STANFORD EXPLORATION PROJECT

Gabriel Alvarez, Brad Artman, James Berryman, Biondo Biondi, Morgan Brown, Weitian Chen, Jon Claerbout, Robert Clapp, Marie Clapp, William Curry, Antoine Guitton, Seth Haines, Jesse Lomask, Paul Sava, Guojian Shan, Jeff Shragge, William Symes, Thomas Tisserant, Alejandro Valenciano, Ioan Vlad, and Guanquan Zhang

Report Number 113, May 2003
Preface

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

**ER** denotes Easily Reproducible and are the results of processing described in the paper. The author claims that you can reproduce such a figure from the programs, parameters, and makefiles included in the electronic document. The data must either be included in the electronic distribution, be easily available to all researchers (e.g. SEG-EAGE data sets), or be available in the SEP data library. 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 2.4 (using the Portland Group 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


---

1. http://sepwww.stanford.edu/private/docs/sep113
# Table of Contents

## Multiples

*Guiotton A.*, A comparison of three multiple-attenuation methods for a Gulf of Mexico dataset ................................................................. 1

*Brown M.*, Least-squares joint imaging of primaries and pegleg multiples: 2-D field data test .................................................................................. 17

*Sava P. and Guiotton A.*, Multiple attenuation in the image space ......................... 31

*Haines S., Guiotton A., and Sava P.*, Multiple suppression in the angle domain with non-stationary prediction-error filters ................................. 45

*Guiotton A.*, Multiple attenuation with multidimensional prediction-error filters ........ 57

*Shan G.*, Source-receiver migration of multiple reflections ................................. 75

*Brown M.*, Prestack time imaging operator for 2-D and 3-D pegleg multiples over nonflat geology ................................................................. 85

*Brown M.*, Amplitude modeling of pegleg multiple reflections .............................. 97

## Migration

*Biondi B.*, Narrow-azimuth migration of marine streamer data ............................ 107

*Shan G. and Zhang G.*, Equivalence between shot-profile and source-receiver migration ... 121

*Valenciano A. A. and Biondi B.*, Multichannel deconvolution imaging condition for shot-profile migration ......................................................... 127

*Artman B., Shragge J., and Biondi B.*, Operator aliasing in wavefield continuation migration 139

*Shragge J.*, Phase-shift migration of approximate zero-offset teleseismic data ............ 145

*Shragge J. and Artman B.*, Imaging with buried sources ................................. 157

*Vlad I., Tisserant T., and Biondi B.*, Improving the amplitude accuracy of downward continuation operators ......................................................... 163

## Angle gathers

*Biondi B. and Symes W.*, Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging .......................... 177
Tisserant T. and Biondi B., Wavefield-continuation angle-domain common-image gathers in 3-D ................................................................. 211
Sava P., Angle decompositions of images migrated by wavefield extrapolation ............ 221

Inversion and filtering

Clapp R. G., Multiple realizations and data variance: Successes and failures .......... 231
Lomask J., Flattening 3-D data cubes in complex geology ........................................ 247
Guittion A., Amplitude balanced PEF estimation .................................................. 261
Haines S. and Guittion A., Coherent noise suppression in electroseismic data with non-stationary prediction-error filters ............................. 277
Guittion A., Subtraction versus filtering for signal/noise separation ....................... 285
Vlad I., Enhanced random noise attenuation ...................................................... 291

Velocity

Sava P. and Biondi B., Wave-equation migration velocity analysis by inversion of differential image perturbations ................................................................. 299
Clapp R. G. and Clapp M. L., Steering filters in 3-D: Tubes rather than planes .......... 313
Clapp R. G., Semblance picking ............................................................................. 323
Chen W., Realization and analysis of an integration tomography scheme ............... 331

Migration and illumination

Clapp M. L., The oddities of regularized inversion: Regularization parameter and iterations 341
Guittion A., Amplitude and kinematic corrections of migrated images for non-unitary imaging operators .......................................................... 349
Clapp M. L., Directions in 3-D imaging - Strike, dip, both? ................................. 363
Clapp M. L., Illumination compensation: Model space weighting vs. regularized inversion 369

Interpolation

Curry W., Iteratively re-weighted least-squares and PEF-based interpolation .............. 379
Curry W., Parameter optimization for multiscale PEF estimation .......................... 391

Acquisition

Alvarez G., Flexible 3-D seismic survey design .................................................. 417
Vlad I., Ocean-bottom seismometers in Japan .................................................... 433

Rock Physics

Berryman J. G., Dynamic permeability in poroelasticity .................................... 443
Berryman J. G., Poroelastic shear modulus dependence on pore-fluid properties arising in a model of thin isotropic layers ................................................. 455

