will be worse. This fluctuation in the behavior for quasi-P waves is observed in the examples contained in the Figures. Thus, the new approximation does have the advantage of consistency.

Furthermore, the only new parameter needed for implementation is the angle θ_m itself; however, the value θ_m is also determined directly from the same data required to compute all the Thomsen parameters (for example, see TABLE 1). A final advantage that is especially helpful for the practical use of the method is that the corrections needed for all the NMO velocities do not change, and so are exactly the same as for Thomsen's formulation.

DISCUSSION

The work presented here has necessarily been restricted to VTI and HTI symmetries, because these correspond to the simplest and most studied cases in the anisotropy literature. It has sometimes been noted, however, that the HTI symmetry in particular is actually a fairly unrealistic model for seismic exploration problems (Schoenberg and Helbig, 1997;

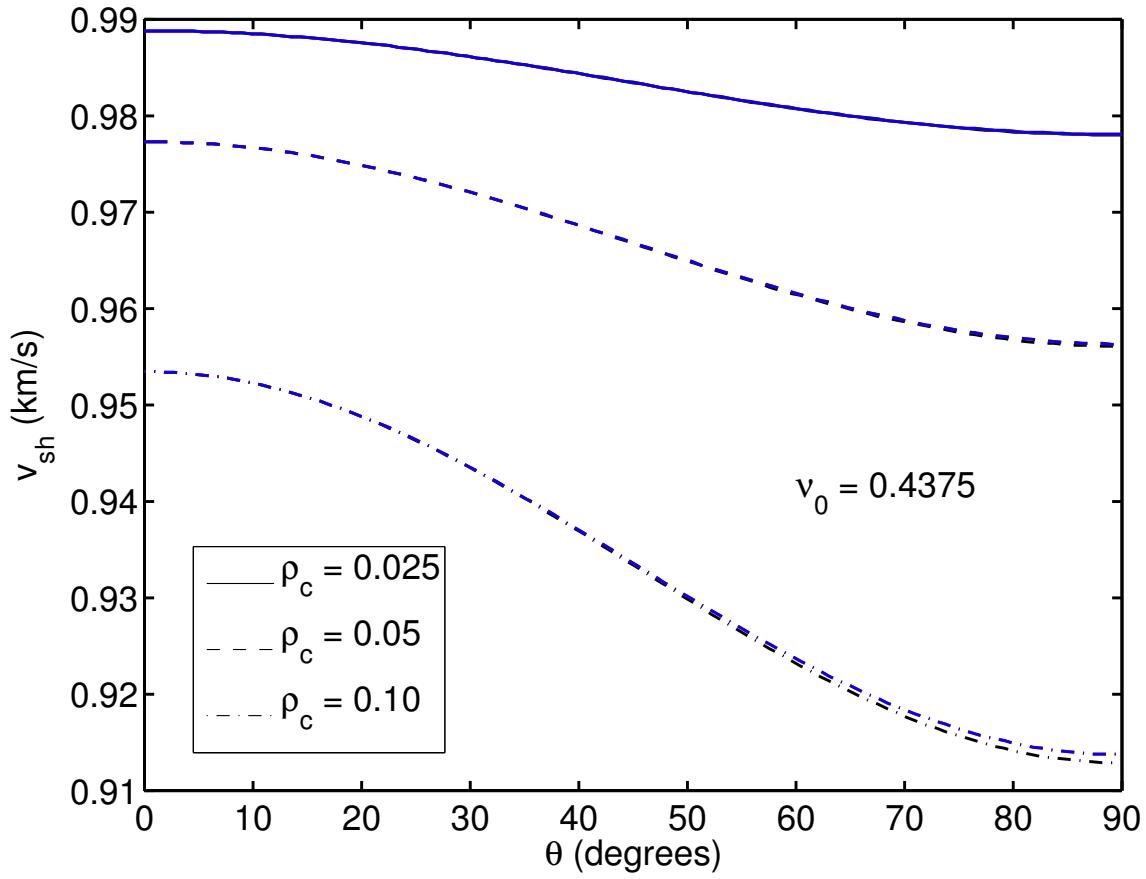


Figure 5: Same as Figure 2, but the value of $\nu_0 = 0.4375$. [jim1/. FIG5]

Tsvankin, 1997, 2005; Thomsen, 2002). The reason for this is that the earth, to a first approximation, is often horizontally layered and such horizontal layering is well-known to produce VTI symmetry (Postma, 1955; Backus, 1962). If aligned vertical fractures are superimposed on this already anisotropic background medium (unlike the isotropic background medium used in the models presented here), then the resulting symmetry is likely to be closer to orthorhombic (having nine independent elastic constants) than to HTI (having at most only five independent constants). This viewpoint no doubt provides a valid criticism of the work presented in the examples as far its value for practical applications. However, the author expects the present paper to be followed by others on this topic, and future efforts will be devoted to obtaining comparable results for such orthotropic systems (Mensch and Rasolofosaon, 1997), and thereby becoming more realistic for exploration purposes.

All the results presented here are specifically for phase velocities of the seismic waves. In heterogeneous media, it is instead the ray (or group) velocities that are needed for ray tracing applications, and in particular for wave equation migration. However, recent work by Zhu *et al.* (2005; 2007) has reformulated the Gaussian beam migration approach to permit direct use of phase velocities, with a corresponding reduction in the overall computational burden. Although it is too soon to be certain which potential applications of the results contained herein may prove to be of value, it is seems likely that this particular application

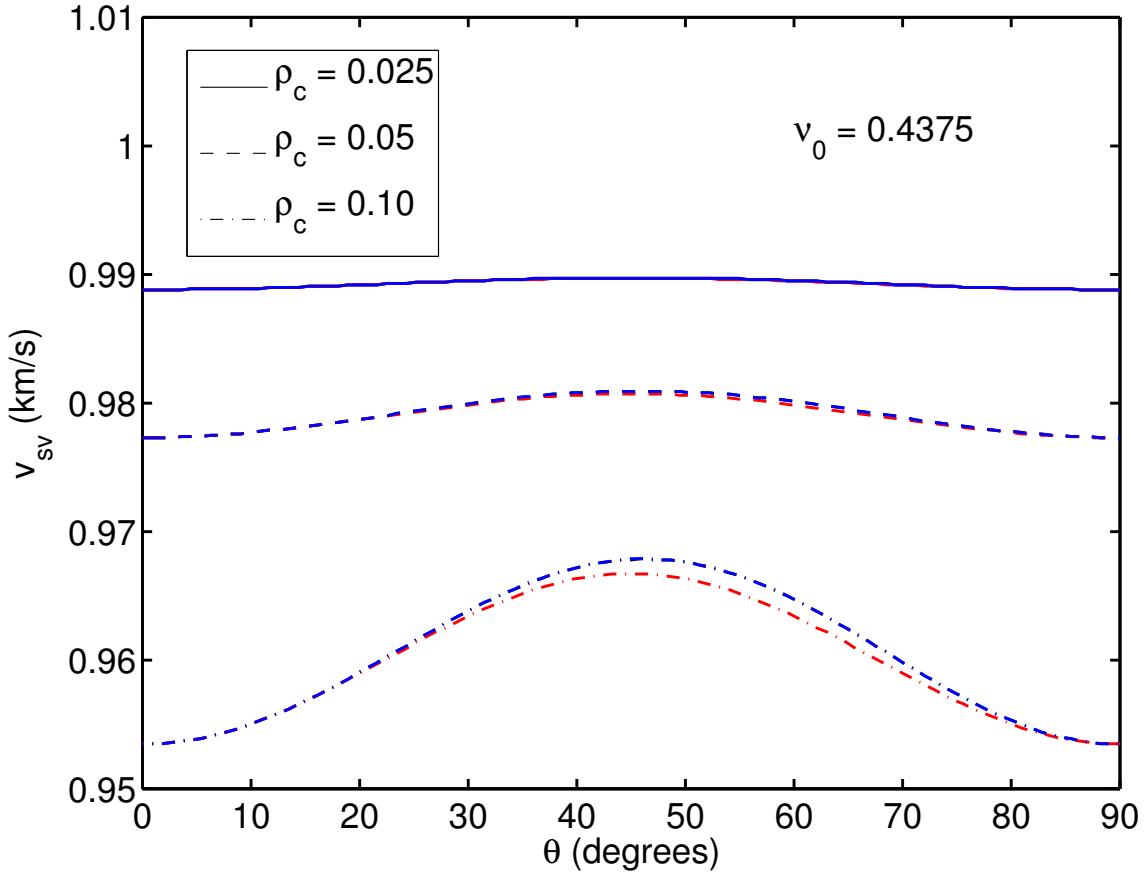


Figure 6: Same as Figure 3, but the value of $\nu_0 = 0.4375$. jim1/. FIG6

will be one of the more interesting ones for seismic data analysis.

ACKNOWLEDGMENTS

The author thanks V. Grechka, M. Kachanov, S. Nakagawa, and I. Tsvankin for helpful discussions and suggestions.

APPENDIX A: $V_{SV}(\theta_M)$

For purposes of comparison, it is useful to know the exact value and also some related approximations to the exact value of the quasi-SV wave speed $v_{sv}(\theta)$ at the angle $\theta = \theta_m = \pi/2$ — which occurs close to (but not exactly at) the extreme value of $v_{sv}(\theta)$ over all angles (see discussion after equation 25 in the main text).

Evaluation gives the exact result

$$v_{sv}^2(\theta_m) = \frac{\sin^2 \theta_m}{2\rho} (c_{11} - c_{44}) \left[\frac{c_{11} + c_{44}}{c_{11} - c_{44}} + \frac{c_{33} + c_{44}}{c_{33} - c_{44}} - 2\sqrt{1 - \zeta_m} \right]. \quad (\text{A-1})$$

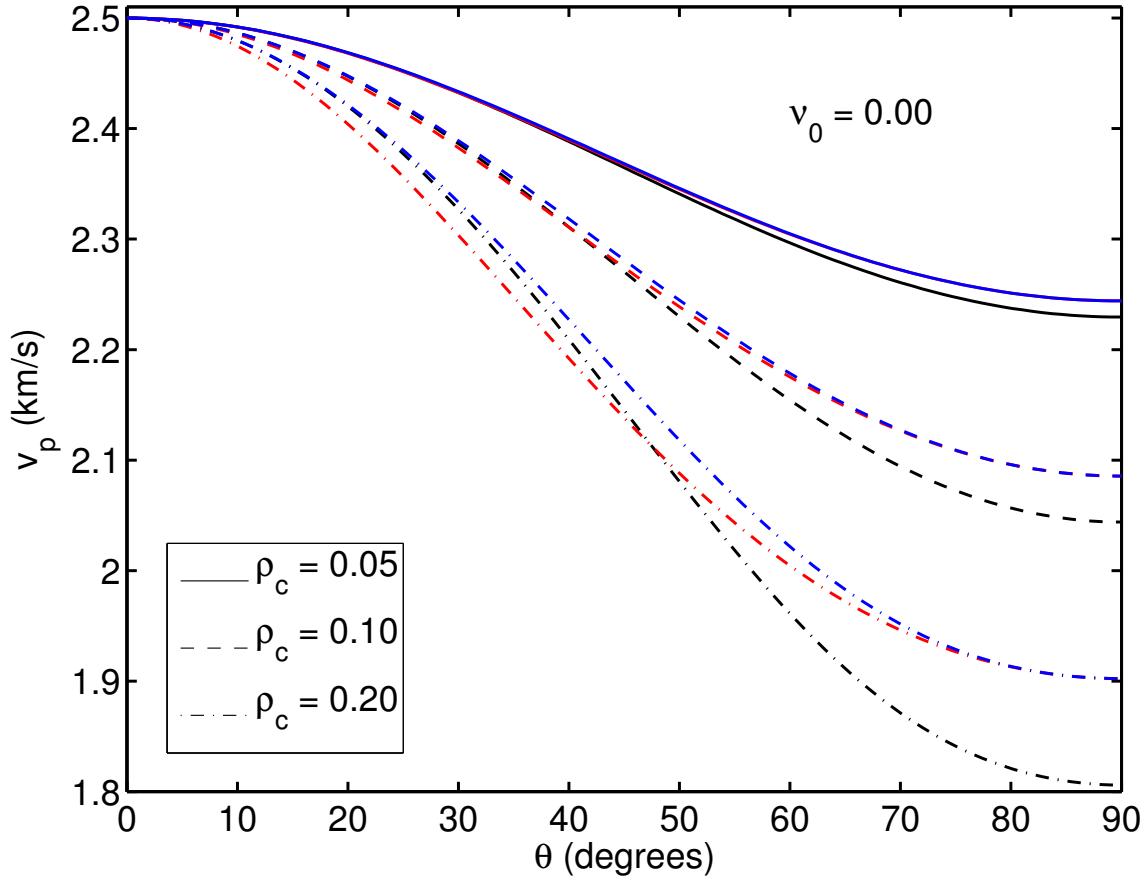


Figure 7: For aligned vertical fractures and HTI symmetry: examples of anisotropic quasi-P compressional wave speed (v_p) for Poisson's ratio of the host medium $\nu_0 = 0.00$. Velocity curves in black are exact for the fracture model discussed in the text. The Thomsen weak anisotropy velocity curves for the same fracture model are then overlaid in red. Finally, the new curves for the extended Thomsen approximation valid for stronger anisotropies are overlaid in blue. If any of these curves is not visible, it is because one or possibly two other curves are covering them. jim1/. FIG7

After substituting $\sin^2 \theta_m = (c_{33} - c_{44})/(c_{11} + c_{33} - 2c_{44})$, expanding the square root $\sqrt{1 - \zeta_m} \simeq 1 - \zeta_m/2$, and several more steps of simplification, a useful approximate expression is

$$v_{sv}^2(\theta_m) \simeq v_s^2(0) \left[1 + \frac{\zeta_m}{2} \frac{(c_{11} - c_{44})(c_{33} - c_{44})}{c_{44}(c_{11} + c_{33} - 2c_{44})} \right]. \quad (\text{A-2})$$

And finally, by approximating the square root of this expression and using (14), we have

$$\frac{v_{sv}(\theta_m)}{v_s(0)} \simeq 1 + \frac{\zeta_m(c_{11} - c_{44})\sin^2 \theta_m}{4c_{44}} = 1 + \left[v_p^2(0)/v_s^2(0) \right] (\epsilon - \delta) \frac{\sin^2 \theta_m}{2}. \quad (\text{A-3})$$

Equation A-3 can be directly compared to Thomsen's formula for $v_{sv}(\theta)$ in equation 8. The only difference is a factor of $2 \cos^2 \theta_m$ in the final term. This factor could be unity if $\theta_m = 45^\circ$, but — since this never happens for anisotropic media — the factor always differs

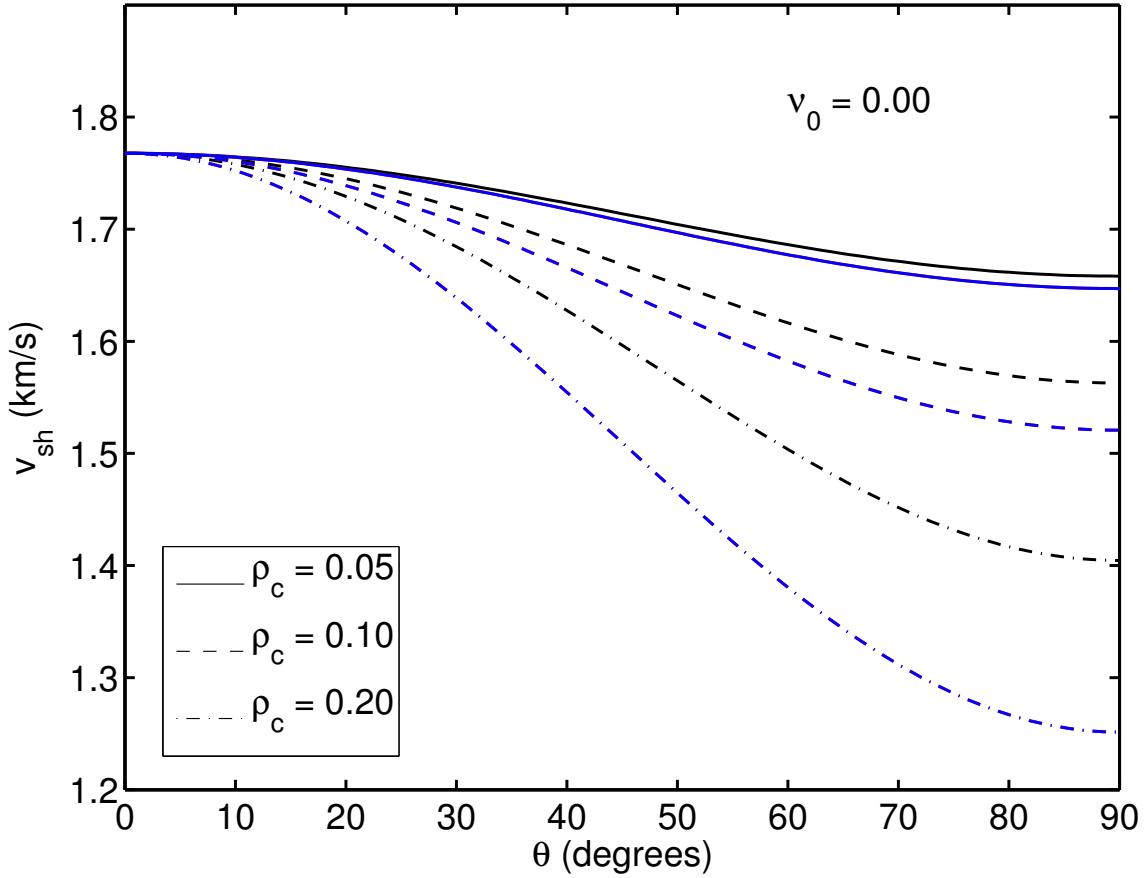


Figure 8: Same as Figure 7 for SH shear wave speed (v_{sh}). jim1/.FIG8

from unity and can be either higher or lower than unity depending on whether θ_m is less than or greater than 45° .