Computers

Clapp R. G., SEPlib programming and irregular data ....................................... 479
Clapp R. G., MPI in SEPlib .............................................................................. 491

SEP article published or in press, 2002-03 ...................................................... 501
SEP phone directory ....................................................................................... 505
Research personnel ....................................................................................... 507
SEP sponsors for 2002-03 ............................................................................ 515
A comparison of three multiple-attenuation methods for a Gulf of Mexico dataset

Antoine Guitton

ABSTRACT

Three multiple attenuation techniques are tested on a Gulf of Mexico dataset. These methods are (1) a hyperbolic Radon transform followed by a mute (2), the Delft approach and (3), a pattern-based technique. The Radon transform separates multiples and primaries according to their moveout. The Delft approach models the multiples and subtracts them by estimating adaptive filters. The pattern-based method uses the multiple model from the Delft approach to extract and separate multiples from the primaries according to their multivariate spectra. Because of the complex geology and the modeling uncertainties introduced by 3-D effects and the acquisition geometry, the Radon transform and the Delft approach do not perform as well as the pattern-based method. In addition, the pattern-based method works significantly better when higher dimension filters are utilized: diffracted multiples are well attenuated while preserving the primaries.

INTRODUCTION

Finding the best multiple attenuation method is an every day challenge in seismic processing. From a simple stack to the most advanced wavefield-based separation techniques, lie a wide range of multiple removal methods. The choice is often driven by the geology, the acquisition geometry and the processing cost. For complex geology, the wavefield undergoes severe distortions, modifications that make the multiple attenuation even more complicated to achieve. In those areas, the most advanced multiple attenuation techniques are usually needed. This paper aims to compare high-end attenuation techniques that can tackle the complexity of the multiple wavefield.

To achieve this goal, I use three multiple removal techniques. The first method consists of separating the multiples with a high-resolution hyperbolic Radon transform (Lumley et al., 1995; Kostov and Nichols, 1995). This method is particularly effective at separating multiples and primaries when both have different moveouts. The high-resolution provides high separability in the Radon domain thanks to a Cauchy regularization of the model space (Sacchi and Ulrych, 1995). This method is very robust and might be the only choice for separating primaries and multiples with, for instance, land data.

The second method is very similar to the Delft approach (Verschuur et al., 1992) where

1email: antoine@sep.stanford.edu
the multiples are first predicted and then adaptively subtracted. This method can predict any
type of multiple as long as enough data are recorded. In addition, no model of the subsurface
is needed. The multiples are removed by estimating non-stationary matching filters in the time
domain (Rickett et al., 2001). The Delft approach works particularly well for marine data.
Some recent examples with land data show also promises (Verschuur and Kelamis, 1997).

The last method separates the primaries and the multiples based and their pattern (Manin
and Spitz, 1995; Brown and Clapp, 2000). The patterns are approximated with non-stationary
multidimensional prediction-error filters (PEFs). These filters are estimated from a noise and
a signal model. The noise model is provided by the Delft approach and the signal model is
estimated by convolving the data and the noise PEFs. This method has the ability to work in
any dimension and is robust to modeling uncertainties (Guitton, 2003).

These three methods are tested on a deep water, 2-D Gulf of Mexico dataset provided
by WesternGeco. One interesting geological feature of this dataset is a shallow salt body
that creates shadow zones and strong multiples. There are also off-plane/3-D multiples that
make the noise attenuation quite challenging. With this dataset, I show that the best multiple
attenuation result is obtained with the pattern-based approach with 3-D filters.

In the next section, I describe each multiple attenuation technique emphasizing their strengths
and weaknesses. Then, I apply each of these methods on the 2-D Gulf of Mexico dataset.

THEORY OF MULTIPLE ATTENUATION

Here I present three multiple attenuation techniques. One method is based on the moveout
discrepancies that exist between primaries and multiples. The second method aims to predict
the multiples and to adaptively subtract them. The third method utilizes the multiple model
to estimate the multivariate spectrum of the primaries and the multiples for a pattern-based
attenuation technique.

Multiple attenuation in the Radon domain

Multiple attenuation with Radon transforms (RT) are popular and robust methods (Foster and
Mosher, 1992). These techniques use the moveout discrepancy between primaries and multi-
ples in order to separate them. In the method used here, I sort the data into Common Mid Point
(CMP) gathers and remap them with a hyperbolic Radon transform (HRT). The estimation of
the Radon domain can be cast as a linear transformation (Thorson and Claerbout, 1985) where,
given a model \( m \), we estimate the data \( d \) via the operator \( L \) (the HRT) as follows:

\[
d = Lm. \tag{1}
\]

The unknown model \( m \) can be estimated in a least-squares sense. We then minimize the
objective function

\[
g(m) = ||Lm - d||^2. \tag{2}
\]
To increase the separability of the multiples and the primaries in the Radon domain $\mathbf{m}$, I add a regularization term in equation (2) that will enforce sparseness in the model space. The regularization term is a Cauchy function (Sacchi and Ulrych, 1995). Thus, we have to minimize the new objective function

$$g(\mathbf{m}) = \| \mathbf{Lm} - \mathbf{d} \|^2 + \epsilon^2 \sum_{i=1}^{n} \ln(b + m_i^2),$$  \hspace{1cm} (3)

where $n$ is the size of the model space and $\epsilon$ and $b$ two constants to be chosen a-priori: $\epsilon$ controls the amount of sparseness in the model space and $b$ relates to the minimum value below which everything in the Radon domain should be zeroed. The least-squares inverse of $\mathbf{m}$ is

$$\hat{\mathbf{m}} = \left( \mathbf{L}'\mathbf{L} + \epsilon^2 \text{diag}(1/(b + m_i^2)) \right)^{-1} \mathbf{L}'\mathbf{d},$$ \hspace{1cm} (4)

where $\text{diag}$ defines a diagonal operator. Because the model or the data space can be large, I estimate $\mathbf{m}$ iteratively. Note that the objective function is non-linear because of the logarithm in the regularization term. Therefore, I use a quasi-Newton method called limited-memory BFGS (Broyden, 1969; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970; Nocedal, 1980) to find the minimum of $g(\mathbf{m})$. This method has proven efficient for the minimization of the Huber function (Guitton and Symes, 1999).

From the estimated model $\mathbf{m}$, I separate the multiples from the primaries in the Radon domain. In the following examples, the muting function is identical for each CMP gather. We transform back the multiples in the image space by applying $\mathbf{L}$, and subtract them from the input data to obtain multiple-free gathers.

Multiple attenuation with Radon transforms is widely used in the industry. For complex geology, however, Radon transforms are not optimal because the moveout discrepancies can be too small or the moveout of both multiples and primaries can be quite distorted (Matson et al., 1999). Therefore, more sophisticated multiple attenuation techniques are needed. Two of these techniques are the Delft approach (Verschuur et al., 1992) and the inverse-scattering method (Weglein et al., 1997). In the next section, I describe my implementation of the Delft approach for multiple attenuation.

**Multiple attenuation with the Delft approach**

The Delft approach (Verschuur et al., 1992) is able to remove surface-related multiples for any type of geology as long as the receiver and the source coverage at the surface is dense enough. One of the main advantage of the Delft method is that no subsurface information is required.

In my implementation of the Delft approach, I first create a model of the multiples by autoconvolving in time and space the shot gathers. I do this convolution once such that the kinematics of all surface-related multiples are accurate. Doing this, the relative amplitudes of the first-order multiples are correct, but higher-order multiples are over-predicted, amplitude-wise (Wang and Levin, 1994; Guitton et al., 2001b).
Once a multiple model has been estimated, it is adaptively subtracted from the data. Note that as pointed out by Berkhout and Verschuur (1997), this first subtraction step should be followed by more iterations. The goal of the iterative procedure is to better estimate and eliminate higher-order multiples (Verschuur and Berkhout, 1997). In this paper, I iterate only once and hope that the adaptive subtraction step is permissive enough to handle all the multiples at once.

Keeping the adaptive-subtraction procedure, I use the non-stationary filtering technology developed at the Stanford Exploration Project to perform the multiple attenuation step (Rickett et al., 2001). The main advantage of these filters is that they are computed in the time domain and thus take the inherent non-stationarity of the multiples and the data into account very efficiently. Therefore, I can locally estimate adaptive filters that will give the best multiple attenuation result. Thereby only one iteration of the Delft approach should be needed. Note that I am estimating two-sided 2-D filters, which gives a lot of degrees of freedom for the matching of the multiple model to the real multiples in the data.