APPENDIX B: HTI FORMULAS FROM VTI FORMULAS

Probably the easiest way to obtain formulas pertinent to HTI (horizontal transverse isotropy) from VTI (vertical transverse isotropy) is to leave the stiffness matrix c_{ij} alone, and simply reinterpret the meaning of the Cartesian indices i, j . For VTI media, one typical choice is $x_3 = z$, where \hat{z} is the vertical direction at the surface of the earth, or more often the direction down into the earth. Then, the angle of incidence θ is measured with respect to \hat{z} , where $\theta = 0$ means parallel to \hat{z} and pointing into the earth, and $\theta = \pi/2$ means horizontal incidence.

Considering aligned vertical fractures, with their axes of symmetry in the direction $x \equiv x_3$, the matrix c_{ij} itself presented in the main text need not change, but the meaning of the angles does change. Clearly, the simplest case to study — and the only one analyzed here — is the case of waves propagating at some angle to vertical but always having a component in the direction $x_3 = x$, while also having $x_2 = y = 0$, thus lying in the xz -plane

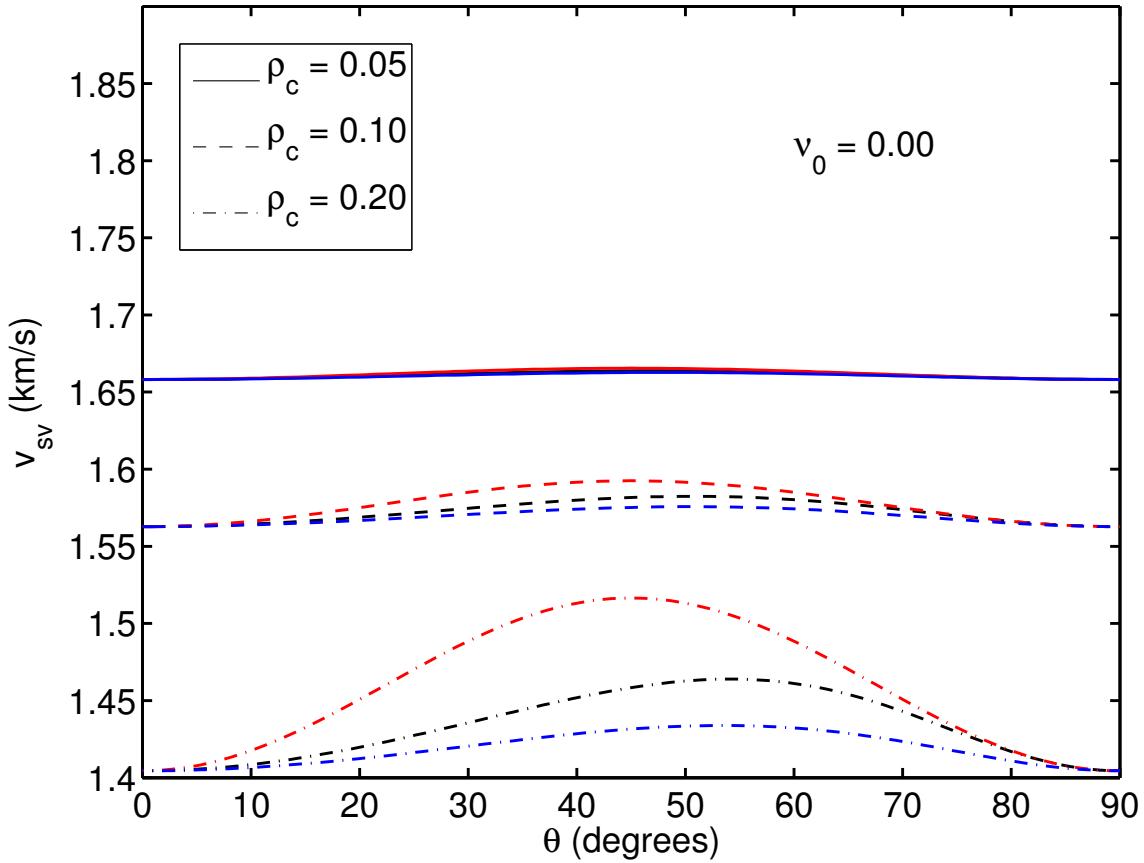


Figure 9: Same as Figure 7 for quasi-SV shear wave speed (v_{sv}). [jim1/. FIG9]

perpendicular to the fracture plane. (This case is special, but all other wave speeds at other angles of propagation are easily found as a linear combination — depending specifically on the azimuthal angle ϕ at the earth surface — of these values and those in the plane of the fractures themselves.) Then,

$$\theta^H + \theta^V = \frac{\pi}{2}, \quad (\text{B-1})$$

where θ^V corresponds exactly to the θ appearing in all the formulas up to equation 39 in the main text, and θ^H is the effective angle in the xz -plane of incidence under consideration, *i.e.*, the one perpendicular to the vertical fractures in the reservoir. To obtain wave speeds at the angle θ^H , just substitute $\theta \equiv \theta^V = \frac{\pi}{2} - \theta^H$, or write the speeds as

$${}^H v_p^2(\theta^H) \equiv v_p^2(\theta^V) = v_p^2\left(\frac{\pi}{2} - \theta^H\right), \quad (\text{B-2})$$

$${}^H v_{sv}^2(\theta^H) \equiv v_{sv}^2(\theta^V) = v_p^2\left(\frac{\pi}{2} - \theta^H\right), \quad (\text{B-3})$$

and

$${}^H v_{sh}^2(\theta^H) \equiv v_{sh}^2(\theta^V) = v_p^2\left(\frac{\pi}{2} - \theta^H\right). \quad (\text{B-4})$$

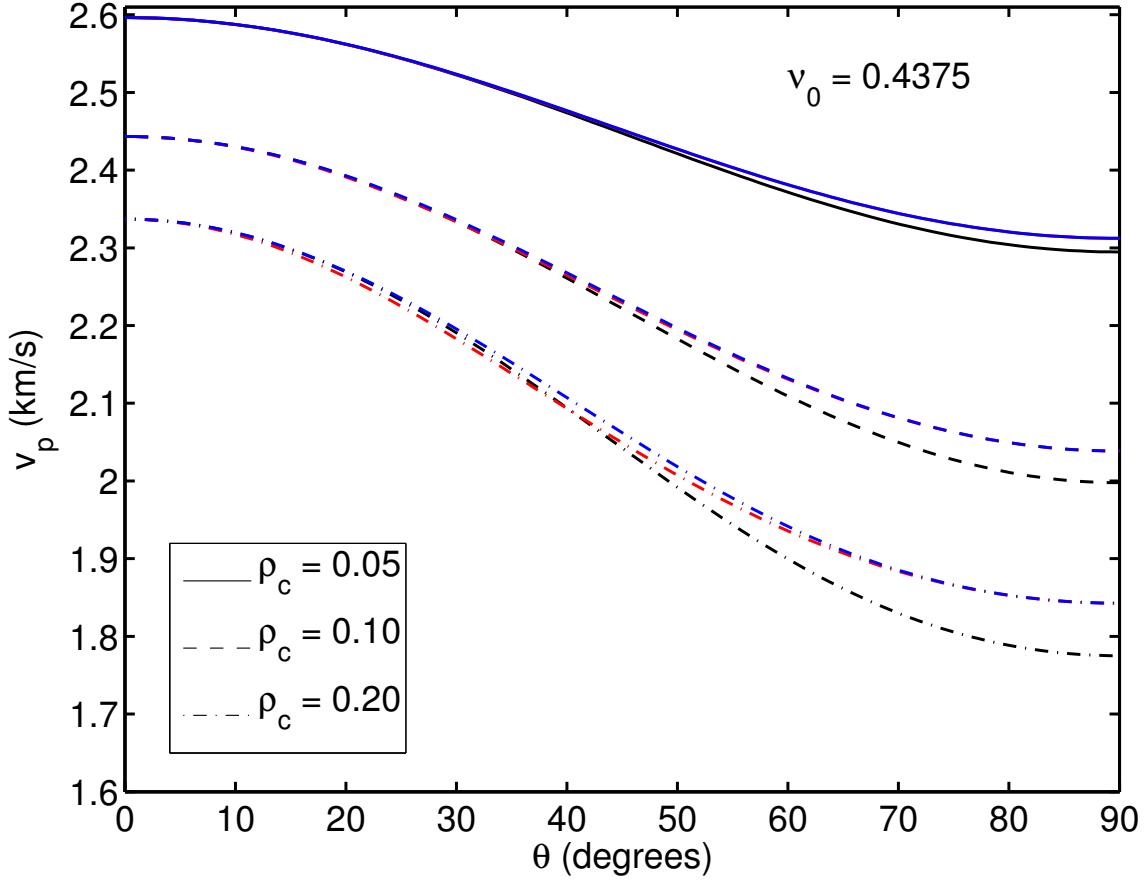


Figure 10: Same as Figure 7, for a different background medium having Poisson's ratio $\nu_0 = 0.4375$. jim1/.FIG10

Since all the angular dependence in the exact formulas is in the form of $\sin^2 \theta$ and $\cos^2 \theta$, and since $\sin^2(\frac{\pi}{2} - \theta) = \cos^2 \theta$ and vice versa, the entire procedure amounts to switching the locations of $\sin^2 \theta$ and $\cos^2 \theta$ with $\cos^2 \theta^H$ and $\sin^2 \theta^H$ everywhere in the exact expressions.

This procedure is obviously very straightforward to implement. The final results analogous to Thomsen's formulas are:

$${}^H v_p(\theta^H) / {}^H v_p(0) \simeq 1 - \frac{\epsilon}{1 + 2\epsilon} \sin^2 \theta^H - \frac{\epsilon - \delta}{1 + 2\epsilon} \frac{2 \cos^2 \theta_m^H \sin^2 \theta^H \cos^2 \theta^H}{[1 - \cos 2\theta^H \cos 2\theta_m^H]}, \quad (\text{B-5})$$

$${}^H v_{sv}(\theta^H) / {}^H v_{sv}(0) \simeq 1 + [c_{33}/c_{44}] (\epsilon - \delta) \frac{2 \cos^2 \theta_m^H \sin^2 \theta^H \cos^2 \theta^H}{[1 - \cos 2\theta^H \cos 2\theta_m^H]}. \quad (\text{B-6})$$

and

$${}^H v_{sh}(\theta^H) / {}^H v_{sh}(0) \simeq 1 - \frac{\gamma}{1 + 2\gamma} \sin^2 \theta^H. \quad (\text{B-7})$$

And the $\theta^H = 0$ velocities are: ${}^H v_p(0) = \sqrt{c_{11}/\rho} = \sqrt{c_{33}(1 + 2\epsilon)/\rho}$, ${}^H v_{sv}(0) \equiv \sqrt{c_{44}/\rho} = v_s(0)$, and ${}^H v_{sh}(0) \equiv \sqrt{c_{66}/\rho} = \sqrt{c_{44}(1 + 2\gamma)/\rho}$. Also, recall that $\cos^2 \theta_m^H = \sin^2 \theta_m^V$.

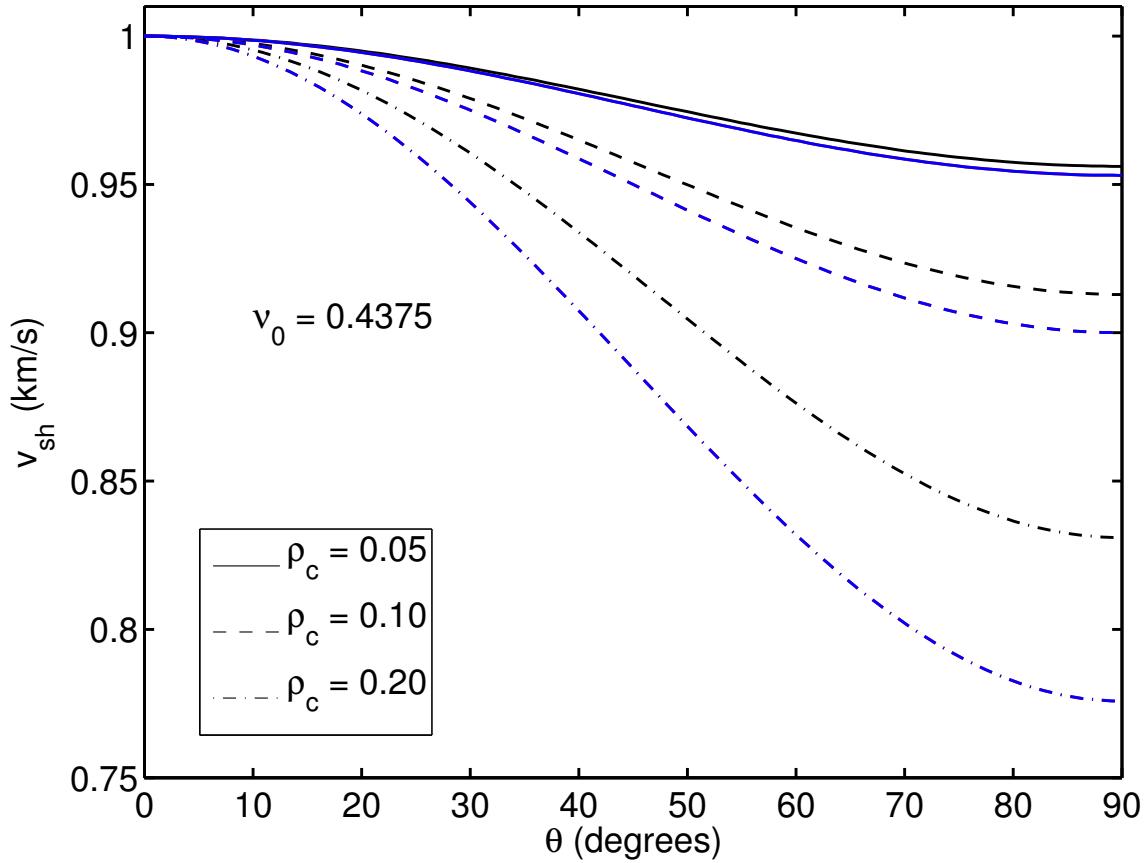


Figure 11: Same as Figure 8, but the value of $\nu_0 = 0.4375$. [jim1/. FIG11]

For azimuthal angles $\phi \neq \pm\frac{\pi}{2}$, the algorithm for computing the wave speeds is to replace $\sin^2 \theta^V$ by $\cos^2 \theta^H \sin^2 \phi$ and $\cos^2 \theta^V = 1 - \sin^2 \theta^V$ by $1 - \cos^2 \theta^H \sin^2 \phi$ in the exact formulas, and corresponding replacements in the approximate ones. Then, there is no angular dependence when $\phi = 0$ or π as our point of view then lies within the plane of the fracture itself. And, when $\phi = \pm\frac{\pi}{2}$, the above stated results for the xz -plane hold.