Thus, given a model of the multiple $M$ and the data $d$, I estimate a bank of non-stationary filters $f$ such that

$$g(f) = \|MF - d\|^2 + \epsilon^2 \|Rf\|^2$$

is minimum. In equation (5), $R$ is the Helix derivative (Claerbout, 1998) that smooths the filter coefficients across micro-patches (Crawley, 2000) and $\epsilon$ a constant to be chosen a-priori. Note that $M$ corresponds to the convolution with the model of the multiples $m$ (Robinson and Treitel, 1980). Remember that this model of the multiples is obtained by convolving in space and time the input data:

$$m(\omega) = d(\omega) * d(\omega)$$

where $*$ defines the convolution process detailed in Verschuur et al. (1992) and $m(\omega)$ and $d(\omega)$ are the multiple model and the data for one frequency, respectively. In equation (5), the filters are estimated iteratively with a conjugate-gradient method.

The Delft approach is widely used in the industry and is known to give the best multiple attenuation results for complex geology (Dragoset and Jericevic, 1998). However, it has been shown that this method suffers from the approximation made during the adaptive filtering step. For instance, when “significant” amplitude differences exist between the primaries and the multiples, the multiple model might be matched to the primaries and not to the multiples. A solution to this problem is using the $\ell^1$ norm in equation (5) (Guitton and Verschuur, 2002). Another assumptions made in equation (5) is that the signal has minimum energy. Spitz (1999) illustrates the shortcomings of this assumption and advocates that a pattern-based method is a better way of subtracting multiples to the data. In the next section, I present such a methodology. This technique uses non-stationary multidimensional prediction-error filters to estimate the pattern of both the primaries and the multiples. Then, multiples and primaries are separated according to their multivariate spectra (Claerbout and Fomel, 2001).
Multiple attenuation with a pattern-based approach

Pattern-based techniques have been recently proposed as a way of removing coherent noise (Manin and Spitz, 1995; Guitton and Cambois, 1998; Brown and Clapp, 2000; Guitton et al., 2001a; Soubaras, 2001). The philosophy behind these methods is that the noise and the signal have different moveouts and/or amplitude behaviors that can be utilized to discriminate and separate them. The patterns can be reliably estimated with prediction-error filters (Claerbout and Fomel, 2001). In this section, I describe my implementation of a pattern-based technique.

Guitton (2003) describes in detail the pattern-based method used in this paper. The main idea is that the patterns of both the noise $n$ and the signal $s$ can be estimated with time domain non-stationary prediction-error filters. Thanks to the helical boundary conditions (Mersereau and Dudgeon, 1974; Claerbout, 1998), these filters can have any dimension.

The two important components of the proposed method are the noise PEFs $N$ and the signal PEFs $S$. Then, in Guitton (2003) I show that the signal can be estimated by minimizing the objective function

$$g(s) = \|M(Ns - Nd)\|^2 + \epsilon^2 \|MSs\|^2$$

(7)

where $M$ is a masking operator that preserves the signal where no multiples are present and $d$ is the data vector (primaries plus multiples). As demonstrated by Abma (1995), this approach is similar to Wiener filtering. Therefore, the noise and signal are separated according to their multivariate spectra that we approximate with non-stationary PEFs. One important assumption made is that the noise and signal are uncorrelated. If they are correlated, then complications arise like estimating $\epsilon$ accurately or computing the cross-spectrum between the noise and the signal.

To estimate the noise and signal PEFs, we need to derive a model of the noise and a model of the signal. For the multiple attenuation problem, the model given by the Delft approach is the best choice. This choice is not restrictive because we could use other techniques to estimate a model of the multiples. For instance, Haines et al. (2003) show that a Radon transform can give a satisfying model of the multiples while improving on the Radon demultiple result. Another important assumption made with the pattern-based approach is that the model has accurate kinematics and amplitudes. It is well established that with one iteration only, the Delft approach does not properly model amplitudes for high-order multiples (Wang and Levin, 1994; Guitton et al., 2001b). Guitton (2003) shows that 3-D filters cope better with modeling inaccuracies than 2-D filters. Once the multiple model is computed and the noise PEFs $N$ are estimated, I convolve $N$ with the data $d$ to obtain a signal model from which I estimate the signal PEFs $S$ (Spitz, 2001, personal communication).

The proposed pattern-based approach works very well. I show in Guitton (2003) a very detailed comparison of this technique for 2-D and 3-D filters. In the next section, I compare the three multiple attenuation techniques, i.e., Radon, Delft and the pattern-based approach on a Gulf of Mexico dataset provided by WesternGeco. This example proves that the best multiple attenuation result is obtained with the pattern-based approach and 3-D filters.
MULTIPLE ATTENUATION RESULTS

Now, I show the multiple attenuation results obtained with field data from the Gulf of Mexico (Figure 1). This dataset corresponds to a deep water survey (water bottom below one second) with a shallow salt body, thus creating strong multiples. We have only one streamer for this dataset making the modeling of 3-D multiples impossible. The three methods detailed in the preceding sections are used to attenuate the multiples.