Wave equation reciprocity guarantees that the polarizations of the various waves are of the same types as our mental translation from VTI media to HTI media is made.

It is also worth pointing out that the labels *SH* and *SV* for the shear waves — although analogous — are, however, surely not strictly valid for the HTI case. For VTI media, the quasi-*SH*-wave really does have horizontal polarization at least at $\theta = 0$ and $\pi/2$, whereas the corresponding wave for HTI media, instead has polarization parallel (\parallel) to the fracture plane. For VTI media, the so-called quasi-*SV*-wave has its polarization in the plane of propagation, but this polarization direction is only truly vertical for $\theta = \pm\frac{\pi}{2}$, at which point its polarization is both vertical and perpendicular to the horizontal plane of symmetry. The corresponding situation for HTI media has the wave corresponding to the *SV*-wave with polarization again in the plane of propagation, but this is actually only vertical at $\theta^H = \frac{\pi}{2}$, and also parallel to the fracture plane; however, for all other angles its

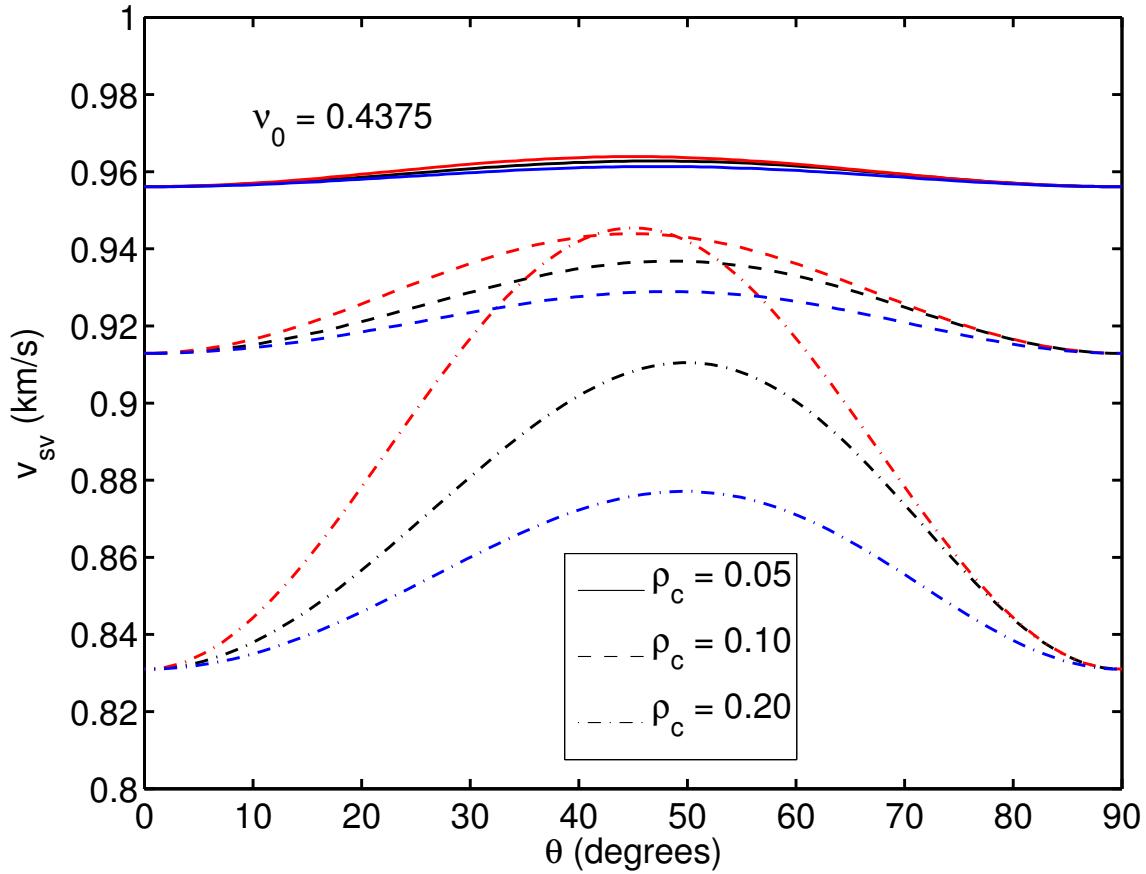


Figure 12: Same as Figure 9, but the value of $\nu_0 = 0.4375$. [jim1/.FIG12]

polarization has a component that is perpendicular (\perp) to the plane of the fractures. So a much more physically accurate naming convention for these waves would make use of the following designations:

$${}^H v_{sh}(\theta^H) \rightarrow {}^H v_{s\parallel}(\theta^H), \quad (\text{B-8})$$

for the HTI wave corresponding to the quasi-SH-wave in the VTI case, and

$${}^H v_{sv}(\theta^H) \rightarrow {}^H v_{s\perp}(\theta^H), \quad (\text{B-9})$$

for the HTI wave corresponding to the quasi-SV-wave in the VTI case. Although this notation is hereby being recommended, it will actually not be used in the main text as the current choices (as well as the various caveats) will no doubt be sufficiently familiar to most readers that it is probably not be essential to make this change in the present paper. In closing, also note that Thomsen (2002) uses the same \parallel and \perp notation for very similar purposes.

REFERENCES

- Backus, G. E., 1962, Long-wave elastic anisotropy produced by horizontal layering: *Journal of Geophysical Research*, **67**, 4427–4440.

- Berryman, J. G., 1979, Long-wave elastic anisotropy in transversely isotropic media: *Geophysics*, **44**, 896–917.
- Berryman, J. G., and V. Grechka, 2006, Random polycrystals of grains containing cracks: Model of quasistatic elastic behavior for fractured systems: *Journal of Applied Physics*, **100**, 113527.
- Dellinger, J., F. Muir, and M. Karrenbach, 1993, Anelliptic approximations for TI media: *Journal of Seismic Exploration*, **2**, 23–40.
- Fomel, S., 2004, On anelliptic approximations for qP velocities in VTI media: *Geophysical Prospecting*, **52**, 247–259.
- Fowler, P. J., 2003, Practical VTI approximations: A systematic anatomy: *Journal of Applied Geophysics*, **54**, 347–367.
- Grechka, V., and M. Kachanov, 2006a, Effective elasticity of fractured rocks: *The Leading Edge*, **25**, 152–155.
- Grechka, V., and M. Kachanov, 2006b, Effective elasticity of rocks with closely spaced and intersecting cracks: *Geophysics*, **71**, D85–D91.
- Kachanov, M., 1980, Continuum model of medium with cracks: *ASCE Journal of Engineering Mechanics*, **106**, 1039–1051.
- Mensch, T., and P. Rasolofosaon, 1997, Elastic-wave velocities in anisotropic media of arbitrary symmetry-generalization of Thomsen's parameters ϵ , δ , and γ : *Geophysical Journal International*, **128**, 43–64.
- Musgrave, M. J. P., 1954, On the propagation of elastic waves in aeolotropic media: *Proceedings of the Royal Society of London*, **A226**, 339–366.
- Musgrave, M. J. P., 1959, The propagation of elastic waves in crystals and other anisotropic media: *Reports on Progress in Physics*, **22**, 74–96.
- Musgrave, M. J. P., 2003, *Crystal Acoustics*: Acoustical Society of America, Leetsdale, Pennsylvania, Chapter 8.
- Pederson, Ø., B. Ursin, and A. Stovas, 2007, Wide-angle phase-slowness approximations: *Geophysics*, **72**, S177–S185.
- Postma, G. W., 1955, Wave propagation in a stratified medium: *Geophysics*, **20**, 780–806.
- Rüger, A., 2002, *Reflection Coefficients and Azimuthal AVO Analysis in Anisotropic Media*: Geophysical Monographs Series, Number 10, SEG, Tulsa, OK.
- Sayers, C. M., and M. Kachanov, 1991, A simple technique for finding effective elastic constants of cracked solids for arbitrary crack orientation statistics. *International Journal of Solids and Structures*, **27**, 671–680.
- Schoenberg, M., and K. Helbig, 1997, Orthorhombic media: Modeling elastic wave behavior in a vertically fractured earth: *Geophysics*, **62**, 1954–1974.
- Thomsen, L., 1986, Weak elastic anisotropy: *Geophysics*, **51**, 1954–1966.

- Thomsen, L., 2002, *Understanding Seismic Anisotropy in Exploration and Exploitation*: 2002 Distinguished Instructor Short Course, Number 5, SEG, Tulsa, OK.
- Tsvankin, I., 1997, Anisotropic parameters and P -wave velocity for orthorhombic media: *Geophysics*, **62**, 1292–1309.
- Tsvankin, I., 2005, *Seismic Signatures and Analysis of Reflection Data in Anisotropic Media*: Handbook of Geophysical Exploration, Seismic Exploration, Volume 29, Elsevier, Amsterdam.
- Zhu, T., S. H. Gray, and D. Wang, 2005, Kinematic and dynamic ray tracing in anisotropic media: Theory and application: 75th Annual International Meeting, SEG, Expanded Abstracts, 386–389.
- Zhu, T., S. H. Gray, and D. Wang, 2007, Prestack Gaussian-beam depth migration in anisotropic media: *Geophysics*, **72**, S133–S138.

Prediction error filters to enhance differences

Robert G. Clapp¹

ABSTRACT

Prediction Error Filters (PEFs) capture the covariance of a dataset. In this paper I use PEFs to quantify and highlight difference between two volumes. A series of PEFs are estimated on one volume and then applied to a second. The resulting hypercube is an indicator of where, and how much, two volumes differ.

INTRODUCTION

Developing and debugging a new approach to an old problem involves constantly comparing your ‘improved’ result to the ‘old’ approach. For 2-D volumes a movie flipping between the ‘old’ and ‘improved’ images is an effective mechanism for the well trained eye. When the dimensionality of your volume increases and/or the training of the observer decreases the human eye approach becomes less useful.

Prediction Error Filters (PEFs) (Claerbout, 1999) provide an estimate of a volume’s inverse covariance, with *stationary* statistics. By using non-stationary Prediction Error Filters (Crawley et al., 1998) or by breaking the problem into patches (Claerbout, 1992) we can characterize some level of non-stationary statistics. Schwab (1998) showed that by estimating a PEF within small patches and then applying the filter on the patch, event’s subtle features such as faults become more visible.

In this paper I use a variation on the same technique to highlight differences between volumes (‘a’ and ‘b’). I estimates PEFs within small patches on one volume ‘a’ then apply the PEF to both ‘a’ and ‘b’. I then apply a simple algebraic combination of the volumes resulting from applying the PEF to form a measure of image difference. I compare this technique to a more standard histogram matching approach and apply it on both 2-D and 3-D volumes.

METHODOLOGY

PEFs attempt to capture the inverse spectrum of the data. In the 1-D case, we could calculate a filter with the inverse spectrum by transforming into the frequency domain and then doing a sample by sample division,

$$Y(\omega) = \frac{1}{D(\omega)}, \quad (1)$$

where $Y(\omega)$ is the filter and $D(\omega)$ is the data in the Fourier domain. There are two problems with this approach. First $D(\omega)$ can be small or zero valued and $y(t)$ is not compact.

¹e-mail: bob@sep.stanford.edu,

Claerbout (1999) shows that a compact filter can be estimated by solving the least squares system

$$\mathbf{y} = (\mathbf{D}'\mathbf{D})^{-1}\mathbf{D}\mathbf{0}, \quad (2)$$

where \mathbf{y} is a filter whose zero lag is fixed at one, $\mathbf{0}$ is a vector of 0s, and \mathbf{D} is convolution with the data. In general the shape of filter is arbitrary but needs to be large enough to capture the spectrum of the wavelet and the dips present in the volume.

The residual \mathbf{r} of the estimation procedure can be calculated by convolving the filter \mathbf{Y} with the data

$$\mathbf{r} = \mathbf{Y}\mathbf{d}. \quad (3)$$

The residual will be large when the filter is not large enough to fully describe the stationary spectrum or the data is non-stationary. The left panel of Figure 1 shows the result of migrating the Marmousi dataset using a standard downward continuation based migration. The right panel shows the result of first estimating a series of PEFs in overlapping patches on the migrated image and then applying the PEFs to the migrated volume (applying equation 3). Note the areas of large residual generally correspond to unconformities and fault locations.

The next step is to apply this same series of PEFs to another image. The first question is what happens if we apply a filter estimated on volume ‘a’ to volume ‘b’ which has significantly different spatial statistics. The left panel of Figure 2 shows a simple plane wave. The right panel of Figure 2 shows the result applying the filters estimated from the Marmousi migration. The dominant feature is still the planewave. The amplitude of the residual is on average an order of magnitude higher than the residual shown in the right panel of Figure 1. If dataset ‘b’ has a spectrum close to ‘a’ we get a different result. We should see large values at both where the stationarity assumption of the PEF is invalid and at places where the covariance description of ‘a’ and ‘b’ are different. Figure 3 illustrates this point. Both the left panel of Figure 1 and 3 are calculated by a source-receiver Phase-Shift Plus Interpolation (PSPI) algorithm. The left panel of Figure 1 shows the result of using up to eight reference velocities, the left of panel of Figure 3 uses a single reference velocity at each depth step. The right panel of Figure 3 shows the result of applying equation 3 using the filter calculated from the eight velocity migration. Note that in addition to the large residual locations seen in the right panel of Figure 1, we now see additional locations. Generally the large values are at and below areas of large dip, where the first order split step correction is least accurate.

What we really would like is just the differences caused by the change in the migration algorithm. For notational convenience we will define $\mathbf{r}_{a,b}$ as the residual of applying a filter estimated on dataset ‘a’ to dataset ‘b’. Simply dividing $\mathbf{r}_{a,a}$ by $\mathbf{r}_{a,a}$ is not feasible due to the zero in $\mathbf{r}_{a,a}$. One approach to this problem is adding an epsilon term to the denominator. Another approach is smoothing. We first take the absolute value \mathbf{A} , and then smooth the resulting volume. As a result, we end up with an estimate of the fitting error \mathbf{e} ,

$$\mathbf{e} = \frac{\mathbf{SAr}_{a,b}}{\mathbf{SAr}_{a,a}} - 1 \quad (4)$$

An alternate approach is to add a scaling term that emphasizes errors where the original

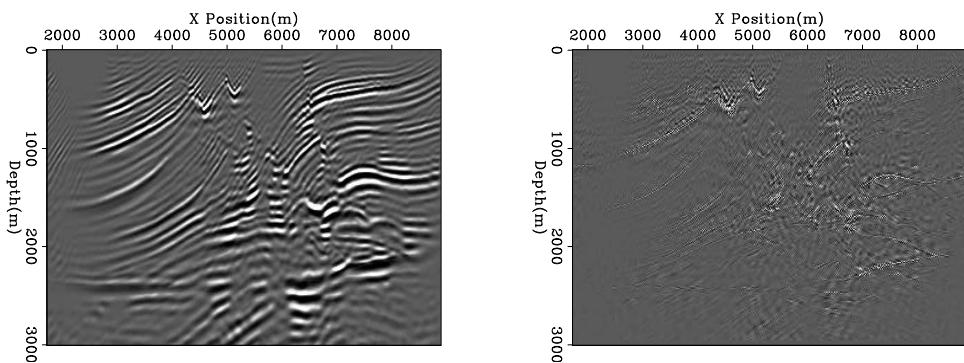


Figure 1: The left panel the result of PSPI migration of the Marmousi dataset using 8 reference velocities. The right panel shows the result of equation 3. bob2/. base

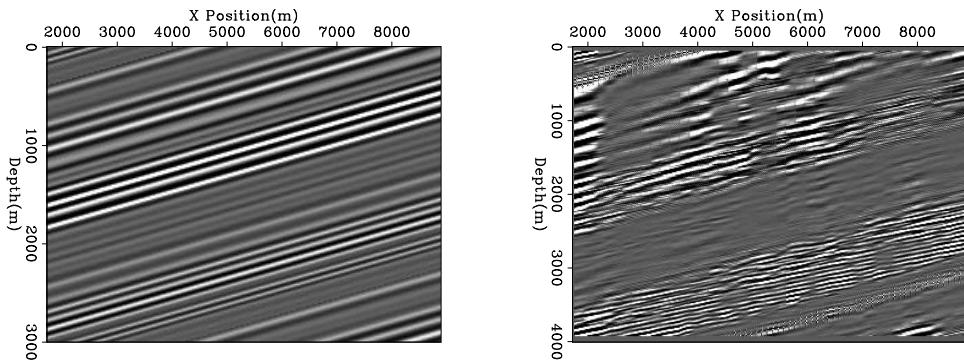


Figure 2: The left panel is a planewave. The right panel is the result of applying the filter estimated from the 8 velocity Marmousi image. bob2/. different

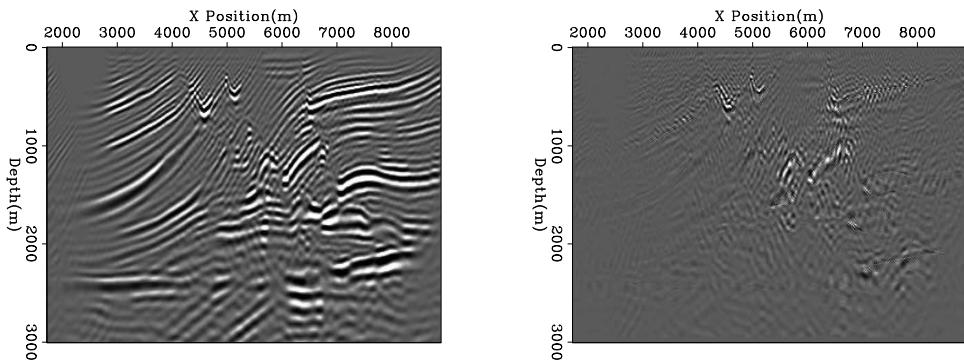


Figure 3: The left panel is the result of PSPI migration of the Marmousi dataset using one reference velocity. The right panel shows the result of equation 3 using filters calculated from the data shown in the left panel of Figure 1. bob2/. onevel

data is large. We can do this by applying a smoothed envelope function \mathbf{E} to dataset **a**,

$$\mathbf{e}_{\text{scaled}} = \mathbf{E}\mathbf{a} \left(\frac{\mathbf{SAr}_{\mathbf{a},\mathbf{b}}}{\mathbf{SAr}_{\mathbf{a},\mathbf{a}}} - 1 \right). \quad (5)$$

Figure 4 shows the result of applying equation 5 comparing the one and eight reference velocity images. Note how the differences are located at steep dips, where we would anticipate the single reference velocity approach failing.

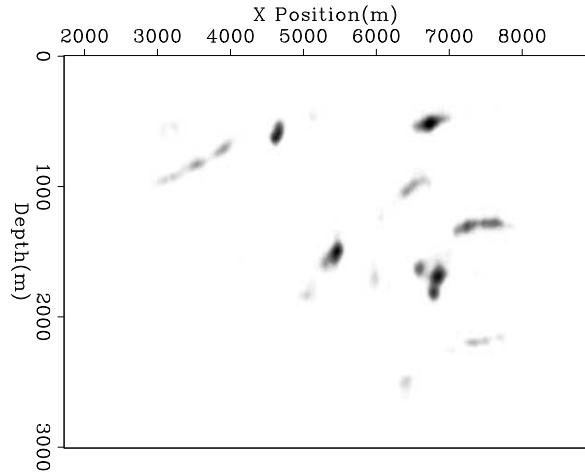


Figure 4: The result of applying equation 5 comparing the one and eight reference velocity images.

`bob2/.error-onevel`

HISTOGRAM NORMALIZATION

The previous section discussed a covariance-based approach to compare two volumes. In this section I will discuss a more tradition amplitude-based approach. Strict differences between the volumes is an option in some instances but often approaches have significantly different amplitude profiles. One solution is called histogram equalization.

My implementation of this approach is to first calculate the amplitude in volume ‘a’ at several different quantiles

$$m_a(i) = Q(a, i), \quad (6)$$

where m_a is the amplitude at a given percentile i of volume a using the quantile function Q . The vector m_a is basically a discrete version of the data’s cumulative distribution function (CDF). I then found the amplitude in volume ‘b’ at the same quantiles producing the amplitude map m_b . Figure 5 shows the cumulative distribution function for the one and eight velocity PSPI migrations shown in the previous section. Note how the two curves are similar, diverging only at their edges. Finally I looped through volume ‘b’, for each sample I found its approximate quantile by finding the samples of m_b that contained the value and performing linear interpolation. I was able to remap into the amplitude profile of ‘a’ using m_a . Figure 6 shows the difference between the one and eight reference velocity PSPI migration after histogram normalization. Note the image seems to emphasize the major reflectors of the image rather than the differences.

PRECISION

Fu et al. (2007) implements some of the core algorithms in reverse time migration and

Figure 5: The solid curve is the CDF for the eight velocity PSPI migration shown in Figure 1, the dashed curve shows the CDF using one velocity (shown in Figure 3). Note how the two curves are similar diverging only at their edges. bob2/. cdf

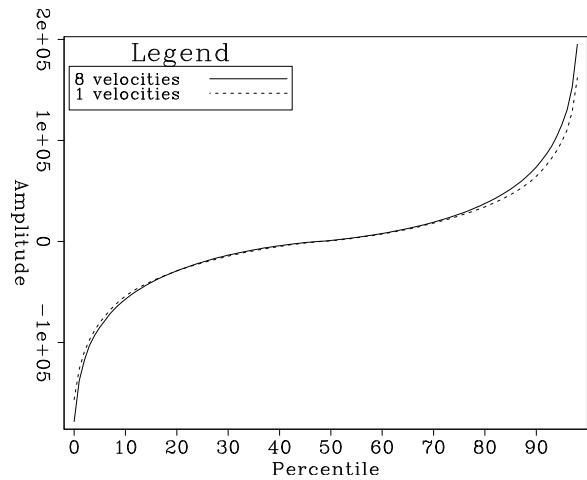
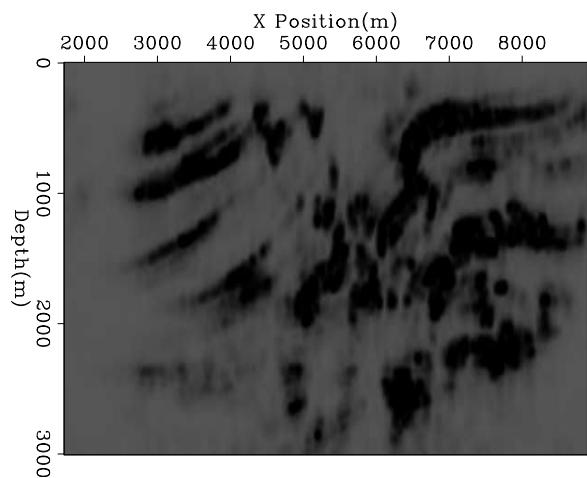


Figure 6: The difference between the one and eight reference velocity PSPI migration after histogram normalization. bob2/. onevel-diff



downward continuation based migration on a FPGA. In many situations compute speed can be traded for precision on a FPGA. Clapp (2007) showed that migration is well suited for reduced precision given the summation implied by the process.

Fu et al. (2007) uses a non-linear, computer driven scheme to test whether a given number representation effects the computed result. In order to use this scheme, some test of whether a result changed in a meaningful way had to be designed. The methodology discussed above offers that potential. By summing up the errors resulting from applying equation 5 a single number that represents how well the covariance of an image has been preserved can obtained.

To test this methodology I applied the same precision limiting scheme described in Clapp (2007). The rows of Figure 7 shows the result of limiting the precision of the FK, FFT, and FX portions of the downward continuation process to 3,5, and 9 bits. The left panel is the zero subsurface offset image in each case. The center panel the error calculation using the PEF method. The right panel shows the result using the histogram matching scheme. In each case the migration result is compared against the full precision image. Note how the errors drastically decreases between 3 and 5 bits. Figure 8 show the total error for 3-9 bits. In this case, both methods seem to be an effective mechanism for testing accuracy. Note how the curve dramatically decreases between 3 and 6 bits then remains relatively constant for the PEF method scheme. The elbow in each curve represents the best tradeoff between bit precision and accuracy.

REFERENCE VELOCITIES

The bit precision limiting scheme used in the previous section should either have a relatively uniform, or somewhat unpredictable, amplitude effect. A more interesting test is to apply both difference detecting methodologies on a problem where the locations of the differences are well known.

PSPI migration accurately handles non-overturning waves when the velocity in a given depth layer is constant. When velocity varies laterally the accuracy decreases as a function of propagation angle. The Marmousi example has significant lateral velocity variation and many steeply dipping events. As a result the image quality varies with the number of reference velocities that are used. Figure 9 shows the effect of changing the number of reference velocities used in the migration. The rows represent migrating with a PSPI migration using one, three, and seven reference velocities. The left column is the migrated image, the center panel is the result of using the PEF method for detecting differences. The right panel shows the result of using the histogram matching. In this case, the PEF approach proves to be better at identifying differences in the images. Note how the PEF approach highlights the steeply dipping features, where the PSPI approximation breaks down. With increasing number of reference velocities only steeper dipping features are displayed. The histogram approach seems to simply highlight the major features in the image, showing limited preference towards large dips. Figure 10 shows the total differences as a function of the number of reference velocities. Note how the effect of increasing the number of reference velocities is much more pronounced in the PEF case.

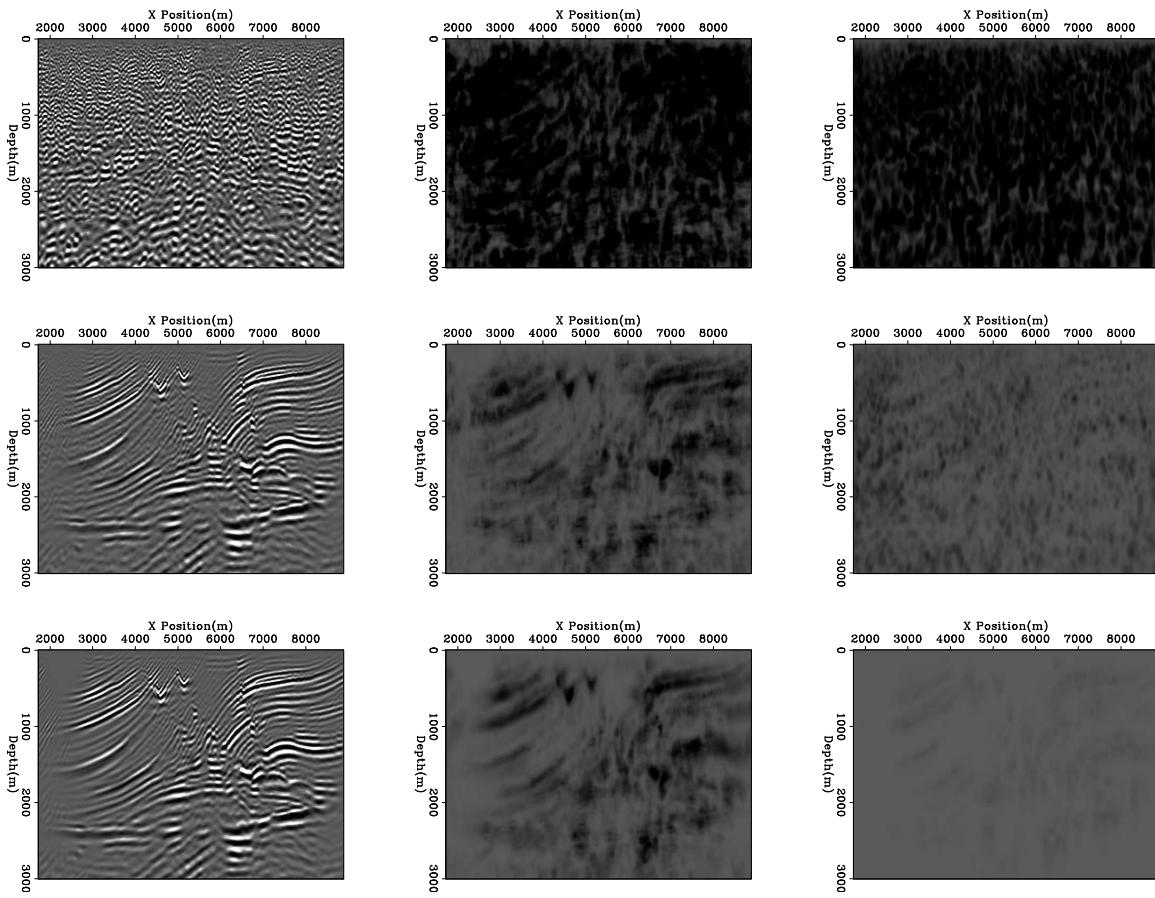


Figure 7: The shows the result of limiting the precision of the FK and FX portions of the downward continuation process to 2,4, and 9 bits. The left panel is the zero subsurface offset image in each case. The center (PEF) and right (histogram based) panel are the error calculation. In each case the migration result is compared; against the full precision image.
bob2/. big-p

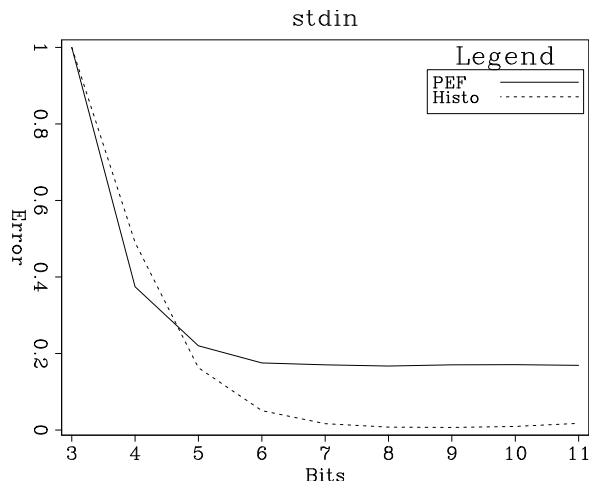


Figure 8: Error vs. bit representation for PSPI migration. Note the elbow in the curve represents the best tradeoff between bit precision and accuracy.
bob2/. p-graph