Note that the Delft and pattern-based approaches are carried out in the shot domain, as opposed to the CMP domain for the HRT. Thus I decided to make the comparison of the three techniques in the CMP domain. Figures 2 and 3 show two CMP gathers along with the multiple model computed with the autoconvolution of the shot gathers. One CMP gather is located outside the salt boundaries, i.e., in the sediments (Figure 2). The other gather is located inside the salt boundaries (Figure 3). These two areas allow us to see the strengths and weaknesses of each method for different environment.

Outside the salt boundaries

Figure 4 displays the multiple attenuation results for the first CMP gather in Figure 2a. The HRT (Figure 4b) does a very decent job at short and long offsets. It especially does better than
Figure 2: CMP gathers extracted at midpoint location 3200 m. (a) CMP gather of the data with multiples. (b) Multiple model. Note that the near offset traces have the wrong amplitudes.

Figure 3: CMP gathers extracted at midpoint location 15500 m. (a) CMP gather of the data with multiples. (b) Multiple model. Note that the near offset traces have the wrong amplitudes.
the adaptive subtraction at short offsets (Figure 4a). The pattern-based methods (Figures 4c and 4d) lead to the best multiple attenuation result at short offsets. The missing information after 3.5 s. in Figures 4c and 4d comes from the convolution with the PEFs (Claerbout and Fomel, 2001). Overall, the pattern-based method with 3-D filters gives the best result while preserving the primaries.

**Inside the salt boundaries**

Figure 5 displays the multiple attenuation results for the second CMP gather in Figure 3a. The multiple attenuation becomes much more challenging because of various 3-D effects not taken into account in the multiple modeling step. The HRT result in Figure 5b shows a lot of multiples for which the moveouts are not hyperbolic, thus not well attenuated in the Radon domain. The adaptive subtraction result in Figure 5a indicates that the modeling inaccuracies at short offsets tend to degrade the estimated signal. The pattern-based method with 2-D PEFs in Figure 5c does better than the adaptive subtraction technique at both short offsets and late times. The adaptive subtraction technique does better than the 2-D PEFs in the middle of the CMP gather. Again, the best multiple attenuation results are obtained with the pattern-based method with 3-D PEFs (Figure 5d). Note that a strong primary at 3.7 s. is slightly attenuated with the pattern-based approach. For this event, primaries and multiples might be correlated across offset and shot position. A bigger \( \epsilon \) in equation (7) would solve the problem by allowing more energy from the signal to be preserved.

**Stacked sections**

I now compare the stacked sections for the three techniques in Figure 6. These figures are clipped to the same value for direct comparison. It is clear from Figure 6 that the best multiple attenuation result is obtained with the pattern-based technique with 3-D PEFs. Then comes the pattern-based method with 2-D PEFs, then the adaptive subtraction and the HRT. The HRT is simply unable to correctly attenuate multiples in the complex area defined by the salt boundaries. However, it does a fairly good job outside the salt boundaries. The quality of the adaptive subtraction method is hampered by modeling uncertainties, especially where the diffracted multiples are the strongest, i.e., between four and five seconds.

It is pleasing to see that the pattern-based method, while preserving the primaries, removes the diffracted multiples very well. The multiple attenuation result with 3-D filters is even more striking: the diffracted multiples are almost completely gone (see at four seconds). Although the model of the multiples is not perfect, there is enough resemblance between the “real” diffracted multiples and the modeled ones across dimensions to obtain a satisfying attenuation.

As a final comparison, I show in Figure 7 the stacked sections of the removed multiples for the three methods. It shows that the pattern-based approach with 3-D filters removed more multiples than any other method.
Figure 4: Multiple attenuation results for the CMP gather in Figure 2. (a) Signal estimated after adaptive subtraction. (b) Signal estimated after HRT and mute. (c) Signal estimated with a pattern-based method and 2-D PEFs. (d) Signal estimated with a pattern-based method and 3-D PEFs. [antoine6-comp-multatt-cmp.3200] [CR,M]
Figure 5: Multiple attenuation results for the CMP gather in Figure 3. (a) Signal estimated after adaptive subtraction. (b) Signal estimated after HRT and mute. (c) Signal estimated with a pattern-based method and 2-D PEFs. (d) Signal estimated with a pattern-based method and 3-D PEFs.
Figure 6: Stacked sections of the estimated signal. (a) Adaptive subtraction result. (b) HRT result. (c) Pattern-based method result with 2-D PEFs. (d) Pattern-based method result with 3-D PEFs. [antoine6-compstack-multatt] [CR,M]
Figure 7: Stacked sections of the removed multiples. (a) Adaptive subtraction result. (b) HRT result. (c) Pattern-based method result with 2-D PEFs. (d) Pattern-based method result with 3-D PEFs. [antoine6-compstack-diff] [CR,M]
DISCUSSION-CONCLUSION