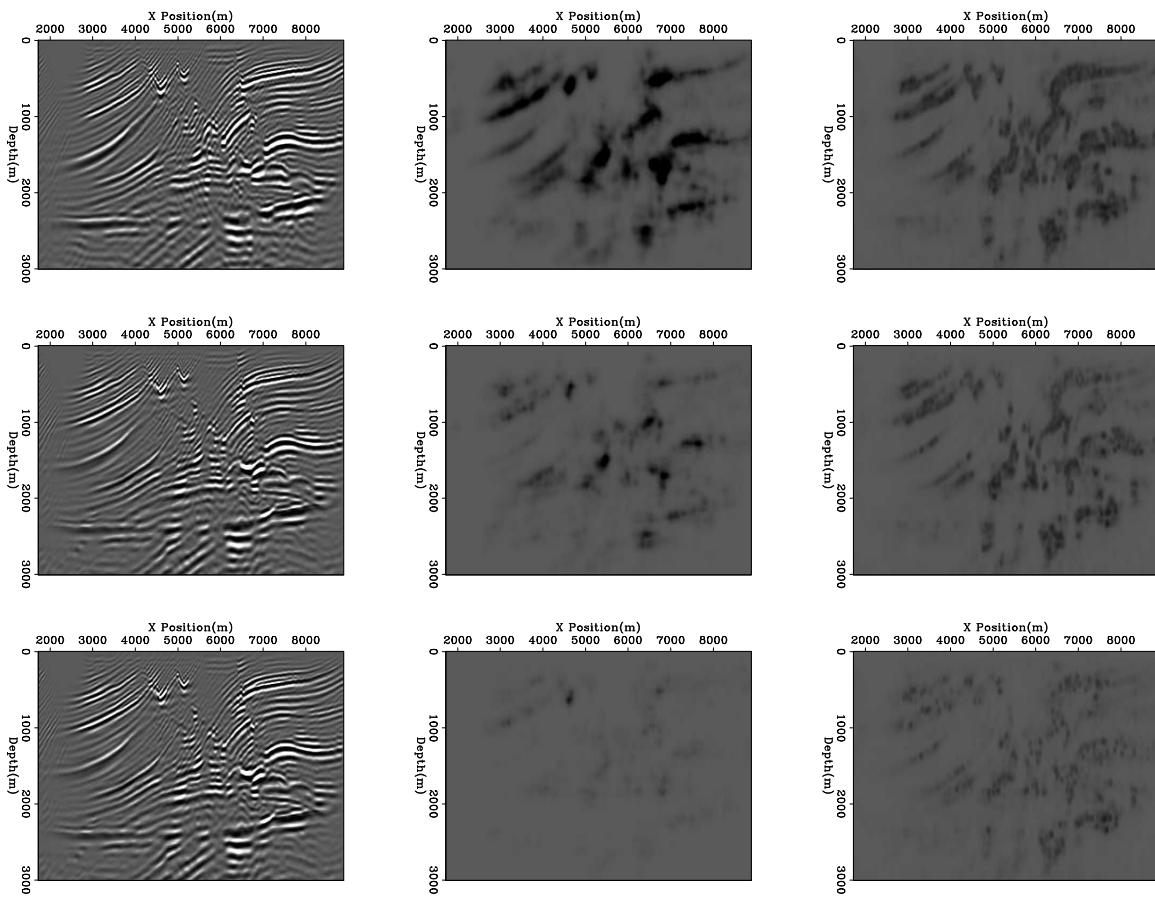
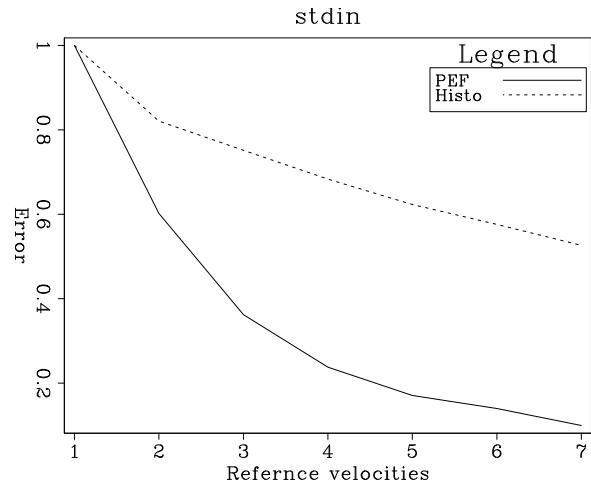


Figure 9: The rows represent migrating with a PSPI migration using one, three, and seven reference velocities. The left column is the migrated image, the center panel is the result of using the PEF method for detecting differences. The right panel shows the result of using the histogram matching. bob2/. big-v

Figure 10: A graph of the differences detected by the PEF (solid line) and histogram matching (dashed line) scheme. bob2/. v-graph



PLANEWAVE MIGRATION

As demonstrated in the last section the PEF method has the ability to highlight subtle differences in images. In this section, I apply the methodology to 3-D plane-wave migrated cubes. The first uses a conventional scheme, the second uses the tilted-coordinate approach described in Shan and Biondi (2007). I broke the image into 30x30x30 patches and used a 3-D PEF that was 8x3x3 in size.

Figure 11 shows three orthogonal slices through a volume migrated with a conventional planewave migration algorithm. Note the salt structure top is poorly delineated in the inline section. Figure 12 shows the same three orthogonal slices using a tilted planewave migration scheme. Note how salt top is significantly better delineated. Figure 13 shows the result of applying the PEF scheme compared to the two 3-D migrated cubes. In the depth slice, the salt structure is highlighted. In addition, several faults become evident. The inline panel and crossline panel highlight the salt top. A subtle fault feature is visible in the crossline and a high-spatial frequency event is evident in the inline. Figure 14 shows the result of using the histogram methodology. The method proves generally ineffective in highlighting any differences other than the salt top.

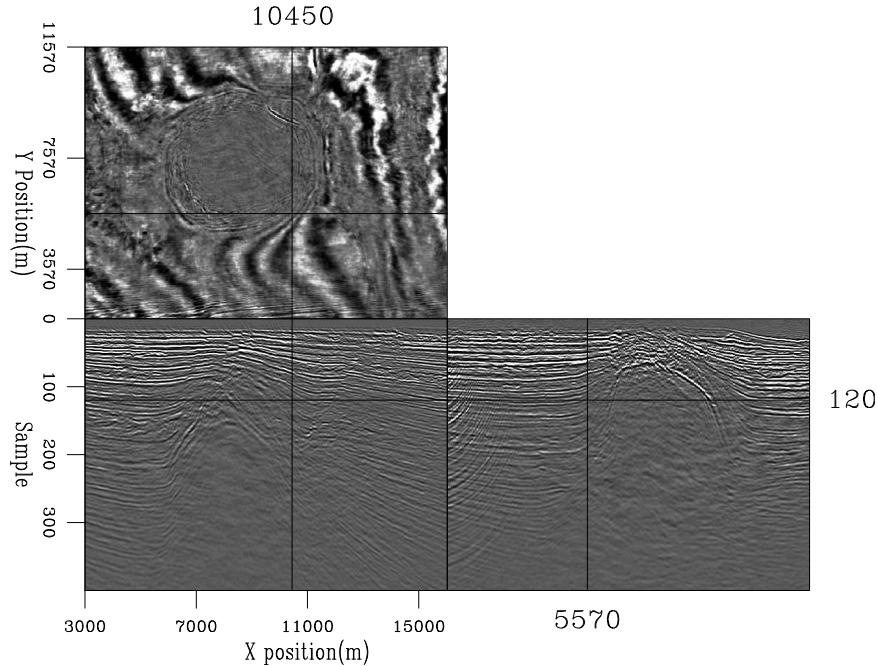


Figure 11: Three orthogonal slices through a migrated cube using a conventional planewave migration algorithm. `bob2/. notiltsame`

CONCLUSIONS

Two schemes, one amplitude based, one covariance based, are described that automatically compare two seismic images. The first scheme uses histogram normalization to equalize the amplitude of the full precision and reduced precision volumes. The second approach

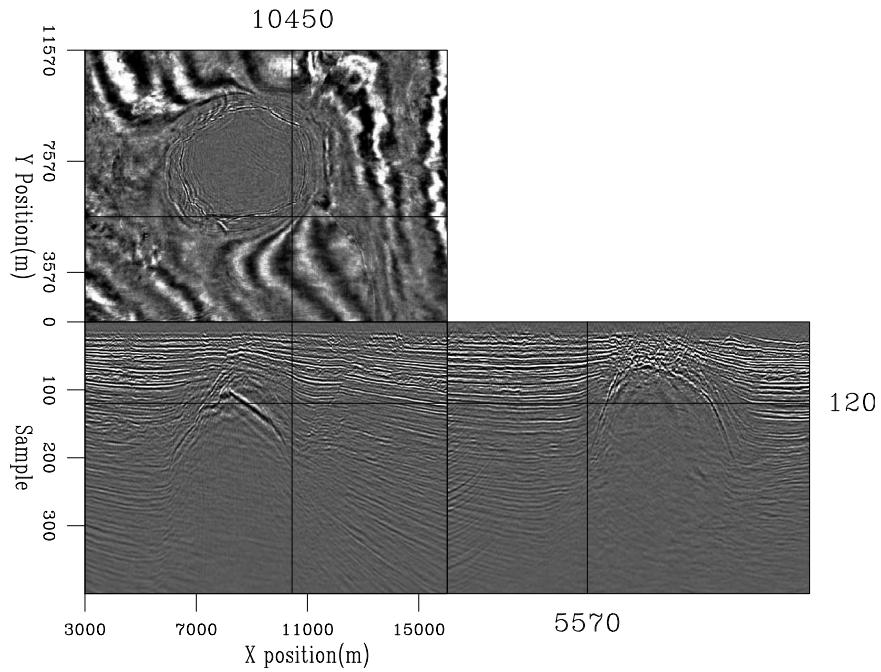


Figure 12: The same slices seen in Figure 11, now through a volume migrated with a tilted-coordinate planewave migration algorithm. `bob2/. tilt`

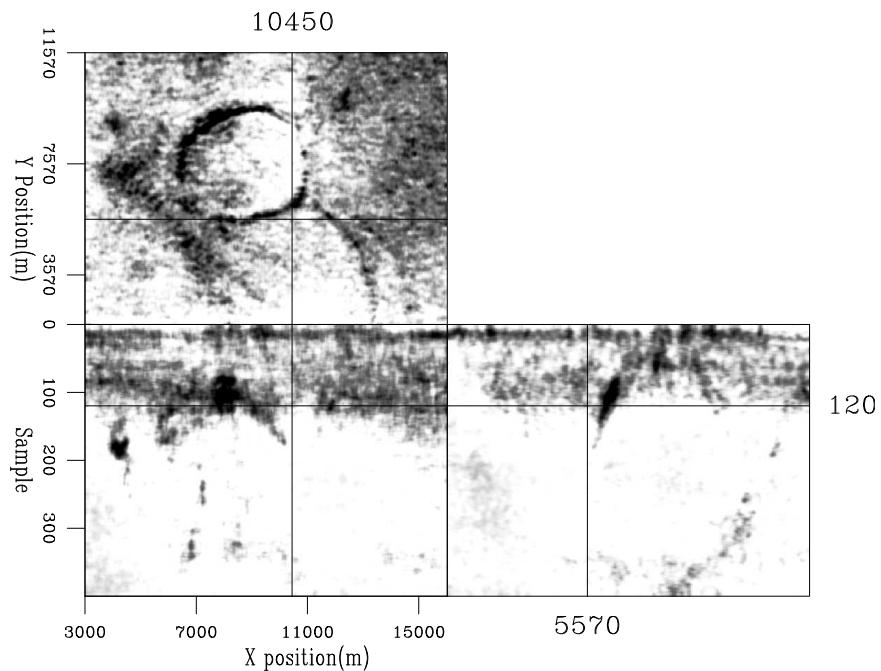


Figure 13: The same slices seen in Figure 11- 12 through a volume created by applying equation 5 to the volumes shown in the previous two figures. Note the highlighting of the salt reflection along with several faults. `bob2/. comp-good`

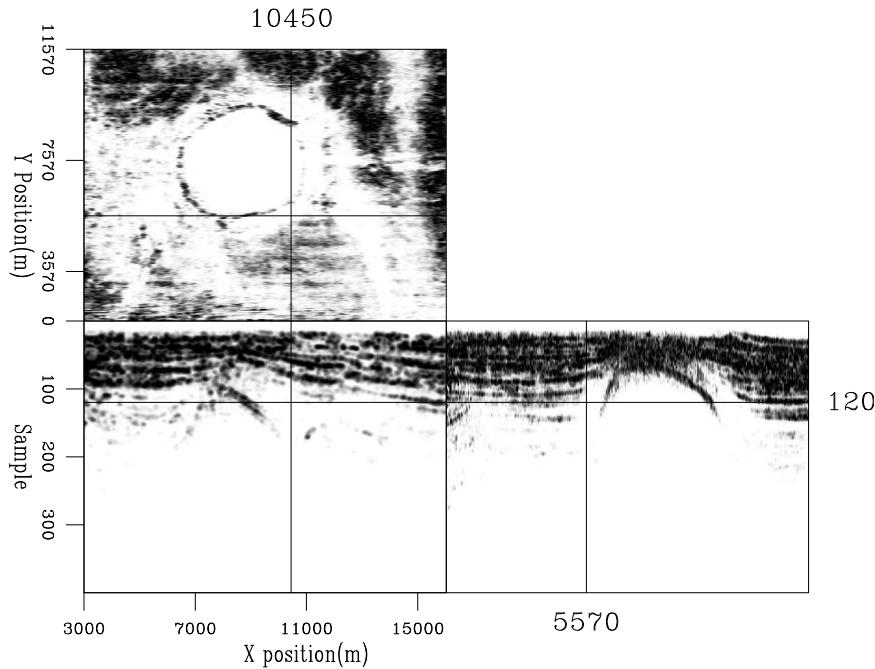


Figure 14: The same slices seen in Figure 11- 13 through a volume created by applying the histogram matching approach. Note how the salt top is visible but the fault features seen in Figure 14 are not. bob2/.comp-histo

compares the covariance of two volumes within small patches. Both methods prove effective, with the covariance based approach showing more consistent behavior.

ACKNOWLEDGMENTS

I would like to thank ExxonMobil for the dataset. I would like to acknowledge Oskar Mencer who insisted that I do this work. Biondo Biondi suggested that histogram normalization would be worth trying.

REFERENCES

- Claerbout, J., 1999, Geophysical estimation by example: Environmental soundings image enhancement: Stanford Exploration Project.
- Claerbout, J. F., 1992, Nonstationarity and conjugacy: Utilities for data patch work: SEP-Report, **73**, 391–400.
- Clapp, R. G., 2007, Moveout analysis with flattening: SEP-Report, **129**, 149–158.
- Crawley, S., R. Clapp, and J. Claerbout, 1998, Decon and interpolation with nonstationary filters: SEP-Report, **97**, 183–192.
- Fu, H., W. Osbourne, R. G. Clapp, and O. Mencer, 2007, Accelerating seismic computations using customized number representations on fpgas: SEP-Report, **131**.
- Schwab, M., 1998, Enhancement of discontinuities in seismic 3-D images using a Java estimation library: PhD thesis, Stanford University.

Shan, G. and B. Biondi, 2007, Angle domain common image gathers for steep reflectors:
SEP-Report, **131**.

Accelerating seismic computations using customized number representations on FPGAs

Haohuan Fu, William Osbourne, Robert G. Clapp, and Oskar Mencer¹

ABSTRACT

Field Programmable Gate Arrays (FPGA) offer significant potential speedups over conventional microprocessors for some applications. For downward continued migration, complex math and Fast Fourier Transforms (FFT) are the dominant cost. Convolution is the dominant cost for reverse time migration. We implement these core algorithms on a FPGA and show speedups ranging from 5 to 15, including the transfer time to and from the processors. We consider methods to further speed up these migration algorithms.

INTRODUCTION

Seismic imaging is the most computationally demanding technology of the oil and gas sector. Downward continued based migration (Gazdag and Sguazzero, 1985) is the most prevalent high-end imaging technique today, and reverse time migration appears to be one of the dominant imaging techniques for the future.

Downward continued migration comes in various flavors including Common Azimuth Migration (Biondi and Palacharla, 1996), shot profile migration, source-receiver migration, planewave or delayed shot migration, and narrow azimuth migrations. The different techniques have varying cost profiles but all share two meaningful computational bottlenecks: transforming to-and-from the wavenumber domain (FFT) and applying the single square root (SSR) or double square root (DSR) condition (complex exponentials).

The cost of explicit space-domain 3-D reverse time migration is dominated by the cost of continuing the source and receiver wavefield a given time step. To progress the wavefield a given time step requires applying a 3-D stencil that can range in size from 7 to 31 points depending on the finite-difference approximation that is chosen.

In this paper, we describe the implementation of DSR equation and the kernel of 3-D acoustic modeling on a FPGA. We begin by giving a basic background of FPGAs. We describe the FPGA programming environment and our methodology for determining the correct trade off between precision and speed. We then describe the implementation procedure for both algorithms. We conclude by discussing potential additional speedup opportunities of both reverse time and wavefield continuation based migration.

PROJECT OVERVIEW

As a first step it is useful to begin by defining some FPGA terminology.

¹e-mail: bob@sep.stanford.edu,

Block RAM Small memory elements (store up to 512 single-precision floats) located within the FPGA. The Xilinx Virtex 4 FX100 FPGA contains 376. All BRAMs can be read and written in parallel and combined into larger memories, leading to very high internal bandwidth.

FIFO First-In-First-Out Memory queue built from Block RAMs. FIFOs exploit temporal locality in data streams.

Slice A unit of area on Xilinx FPGAs. Each slice contains 2 (on most current FPGAs) lookup tables (LUTs), the basic compute unit on an FPGA, each LUT implements any 4-input 1-output logical function. We connect LUTs to implement arithmetic and control logic.

PCI Express x8 State-of-the-art bus for FPGA acceleration. 4000MB/sec peak bandwidth.

FPGAs are Complementary metaloxidesemiconductor (CMOS) technology-based chips containing logic which can be configured to any sequential circuit and a limited number of memory elements including RAMs and registers. The price of reconfigurability is a 10x slower dynamic clock frequency compared to modern processors. We exploit the parallelism and ability to use custom number representations to overcome the lower clock frequency and obtain a higher performance.

The long term goal of this project is to speed up key seismic imaging application by at least a factor of 10x over conventional multi-core hardware.

Downward continued based migration

For downward continued based migration there are four potential computational bottlenecks that vary depending on the flavor of the downward continuation algorithm. In many cases the dominant cost is the FFT step. The dimensionality of the FFT varies from 1-D (tilted plane-wave migration (Shan and Biondi, 2007)) to 4-D (narrow azimuth migration (?)). The FFT cost is often dominant due its $n \log(n)$ cost ratio, n being the number of points in the transform, and the non-cache friendly nature of multi-dimensional FFTs. The FK step, which involves evaluating a square root function and performing complex exponential is a second potential bottleneck. The high operational count per sample can eat up significant cycles. The FX step, which involves a complex exponential, or sine/cosine multiplication, has a similar, but computationally less demanding, profile. Creating subsurface offset gathers for shot profile or plane-wave migration, particularly 3D subsurface offset gathers, can be an overwhelming cost. The large op-count per sample and the non-cache friendly usage can be problematic. Finally, for finite difference based schemes a significant convolution cost is involved.

Last summer the focus was on speeding up 1 and 2-D FFTs. Speedup ranged from 8x-16x depending on the required data precision. (Pell and Clapp, 2007) demonstrated that the subsurface offset calculation can be sped up by a factor of 20x-40x. This summer, the focus was speeding up the FK step by implementing both a table lookup and complex exponential on the FPGA.

Reverse time migration

The primary bottleneck of reverse time migration is applying the finite-difference stencil. In addition to the large operation count (5 to 31 samples per cell) the access pattern has poor cache behavior for real size problems. Beyond applying the 3-D stencil the next major cost is implementing damping boundary conditions. Methods such as Perfectly Matched Layers (PML) can be costly (Berenger, 1994). Finally, if you want to use reverse time migration for velocity analysis, subsurface offset gathers need to be generated. The same cost profile that exists in downward continued based migration exists for reverse time migration.

Last summer the focus was on implementing the 2-D elastic modeling convolutional kernel. We achieved a speed up of 8-16x, again depending on data precision. This summer we concentrated on 3-D acoustic modeling kernel.

BACKGROUND

Number Representation

Precision and range are key resources to be traded off against the performance of a computation. We looked at three different types of number representation: fixed-point, floating-point and logarithmic. Consider the case when U_i is represented as a fixed-point number, with an integer part I which is a bits in length, and a fraction part F which is b bits in length.

$i_{a-1} \dots i_2 i_1 i_0$	$f_0 f_1 f_2 \dots f_{b-1}$
-----------------------------	-----------------------------

The integer bit-width, which represents the dynamic range of the number, is calculated according to equation (1):

$$k \geq \lceil \log_2(|\max(U_i) - \min(U_i)|) \rceil \quad (1)$$

For the floating-point number system, let U_i represent a floating-point number $(-1)^S \cdot M \cdot 2^E$, where S is the sign bit, M is the mantissa with a bit-width of m bits, and E is the exponent with a bit-width of e bits.

S	$i_0 i_1 i_2 \dots i_{m-1}$	$f_{e-1} \dots f_2 f_1 f_0$
-----	-----------------------------	-----------------------------

The value of the mantissa M is expressed as:

$$M = \sum_{i=0}^{m-1} a_i 2^{-i} \quad (2)$$

where $a_i \in \{0, 1\}$.

It is possible to relate the bit-width m of the mantissa of the node to the error when representing the mantissa by a finite bit-width $Errf_{lt}$, as follows:

$$Errf_{lt}(m) = \begin{cases} 2^{-m} \times 2^E & \text{if round-to-nearest} \\ 2^{-(m-1)} \times 2^E & \text{if truncation} \end{cases}$$

where E is the value of the exponent at the node.

Since there is no standard to encode logarithmic numbers, in this report we use a fixed-point format to store the logarithmic value.

A stream Compiler (ASC)

We use our object-oriented ASC FPGA programming tool to develop a range of different solutions. ASC, A Stream Compiler, was developed following research at Stanford University and Bell Labs, and is now commercialized by Maxeler Technologies. ASC enables the use of FPGAs as highly parallel stream processors. ASC is a C-like programming environment for FPGAs. ASC code makes use of C++ syntax and ASC semantics which allow the user to program on the architecture-level, the arithmetic-level and the gate-level. In contrast to other methodologies, ASC provides the productivity of high-level hardware design tools and the performance of low-level optimized hardware design. On the arithmetic level, PAM-Blox II provides an interface for custom arithmetic optimization. On the higher level, ASC provides types and operators to enable research on custom data representation and arithmetic. ASC hardware types are HWint, HWfix and HWfloat. Utilizing the data-types we build libraries such as a function evaluation library or develop special circuits to solve particular computational problems such as graph algorithms. A simple example of an ASC description for a stream architecture that doubles the input and adds '55' looks as follows:

```
%\lstset{language=ASC}
%\begin{figure}[ht]
%\begin{lstlisting}
// ASC code starts here
STREAM_START;

// Hardware Variable Declarations
HWint in(IN);
HWint out(OUT);
HWint tmp(TMP);

STREAM_LOOP(16);
tmp = (in << 1) + 55;
out = tmp;

// ASC code ends here
STREAM_END;
%\end{lstlisting}
%\caption{A simple ASC example} \label{fig:asc_example}
%\end{figure}
```

The ASC code segment shows HWint variables and the familiar C syntax for equations and assignments. Compiling this program with ‘gcc’ and running it creates a net-list which can be transformed into a configuration bitstream for an FPGA.

CUSTOMIZED NUMBER REPRESENTATIONS

FPGA-based implementations have the advantage over current software-based implementations of being able to use customizable number representations in their circuit designs. On a software platform, users are usually constrained to a few fixed number representations, such as 32/64-bit integers and single/double-precision floating-point; while the reconfigurable logic and connections on an FPGA enables the users to explore various number formats with arbitrary bit-widths. Furthermore, users are also able to design the arithmetic operations for these customized number representations, can thereby providing a highly customized solution for a given problem.

In general, to provide a customized number representation for an application, we have three questions to solve:

- Which number representation should we use?

There are existing FPGA applications using fixed-point, floating-point and logarithmic numbers. Fixed-point has simple arithmetic implementations, while floating-point and logarithmic number systems (LNS) provide a wide representation range.

- How do you determine the bit-width of the variables in the design?

This problem is generally referred to as bit-width or word-length optimization (Lee et al., 2006; Constantinides et al., 2001). We can further divide this into two different parts: **range analysis** considers the problem of ensuring that a given variable inside a design has a sufficient number of bits to represent the range of the numbers; while in **precision analysis**, the objective is to find the minimum number of precision bits for the variables in the design such that the output precision requirements of the design are met.

- How do you implement the arithmetic operations for the customized number representations?

The arithmetic operations of each number system are quite different. For instance, in LNS, multiplication, division and exponential operations become as simple as addition or shift operations, while addition and subtraction become non-linear functions to approximate. The arithmetic operations of regular data formats, such as fixed-point and floating-point, also have different algorithms with different design characteristics. Evaluation of elementary functions also plays a large part in seismic applications (trigonometric and exponential functions). Different evaluation methods and configurations can be used to produce evaluation units with different accuracies and performance.

This section discusses our approaches to finding a solution to these three problems. The approaches are partly based on our previous work on bit-width optimization (Lee et al., 2006) and comparison between different number representations (Fu et al., 2006, 2007). As shown in Fig. 1, we manually partition the Fortran program into two parts: one part will run in software and the other in hardware (target code). The first step is to profile the target code to acquire information about the distribution of values that each variable can take. In the second step, based on the range information, we map the Fortran code into

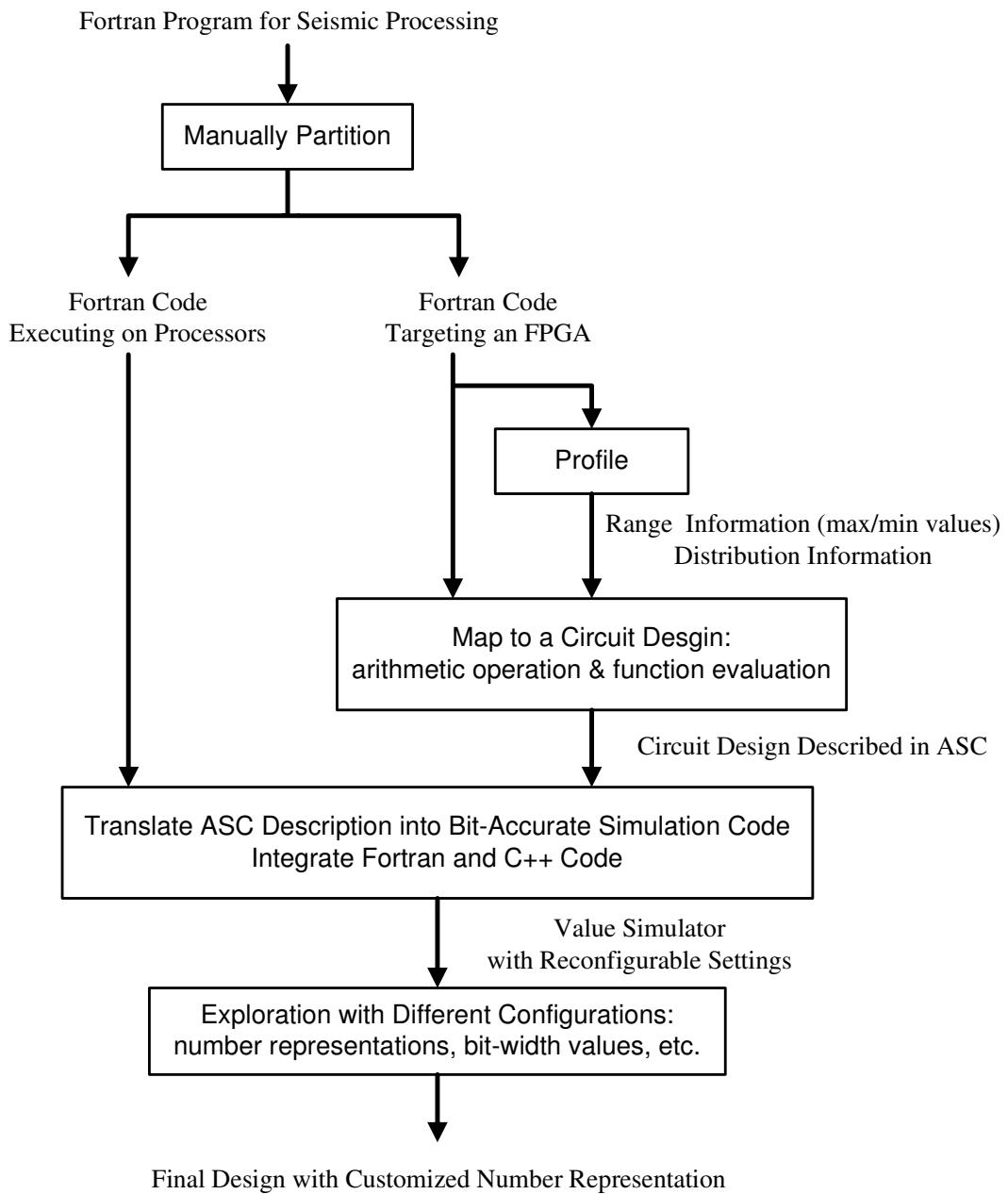


Figure 1: Basic steps to achieve a hardware design with customized number representations.
 bob1/. cus-rep-wf

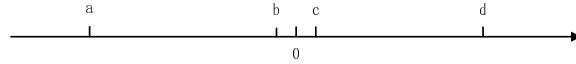
a hardware design described in ASC format, which includes implementation of arithmetic operations and function evaluation. In the third step, the ASC description is translated into bit-accurate simulation code, and merged into the original Fortran program to provide a value simulator for the original application. Using this value simulator, explorations can be performed with configurable settings such as different number representations, different bit-widths and different arithmetic algorithms. Based on the exploration results, we can determine the optimal number format for this application with regards to certain characteristics such as circuit area and performance.

Profiling

In the profiling stage, the major objective is to collect range and distribution information for the variables. The idea of our approach is to instrument every target variable in the code, adding function calls to initialize data structures for recording range information and to modify the recorded information when the variable value changes.

For the range information of the target variables (variables to map into the circuit design), we keep a record of four specific points on the axis, shown in figure 2.

Figure 2: bob1/. axis



The points a and d presents the values far away from zero, i.e., the maximum absolute values that need to be represented. Based their values, the integer bit-width of fixed-point numbers can be determined. Points b and c represent the values close the zero, i.e., the minimum absolute values that need to be represented. Using the minimum and maximum values, the exponent bit-width of floating-point numbers and integer bit-width of logarithmic numbers can be determined.

For the distribution information of each target variable, we keep a number of buckets to store the frequency of values at different intervals. Fig. 3 shows the distribution information recorded for the real part of variable `wf1d` (a complex variable). In each interval, the frequency of positive and negative values are recorded separately. The results show that, for the real part of variable `wf1d`, in each interval, the frequencies of positive and negative values are quite similar, and the major distribution of the values falls into the range 10^{-1} to 10^4 .