In this paper, I compared multiple attenuation results obtained with three different techniques. These methods are (1), the hyperbolic Radon transform (HRT), for which multiples are attenuated according to their moveout (2), the Delft approach, for which the multiples are first predicted and then adaptively subtracted from the data and (3), a pattern-based method, for which the multiples and primaries are separated according to their multivariate spectra. These three techniques were tested on a deep water, Gulf of Mexico dataset. This dataset features a shallow salt body that creates strong complex multiples in the subsurface. The main goal of this paper was to illustrate the strengths and weaknesses of each method. It also intended to show that our most recent developments in non-stationary prediction-error filters estimation could be effectively used for multiple attenuation.

From my results, it appears that the HRT is unable to properly attenuate multiples in complex geology. This problem comes from the moveouts of the multiples that cannot be described by simple functions. This is particularly true when the multiples associated with the rugose salt are present in the gather. With non-hyperbolic events, the energy of the multiples tends to spread out in the Radon domain. Therefore, the primaries and the multiples do not separate very well. However, the HRT does a good job at removing multiples outside the salt boundaries. There, the HRT compares favorably with the Delft approach. For complex geology, applying the HRT in the image space (Sava and Guitton, 2003) is a better alternative.

The Delft approach with non-stationary adaptive filtering removes more multiples inside the salt boundaries than the HRT. In particular, it attenuates complex multiples much better. One drawback of the adaptive filtering approach, however, is that it tends to be very sensitive to inaccuracies in the multiple model. Thus, the quality of the subtraction decreases at short offset and where unmodeled multiples are present like diffracted multiples or other 3-D effects.

It is very pleasing to see that the the pattern-based method leads to the best multiple attenuation results both inside and outside the salt boundaries. One main advantage of these techniques is that they tend to cope slightly better with inaccuracies in the multiple model than the adaptive subtraction. This comes from the fact that the prediction-error filters approximate multidimensional spectra. Therefore, as long as the multiple model has the same “character” than the real multiples, the attenuation works. In addition, more dimensionality in the filters and in the data noticeably increases the quality of the multiple attenuation results. Thus, adding more dimensions to our data and filter, we can better discriminate between the primaries and the multiples because they look more different.

From this analysis it appears that the pattern-based approach with 3-D non-stationary filters yields the best multiple attenuation result. At this stage, it is important to keep in mind that the pattern-based approach works as long as the primaries and multiples do not have the same multivariate spectrum. If they do, we can either try to estimate the cross-spectrum, estimate the regularization parameter better or choose another method. The Gulf of Mexico dataset does not seem to cause this kind of problem, however. The next “natural” step consists in testing this technique with 3-D data. Of course, I need to devise a strategy for modeling the multiples in 3-D. The convolutional model of the Delft approach might not be feasible in practice and
more practical approaches, like the one currently developed by Brown (2003), will be needed.

ACKNOWLEDGMENTS

I thank WesternGeco for providing the data.

REFERENCES


Crawley, S., 2000, Seismic trace interpolation with nonstationary prediction-error filters: SEP–104.


Least-squares joint imaging of primaries and pegleg multiples:  
2-D field data test

Morgan Brown¹

ABSTRACT

In this paper I present an improved least-squares joint imaging algorithm for primary reflections and pegleg multiples (LSJIMP) which utilizes improved amplitude modeling and imaging operators presented in two companion papers in this report (Brown, 2003b,a). This algorithm is applied to the entire Mississippi Canyon 2-D multiples test dataset and demonstrates a good capability to separate pegleg multiples from the data.

INTRODUCTION

Multiple reflections are generally treated as noise in the reflection seismic experiment, and strategies for their removal occupy a prominent place in the exploration seismic literature. Tantalizingly, though, multiples often penetrate deeply into the earth, and thus have the potential to aid the imaging of the prospect zone. Algorithms to structurally image multiples have been in the exploration geophysics literature for over a decade (Reiter et al., 1991; Berkhout and Verschuur, 1994; Yu and Schuster, 2001; Guitton, 2002; Shan, 2003), though none successfully addresses perhaps the most important question: How can any usable information from the multiples be rigorously extracted and exploited? In a previous work (Brown, 2002), I presented a least-squares joint imaging technique which tentatively answers this question.

In two companion papers in this report (Brown, 2003b,a), I derived operators which, to reasonable accuracy, model the physical (kinematic and amplitude) connection between pegleg multiples and their corresponding primary events. These operators transform a pegleg multiple into a “copy” of its primary. My regularized least-squares joint imaging scheme, “LSJIMP”, (Least-squares Joint Imaging of Multiples and Primaries) then exploits this redundancy to separate the pegleg multiples from the data, and at the same time, “spreads” additional information provided by the multiples into the primary image.

I apply LSJIMP to the Mississippi Canyon 2-D multiples test data, and it demonstrates good noise removal and signal preservation characteristics. LSJIMP’s inherent efficiency, combined with the use of Message Passing Interface (MPI) parallelization ensure that its computational performance compares favorably with other advanced multiple suppression techniques.

¹email: morgan@sep.stanford.edu
LEAST-SQUARES JOINT IMAGING OF MULTIPLES AND PRIMARIES

To the uninitiated, LSJIMP may seem at best a perplexing idea; at worst, an egregious display of circular reasoning. Signal/noise separation and imaging have traditionally occupied mutually exclusive domains. Usually multiple suppression is performed as a prerequisite to imaging. While some recent approaches do the multiple suppression after imaging (Sava and Guitton, 2003), the two steps are still largely independent.

With LSJIMP, the multiple separation and imaging steps are innately intertwined. Figure 1 motivates the fundamental difference between LSJIMP and conventional multiple suppression and imaging. The “NMO for primaries” panel is the usual domain of multiple suppression/imaging algorithms. There, the signal is the primary reflections and the noise is the multiples. LSJIMP expands the dimensionality of the problem. In the other two panels, the signal is not the primaries, but instead the flattened multiples. The key observation is that the signal in each panel is consistent with the signal in the other panels.

Nearly every existing multiple suppression technique exploits differences between multiples and primaries in the left panel only. LSJIMP does that, but the novelty of the method is that it also exploits similarities between signal across panels to improve the separation. “Noise” is sometimes “signal”, and vice versa, depending on which image one views. Why not use all the information at our disposal?

Figure 1: Solid lines in each panel are “signal”; dashed lines are “noise”. Left: A CMP gather after NMO. The seabed (WB) produces pegleg multiples WBM and RPL. Reflector R also produces pegleg RM. Center: After NMO for seabed peglegs, WBM and RPL are aligned with WB and R. Right: After NMO for R’s peglegs, RM is aligned with R and RPL.

Nemeth Forward Model

Nemeth et al. (1999) introduced an under-determined least-squares formulation to jointly image compressional waves and various (non-multiple) coherent noise modes. Guitton et al. (2001) extended this technique to multiple suppression, using a prior noise model and prediction-error filters to model the noise and signal.

We can model the recorded data, here a CMP gather, as the superposition of the primary reflections and \( p \) orders of pegleg multiples from a single interface (e.g., the seabed). The
$i^{th}$ order pegleg has $i + 1$ “legs”, or independent arrivals (Brown, 2003a). If we denote the recorded data as $d$, the primaries as $d_0$, and the $k^{th}$ leg of the $i^{th}$ order pegleg as $d_{i,k}$, the model of the data takes the following form.

$$d = d_0 + \sum_{i=1}^{p} \sum_{k=0}^{i} d_{i,k}, \quad (1)$$

The main goal of LSJIMP is to use the multiples as a constraint on the primaries. Thus the model space of the LSJIMP process will be a collection of images, one corresponding to the primaries, and one for each leg of each order of pegleg. When the process is finished, the events in each image should have the same timing and AVO signature as the primaries; they should be “copies of the primaries”. The forward modeling equation is the physical mapping which transforms these copies of the primaries into the multiples recorded in the data.

Brown (2003b) derived “HEMNO” (Heterogeneous Earth Multiple NMO Operator), an operator which independently images each leg of a pegleg (flattens and shifts to the zero-offset traveltime of the primary). Brown (2003a) derived a series of linear operators which modify the amplitude of peglegs to account the effects of the multiple leg of the raypath (reflection and transmission). Taken together, these operators transform a single-CMP image that resembles the primaries, into “data” that resembles a pegleg multiple. Let us rewrite equation (1):

$$d = N_0 m_0 + \sum_{i=1}^{p} \sum_{k=0}^{i} R_i N_{i,k} S_i G_i m_{i,k} \quad (2)$$

$m_0$ is the primary image, flattened by the adjoint of the NMO operator $N_0$. $m_{i,k}$ is image of the $k^{th}$ leg of the $i^{th}$-order pegleg, flattened by $N_{i,k}$, the adjoint of the HEMNO equation. To all $i^{th}$-order peglegs, $G_i$ applies a differential geometric spreading correction, $S_i$ applies Snell resampling, and $R_i$ applies the appropriate reflection coefficient. The motivation and implementation of $G_i$, $S_i$, and $R_i$ are discussed in detail by Brown (2003a).