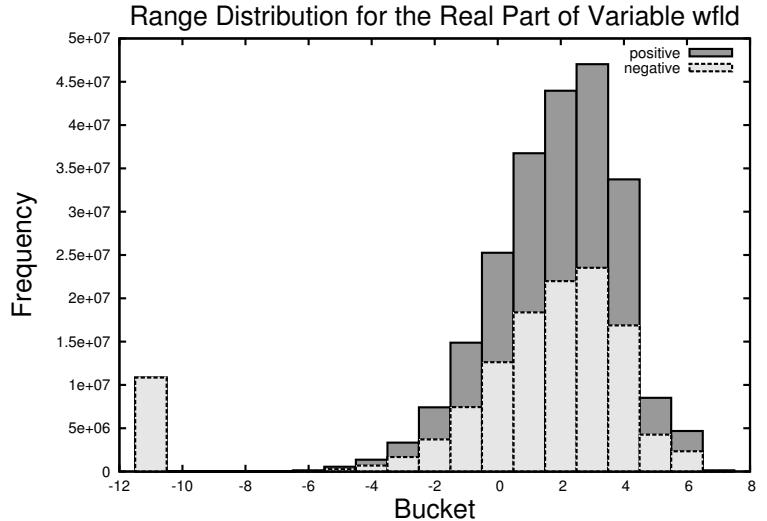
The distribution information provides a rough metric for the users to make an initial guess about which number representations to use. If the values of the variables cover a wide range, floating-point and logarithmic number formats are usually more suitable. Otherwise, fixed-point numbers shall be enough to handle the range.

Circuit Design: Basic Arithmetic & Elementary Function Evaluation

After profiling range information for the variables in the target code, the second step is to map the code into a circuit design described in ASC.

Figure 3: Range distribution of the real part of variable ‘wfld’. The left-most bucket with index= -11 is reserved for zero values. The other buckets with index= x store the values in the range $(10^{x-1} - 10^x)$.

`bob1/. range-wfld`



As a high-level FPGA programming language, ASC provides hardware data-types, such as HWint, HWfix and HWfloat. Users can specify the bit-width values for hardware variables, and ASC automatically generates corresponding arithmetic units for the specified bit-widths. It also provides configurable options to specify different optimization modes, such as AREA, LATENCY and THROUGHPUT. In the THROUGHPUT optimization mode, ASC automatically generates a fully-pipelined circuit. These features make ASC an ideal hardware compilation tool to re-target a piece of software code onto the FPGA hardware platform.

ASC does not have inherent support for LNS numbers. This part is covered by our previous work on the LNS arithmetic library generator (Fu et al., 2007), which produces optimized LNS arithmetic units with customizable bit-width values, also in ASC format.

Thus, with support for fixed-point, floating-point and LNS arithmetic operations, the target Fortran code can be transformed into ASC C++ code in a straightforward manner. We also have interfaces provided by ASC and the LNS library generator to modify the internal settings of these arithmetic units.

In seismic applications, evaluation of elementary functions takes a large part in the application. For instance, in the first piece of target code we try to accelerate, the ‘complex exponential function’. A large part of the computation is to evaluate the square root and sine/cosine functions. To map these functions into efficient units on the FPGA board, we use a table-based uniform polynomial approximation approach, based on Dong-U Lee’s work on optimizing hardware function evaluation (Lee et al., 2005). The evaluation of the two functions can be divided into three different phases (Muller, 1997):

- Range Reduction: reduce the range of the input variable x into a small interval that is convenient for the evaluation procedure. The reduction can be multiplicative (e.g. $x' = x/2^{2n}$ for square root function) or additive (e.g. $x' = x - 2\pi n$).
- Function Evaluation: approximate the value of the function using a polynomial within the small interval.

- Range Reconstructions: map the value of the function in the small interval back into the full range of the input variable x .

To keep the whole unit small and efficient, we use degree-one polynomial so that only one multiplication and one addition are needed to produce the evaluation result. Meanwhile, to preserve the approximation error at a small scale, the reduced evaluation range is divided into uniform segments. Each segment is approximated with a degree-one polynomial, using the minimax algorithm. In the case of the ‘complex exponential’ code segment, the square root function is approximated with 384 segments in the range of $[0.25, 1]$ with a maximum approximation error of 4.74×10^{-7} , while the sine and cosine functions are approximated with 512 segments in the range of $[0, 2]$ with a maximum approximation error of 9.54×10^{-7} .

Bit-accurate Value Simulator

As discussed earlier, based on the range information, we are able to determine the integer bit-width of fixed-point and LNS numbers and the exponent bit-width of floating-point numbers. The remaining bit-widths, such as the fractional bit-width of fixed-point and LNS numbers, and the mantissa bit-width of floating-point numbers, are predominantly related to the precision of the calculation in order to find out the minimum acceptable values for these precision bit-widths, we need a mechanism to determine whether a given set of bit-width values produce satisfactory results for the application.

In our previous work on function evaluation or other arithmetic designs, we set a requirement of the absolute error of the whole calculation, and use a conservative error model to determine whether the current bit-width values meet the requirement (Lee et al., 2006). However, a specified requirement for absolute error does not work for seismic processing. To find out whether the current configuration of precision bit-width is accurate enough, we need to run the whole program to produce the resulting image, to find out whether the image contains the correct pattern information. Thus, to enable exploration of different bit-width values, a value simulator for different number representations is needed to provide bit-accurate simulation results for the hardware designs.

With the requirement to produce bit-accurate results as the corresponding hardware design, the simulator also needs to be efficiently implemented, as we need to run the whole application (which takes days using the whole input dataset) to produce the image.

In our approach, the simulator works with ASC format C++ code. It re-implements the hardware data-types, such as HWfix, HWfloat and HWlns, and overloads their arithmetic operators with the corresponding simulation code.

For HWfix variables, the value is stored in a 64-bit signed integer, while another integer is used to record the fractional point. The basic arithmetic operations are mapped into shifts and arithmetic operations of the 64-bit integers.

For HWfloat variables, the value is stored in a 64-bit double-precision floating-point number, with two other integers used to record the exponent and mantissa bit-width. To keep the simulation simple and fast, the arithmetic operations are processed using double-precision floating-point values. However, to keep the result bit-accurate, during each assignment, using functions `frexpl` and `ldexpl`. The double-precision value is decomposed into

mantissa and exponent, truncated according to the exponent and mantissa bit-width, and combined back into the double value.

The arithmetic operations of HWlns are implemented using HWfix numbers. Thus, we call the HWfix simulation code to perform the calculations of HWlns.

Number Representation Exploration

Based on all the above modules, we can now perform exploration of different number representations for the FPGA implementation of a specific piece of Fortran code.

The current tools support two different number representations: fixed-point and floating-point numbers (the value simulator for LNS is still in progress). For all the different number formats, the users can also specify arbitrary bit-widths for each different variable.

There are usually a large number of different variables involved in one circuit design. In our previous work, we usually apply heuristic algorithms, such as ASA (Ingber, 2004), to find out a close-to-optimal set of multiple values for the bit-widths of different variables. The heuristic algorithms may require millions of test runs to check whether a specific set of values meet the constraints or not. This is acceptable when the test run is only a simple error function and can be processed in nanoseconds. In our seismic processing application, depending on the problem size, it takes half an hour to several days to run one test set. Thus, heuristic algorithms become impractical.

A simple and straightforward method to solve the problem is to use uniform bit-width over all the different variables to either iterate over a set of possible values or use a binary search algorithm to jump to an appropriate bit-width value.

Based on the range information and the internal behavior of the program, we can also try to divide the variables in the target Fortran code into several different groups, and assign a different uniform bit-width for each different group. For instance, in the ‘complex exponential’ function, there is a clear boundary that the first half performs square, square root and division operations to calculate an integer result, and the second half uses the integer result as a table index, and performs sine, cosine and complex multiplications to get the final result. Thus, in the hardware circuit design, we divide the variables into two groups based on which half they belong to. Furthermore, in the second half of the function, some of the variables are trigonometric values in the range of $[-1, 1]$, while the other variables represent the seismic image data and scale up to 10^6 . Thus they can be further divided into two groups and assigned bit-widths separately.

```

! generation of table step%ctable

do i=1,size(step%ctable)
    k=ko*step%dstep*dsr%phase(i)
    step%ctable(i)=dsr%amp(i)*cmplx(cos(k),sin(k))
end do

! the core part of function wei_wem

```

```

do i4=1,size(wfld,4)
    do i3=1,size(wfld,3)
        do i2=1,size(wfld,2)
            do i1=1,size(wfld,1)

                k = sqrt(step%kx(i1,i3)**2 + step%ky(i2,i4)**2)
                itable =max(1, min(int(1 + k/ko / dsr%d) , dsr%n))
                wfld(i1,i2,i3,i4,i5)=wfld(i1,i2,i3,i4,i5)*step%ctable(itable)

            end do
        end do
    end do
end do

```

CASE STUDY I: COMPLEX EXPONENTIAL

Brief Introduction

The code above is the computationally intensive portion of the FK step in a downward continuation based migration. The governing equation for the FK step is the Double Square Root Equation (DSR) (?). The DSR equation describes how to downward continue a wavefield U one depth Δz step. The equation is valid for a constant velocity medium v and is based on the wave number of the source k_s and receiver k_g . The DSR equation can be written as,

$$U(\omega, k_s, k_g, z + \Delta z) = \exp \left[-i\omega v \left(\sqrt{1 - \frac{vk_g}{\omega}} + \sqrt{1 - \frac{k_s v}{\omega}} \right) \right] U(\omega, k_s, k_g, z), \quad (3)$$

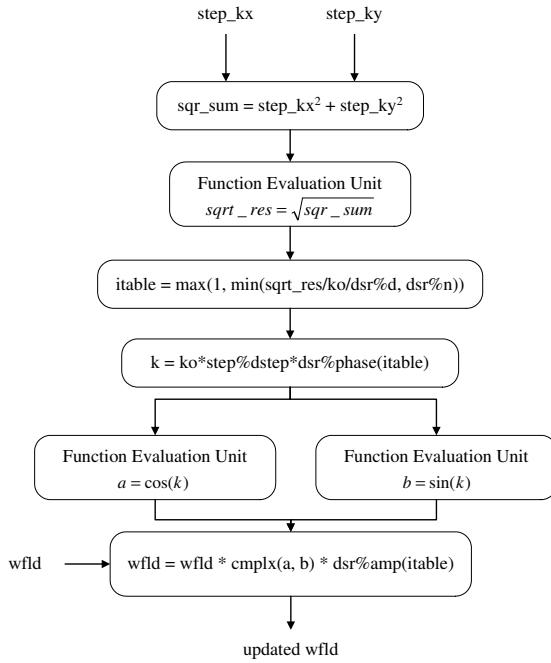
where ω is frequency. The code takes the approach of building *a priori* a relatively small table of the possible values of $\frac{vk}{\omega}$. The code then performs a table lookup that converts a given $\frac{vk}{\omega}$ value to an approximate value of the square root.

In practical applications `wfld` contains millions of elements. The computation pattern of this function makes it an ideal target to map to a streaming hardware circuit on an FPGA.

Circuit Design

The mapping from the software code to a hardware circuit design is straightforward for most parts. Fig. 4 shows the general structure of the circuit design. Compared with the software Fortran code shown above, one big difference is the handling of the sine and cosine functions. In the software code, the trigonometric functions are calculated outside of the five-level loop, and stored as a look-up table. In the hardware design, to take advantage of the parallel calculation capability provided by the numerous logic units on the FPGA, the calculation of the sine/cosine functions are merged into the processing core of the inner loop. Three function evaluation units are included in this design, to produce values for the square

Figure 4: General structure of the circuit design for the ‘wei_wem’ function. bob1/. wei-wem-circuit



Variable	step%x	ko	wfld_real	wfld_img
Max	0.377	0.147	3.918e6	3.752e6
Min	0	7.658e-3	4.168e-14	5.885e-14

Table 1: Profiling results for the ranges of typical variables in function ‘wei_wem’. ‘wfld_real’ and ‘wfld_img’ refer to the real and imaginary parts of the ‘wfld’ data. ‘Max’ and ‘Min’ refer to the maximum and minimum absolute values of variables.

root, cosine and sine functions separately. As mentioned in earlier, all three functions are evaluated using degree-one polynomial approximation with 386 to 512 uniform segments.

The other task in the hardware circuit design is to map the calculation into arithmetic operations of certain number representations. The previous table shows the value range of some typical variables in the ‘wei_wem’ function. Some of the variables (in the part of square root and sine/cosine function evaluations) have a small range within [0, 1], while other values (especially ‘wfld’ data) have a wider range from 10^{-14} to 10^6 . If we use floating-point or LNS number representations, their wide representation ranges are enough to handle these variables. However, if we use fixed-point number representations in the design, special handling is needed to achieve acceptable accuracy over wide ranges.

The first issue to consider in fixed-point designs is the division after the evaluation of the square root, $\frac{\sqrt{step\%x^2+step\%y^2}}{ko}$. Suppose the error in the square root result $sqrt_res$ is E_{sqrt} , and the error in variable ko is E_{ko} , assuming the division unit itself does not bring extra error, the error in the division result is given by $E_{sqrt} \cdot \frac{sqrt_res}{ko} + E_{ko} \cdot \frac{sqrt_res}{ko^2}$. As ko holds a dynamic range from 0.007658 to 0.147, and $sqrt_res$ has a maximum value of 0.533 (variables $step\%x$ and $step\%y$ have similar ranges), in the worst case, the error from $sqrt_res$ can be magnified by 70 times, and the error from ko magnified by approximately 9000 times. The values of $step\%x$, $step\%y$ and ko come from the software program as input

values to the hardware circuit.

To solve this problem, we perform shifts at the input side to keep the three values $step\%x$, $step\%y$ and ko in a similar range. For ko and the larger value between $step\%x$ and $step\%y$, we perform the shifts so that the leading one of them is just right to the fractional point (in the form of $0.1 \dots$); for the smaller value between $step\%x$ and $step\%y$, we assure it is shifted by the same distance as the larger value. The shifting distance difference between the ko and $step\%x$ is recorded, so that after the division, the result can be shifted back into the correct scale. In this way, the $sqrt_res$ has a range of [0.5, 1.414] and ko has a range of [0.5, 1]. Thus the division only magnifies the errors by an order of 3 to 6. Meanwhile, as the three variables $step\%x$, $step\%y$ and ko are originally in single precision floating-point representation in software, when we pass their values after shifts, a large part of the information stored in the mantissa part can be preserved. Thus, a better accuracy is achieved through the shifting mechanism for fixed-point designs.

Figure 5: Maximum and average errors for the calculation of the table index when using and not using the shifting mechanism in fixed-point designs, with different uniform bit-width values from 10 to 20.
bob1/.itable-error

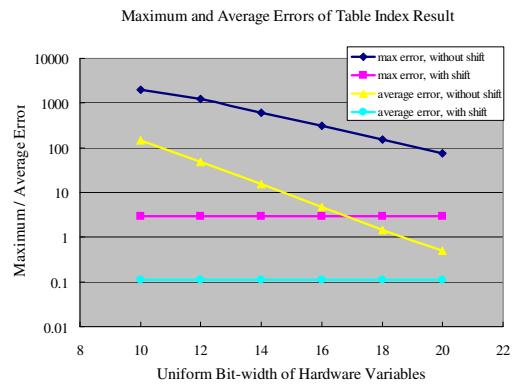


Fig. 5 shows experimental results about the accuracy of the table index calculation when using shifting or not using shifting, with different uniform bitwidths. The possible range of the table index result is from 1 to 2001. As it is the index for tables with smooth sequential values, an error within five indices is generally acceptable. We assume that the table index results calculated with double precision floating-point representation are accurate enough and use them as the true values for error processing. When the uniform bit-width of the design changes from 10 to 20, designs using the shifting mechanism show a stable maximum error of 3 and an average error around 0.11. On the other hand, the maximum error of designs without shifting vary from 2000 to 75, and the average errors vary from approximately 148 to 0.5. These results show that the shifting mechanism provides much better accuracy for the part of the table index calculation in fixed-point designs.

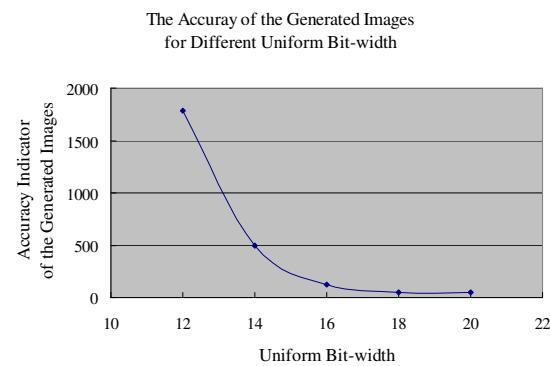
The other issue to consider is the representation of ‘wfld’ data variables. As shown in the table above, both the real and imaginary parts of ‘wfld’ data have a wide range from 10^{-14} to 10^6 . Generally, fixed-point numbers are not suitable for representing such wide ranges. However, in this seismic application, the ‘wfld’ data is used to store the processed image information. It is more important to preserve the pattern information shown in the data values rather than the data values themselves. Thus, by omitting the small values, and using the limited bit-width to store the information contained in large values, fixed-point representations still have a better chance to achieve accurate image in the final step. In our design, for convenience of bit-width exploration, we scale down all the ‘wfld’ data values by a ratio of 2^{-22} so that they fall into the range of [0, 1].

Bit-width Exploration Results

In the first step, we apply uniform bit-width over all the variables in the design. The approach for accuracy evaluation, introduced earlier, is used to provide a value that indicates the quality of the resulted seismic image.

The original software Fortran code of the `S_G_wem` application performs the whole computation using single-precision floating-point. We firstly replace the original Fortran code of function `wei_wem` with a piece of C++ code using double-precision floating-point to generate a full-precision result for comparsion. After that, to investigate the effect of the variables' bit-widths in function `wei_wem` on the accuracy of the whole application, we replace the code of function `wei_wem` with our simulation code that can be configured with different bit-widths, and generate results for different bit-width settings.

Figure 6: Variation of accuracy for a varying bit-width. `bob1/. unu-bw`

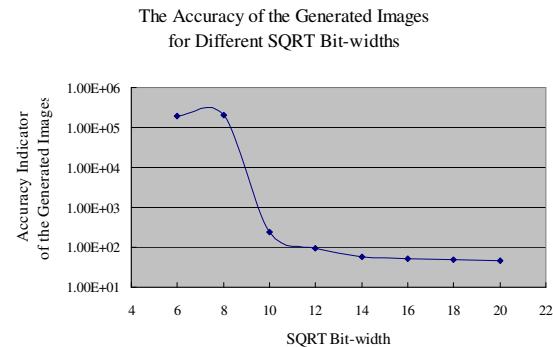


As mentioned earlier, according to their characteristics in range and operational behavior, we can also divide the variables in the design into different groups and apply a uniform bit-width in each group. In the design of function ‘`wei_wem`’, the variables are divided into three groups: SQRT, the part from the beginning to the table index calculation, which includes an evaluation of the square root; SINE, the part from the end of SQRT to the evaluation of the sine and cosine functions; and WFLD, the part that multiplies the complex values of ‘`wfld`’ data with a complex value consisting of the sine and cosine values (for phase modification), and a real value (for amplitude modification). To perform the accuracy investigation, we keep two of the bit-width values constant, and change the other one gradually to see its effect on the accuracy of the entire application.

Fig. 7 shows the accuracy of the generated images when we change the bit-width of the SQRT part from 6 to 20. The bit-widths of the SINE and WFLD parts are set to 20 and 30 respectively. Large bit-widths are used for these two parts so that they do not contribute much to the errors and the effect of variables' bit-width in SQRT can be extracted out. The case of SQRT bit-widths shows a clear cut at the bit-width value of 10. For bit-width values smaller than 10, the generated images show a large accuracy indicator value at the level of 10^5 , which means the pattern in the generated images are highly different from the correct ones. For bit-width values equal to or larger than 10, the accuracy indicator value drops to the level of 10^2 , which indicates a similar accuracy to single-precision floating-point results.

Similarly, Fig. 8 shows the case when we change the bit-width of the SINE part. The SINE bit-width changes from 6 to 20, while the bit-widths of the SQRT and WFLD parts are set to 20 and 30 respectively. There is also a fast decrease at the bit-width value of 8

Figure 7: Accuracy of the generated images for different SQRT bit-widths. The accuracy indicator value shows the difference between the pattern in the generated images and the pattern in the full-precision image. bob1/. sqrt-bw



(not quite evident in a logarithmic scale). The indicator value drops to the level of 10^2 when the bit-width increases to 12.

Figure 8: Accuracy of the generated images for different SINE bit-widths. The accuracy indicator value shows the difference between the pattern in the generated images and the pattern in the full-precision image. bob1/. sine-bw

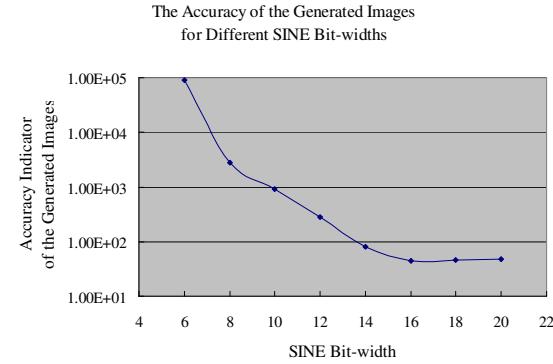
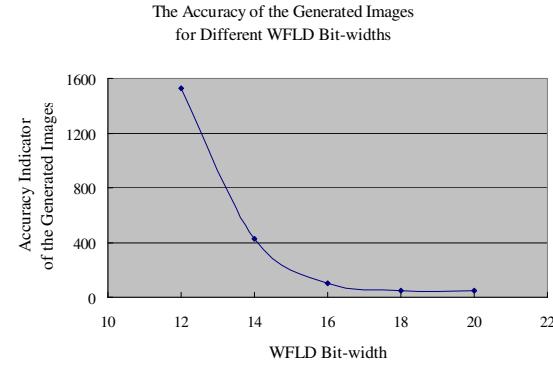


Figure 9: Accuracy of the generated images for different WFLD bit-widths (floating-point). The accuracy indicator value shows the difference between the pattern in the generated images and the pattern in the full-precision image. bob1/. wfld-bw



Hardware Acceleration Results

The hardware acceleration tool used in this project is the FPGA computing platform MAX-1, provided by Maxeler Technologies. It consists of a high performance Xilinx Virtex IV FX100 FPGA, and provides a high bandwidth interface of PCI Express X8 to the software side residing in CPUs. We found a speedup of 8x compared to the CPU implementation.

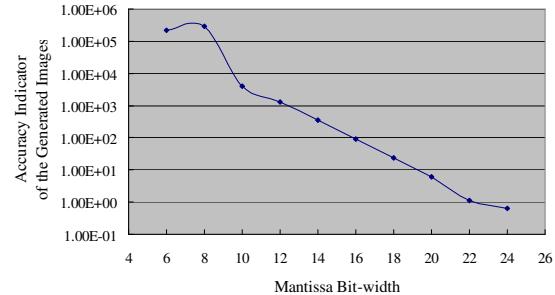
Figure 10: Accuracy of the generated images for different exponent bit-widths. The accuracy indicator value shows the difference between the pattern in the generated images and the pattern in the full-precision image. [bob1/. exp-bw]

The Accuracy of the Generated Images
for Floating-point Designs with Different Exponent Bit-width



Figure 11: Accuracy of the generated images for different mantissa bit-widths (floating-point). The accuracy indicator value shows the difference between the pattern in the generated images and the pattern in the full-precision image. [bob1/. man-bw]

The Accuracy of the Generated Images
for Floating-point Designs with Different Mantissa Bit-width



CASE STUDY II: CONVOLUTION

To test the speedup potential for reverse time migration we implemented a 6th order acoustic modeling kernel. The 3D convolution uses a kernel with 19 elements. Once each line of the kernel has been processed, it is scaled by a constant factor. We extend the approach to the 2D convolution used last year which works by indexing into the stream to obtain values already sent to the FPGA. These values are stored in BRAM FIFOs, automatically generated and assigned by ASC. The convolution was tested on a data size of $700 \times 700 \times 700$.

The main reason for a speedup is that the processor has limited computational resources. Furthermore, the processor uses floating-point units as opposed to fixed-point units. We exploit the parallelism of the FPGA to calculate one result per cycle. When ASC assigns the elements to BRAMs it does so in such a way as to maximize the number of elements that can be obtained from the BRAM every cycle. This means that consecutive elements of the kernel must not in general be placed in the same BRAM. Since we can use variable precision, we reduce the computation overhead, increasing the throughput. To compute the entire computation all at the same time (as is the case when a high-performance processor is used) requires a large local memory (in the case of the processor, a large cache). The FPGA has limited resources on-chip (376 BRAMs which can each hold 512 32-bit values). To solve this problem we break the large data-set into cubes. To utilize all of our input and output bandwidth, we assign 3 processing cores to the FPGA resulting in 3 inputs and 3 outputs per cycle at 125MHz (constrained by the throughput of the PCI-Express bus). This gives us a theoretical maximum throughput of 375M results a second.

The disadvantage to breaking the problem into smaller blocks is that the boundaries of each block are essentially wasted (although a minimal amount of reuse can occur) because they must be reused when the adjacent block is calculated. We do not consider this a problem since the blocks we use are at least $100 \times 100 \times 700$ which means only a small proportion of the data is resent. The amount of BRAM assigned to each block is calculated as follows:

$$\left\lceil \text{Total BRAM} \times \left\lfloor \frac{\text{Input bandwidth}}{\text{Input precision}} \right\rfloor \right\rceil \quad (4)$$

which assumes that the output precision is the same as the input precision. From this we can work out the size of the block. In our case we get $\lfloor 376 * \lfloor (64/21) \rfloor \rfloor = 125$. Due to the number of multipliers and adders required, we cannot fit 3 cores onto the FPGA directly because the number of slices used would be too high. If all of the operations are assigned to the DSP blocks we wouldn't have enough. We therefore choose a hybrid approach in which we break each multiply into 2 parts. We use one 18-bit hard multiplier (1 DSP block) and put the rest of the calculation (3 smaller multipliers) directly into logic.

In software, the convolution we try to accelerate executes in 11.2 seconds on average. The experiment was carried out using a dual-processor machine (each quad-core Intel Xeon 1.86GHz) with 8GB of memory.

In hardware, using the MAX-1 platform we obtain a 5 times speedup. The design uses 48 DSP blocks (30%), 369 (98%) RAMB16 blocks and 30,571 (72%) of the slices on the Virtex -4 chip. This means that there is room on the chip to substantially increase the kernel size. For a larger sized kernel (31 points) the speedup should be virtually linear resulting in a 8x speedup compared to the CPU implementation.

FURTHER POTENTIAL SPEEDUPS

All of the speedups in this paper include the transfer time to and from the processor. If multiple portions of the algorithm are performed on the FPGA without returning to the CPU the additional speedup can be considerable. In the cases shown in this paper the limiting factor is the transfer time. For example if the FFT and FK step can reside simultaneously on the FPGA the cost of the FK step disappears. In the case of acoustic modeling multiple time steps could be applied simultaneously.

CONCLUSIONS

We describe a software methodology for implementing and evaluating algorithmic performance on a FPGA. We found a 8x speedup in implementing (including transfer time) the FK step of downward continued migration on FPGA. In addition we found a 5-8x speedup in implementing a acoustic 3-D convolution kernel.

ACKNOWLEDGMENTS

This project is a joint research effort between the Center for Computational Earth and Environmental Science, Stanford Exploration Project, Computer Architecture Research Group at Imperial College London, and Maxeler Technologies.

REFERENCES

- Berenger, J., 1994, A perfectly matched layer for the absorption of electromagnetic waves: *Journal of Computational Physics*, **114**, 185–200.
- Biondi, B. and G. Palacharla, 1996, 3-d prestack migration of common-azimuth data: *Geophysics*, 1822–1832, Soc. of Expl. Geophys.
- Constantinides, G. A., P. Y. K. Cheung, and W. Luk, 2001, Heuristic Datapath Allocation for Multiple Wordlength Systems: Proc. Design Automation and Test in Europe Conference (DATE), 791–796.
- Fu, H., O. Mencer, and W. Luk, 2006, Comparing Floating-point and Logarithmic Number Representations for Reconfigurable Acceleration: Presented at the Proc. IEEE International Conference on Field-Programmable Technology (FPT).
- , 2007, Optimizing Logarithmic Arithmetic on FPGAs: Presented at the Proc. IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM).
- Gazdag, J. and P. Sguazzero, 1985, Migration of seismic data by phase shift plus interpolation, 323–330. Society Of Exploration Geophysicists.
- Ingber, L., 2004, Adaptive simulated annealing (asa) 25.15. <http://www.ingber.com>.
- Lee, D.-U., A. A. Gaffar, R. C. Cheung, O. Mencer, W. Luk, and G. A. Constantinides, 2006, Accuracy Guaranteed Bit-Width Optimisation: *IEEE Trans. Computer-Aided Design*, **25**, 1990–2000.
- Lee, D.-U., A. A. Gaffar, O. Mencer, and W. Luk, 2005, Optimizing Hardware Function Evaluation: *IEEE Trans. Comput.*, , no. 12.
- Muller, J., 1997, Elementary functions: algorithms and implementation: Birkhauser Boston, Inc.
- Pell, O. and R. G. Clapp, 2007, Accelerating subsurface offset gathers for: SEP-Report, **129**, 253–260.
- Shan, G. and B. Biondi, 2007, Angle domain common image gathers for steep reflectors: SEP-Report, **131**.

SEP PHONE DIRECTORY

Name	Phone	Login Name
Al-Theyab, Abdullah	723-3187	altheyab
Ayeni, Gboyega	723-0463	gayeni
Berryman, James	—	berryman
Biondi, Biondo	723-1319	biondo
Cardoso, Claudio	723-1250	claudio
Claerbout, Jon	723-3717	jon
Clapp, Bob	725-1334	bob
Curry, Bill	723-6006	bill
De Ridder, Sjoerd	723-3187	sjoerd
De Zan, Francesco	725-1331	francesco
Guitton, Antoine	—	antoine
Gunther, Roland	723-6006	rgunther
Halpert, Adam	723-3187	adam
Lau, Diane	723-1703	diane
Shan, Guojian	723-0463	shan
Shen, Xukai	723-3187	xukai
Shragge, Jeff	723-0463	jeff
Tang, Yaxun	723-1250	tang
Valenciano, Alejandro	723-1250	valencia

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, 2007-2008

Name	Company	Telephone	E-Mail
Raymond Abma	BP	(281) 366-4604	abmar1@bp.com
Dimitri Bevc	3DGeo	(408) 450-7840	dimitri@3dgeo.com
Biondo Biondi	SEP	(650) 723-1319	biondo@sep.stanford.edu
Robert Bloor (Co-chair, 1st year)	GX Technology	(713) 789-7250	rib@gxt.com
Luis L. Canales	WesternGeco	(713) 806-5271	lcanales@houston.westerngeco.slb.com
Jon Claerbout	SEP	(650) 723-3717	jon@sep.stanford.edu
Richard Cook	Shell	(713) 245-7195	richard.cook@shell.com
Stewart A. Levin (Co-chair, 2nd year)	Landmark Graphics	(303) 488-3062	salevin@lgc.com

Research Personnel

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.



Guojian Shan received his B.Sc. in Mathematics School of Peking University in July, 1998. From 1998 to 2001, he studied in Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences (CAS), and received his M.S. in Applied Mathematics in July, 2001. He joined SEP in 2001 and is currently working towards a Ph.D. in geophysics. He is a member of the SEG.



Alejandro A. Valenciano received a B.Sc. degree in Physics from Havana University (Cuba) in 1994, and a M.Sc. in Physics from Simon Bolivar University (Venezuela) in 1998. He worked in the Earth Science Department of PDVSA-INTEVEP from 1995 to 2001. He joined SEP to work towards a Ph.D in geophysics in the Fall of 2001.