For clarity, we can rewrite equation (2) in matrix notation:

$$[N_0 \ (R_1 N_{1,0} S_1 G_1) \ (R_1 N_{1,1} S_1 G_1) \ \cdots \ (R_p N_{p,0} S_p G_p) \ \cdots \ (R_p N_{p,p} S_p G_p)] \ [m_0 \ \ m_{1,0} \ \ m_{1,1} \ \ \cdots \ \ m_{p,0} \ \ \cdots \ \ m_{p,p}] = L_n m, \quad (3)$$

and define the data residual as the difference between modeled and recorded data:

$$r_d = d - L_n m, \quad (4)$$

Viewed as a standard least-squares inversion problem, where the model is adjusted to minimize the $L_2$ norm of the data residual, equation (4) is under-determined. The addition of model regularization operators, defined in later sections, forces the problem to be over-determined.
Consistency of the Data and the Crosstalk Problem

Our hope is that, after solving equation (3) for the unknown model, \( m_0 \) contains only primaries (and other non-pegleg events) and the \( m_{1,k} \) contain only peglegs. Unfortunately, simple least-squares minimization of the data residual (4) proves insufficient to properly segregate the various modes. Nemeth et al. (1999) pins the problem on operator “overlap”, or coincident operator range. In this application, operator overlap is most troublesome at near offsets, where primaries and multiples are both flat. If (for instance) \( m_0 \) contains residual first-order pegleg multiple energy, equation (1) will map this energy back into data space, at the position of a first-order multiple at near offsets. The residual multiple energy is called “crosstalk” (Claerbout, 1992). Luckily, Nemeth et al. show that that properly designed model regularization operators can at least partially mitigate the crosstalk problem.

REGULARIZATION OF THE LEAST-SQUARES PROBLEM

In this section, three discriminants between crosstalk and signal are exploited to devise model regularization operators which “guide” the final model toward the one with the peglegs best separated from the data.

1. **Inconsistency with multiple order** - After imaging, corresponding crosstalk events on two model panels have different residual moveout, e.g. residual first-order multiples on \( m_0 \) and residual second-order multiples on \( m_{1,k} \). The moveout difference is negligible at near offsets, but larger than the Fresnel Zone (half a wavelength) at long offsets. Conversely, actual signal events are flat on all the \( m_{i,k} \). **Conclusion**: For fixed \((t,x)\), the difference between one model panel and another will be relatively small where there is signal, but large where there is crosstalk noise, especially at far offsets.

2. **Curvature with offset** - After imaging, signal events are flat, while crosstalk events have at least some residual curvature, especially at far offsets and in regions with a strong velocity gradient. **Conclusion**: Provided that the AVO response of the signal changes slowly with offset, the difference (in \(x\)) between adjacent samples of any \( m_{i,k} \) will be relatively small where there is signal, but large where there is crosstalk noise.

3. **Predictability of “pre-seabed multiple” events** - The third discriminant exploits the inherent predictability of crosstalk to suppress it. Between the seabed reflection and the onset of its first multiple, the recorded data contains only primaries (inter-bed multiples and locally-converted shear waves are generally weak); these strong events spawn the (usually) most troublesome crosstalk events. Fortunately, the pre-seabed-multiple primaries can be used to directly construct a prior model of the crosstalk noise, valid even at near offsets. **Conclusion**: Given an accurate kinematic model of crosstalk noise, a corresponding set of model-space weights used in a model regularization term penalizes crosstalk events but not signal.
**Model Regularization 1: Differencing across multiple order**

We can write the model residual corresponding to the model regularization operator which differences between adjacent \( m_i \) at a fixed \((\tau, x)\).

\[
    r_m^{[1]}(\tau, x, j) = m_j(\tau, x) - m_{j+1}(\tau, x) \quad \text{where} \quad j = [0, p(p + 3)/2]. \tag{5}
\]

\( p \) is the maximum order of multiple included in equation (2). Here we have modified the notation a bit and written \( m_j \) rather than \( m_{i,k} \). This is because the difference (5) is blind to the order or leg of the pegleg corresponding to \( m_j \); it is simply a straight difference across all the model panels.

By design, signal (non-crosstalk) events on the \( m_{i,k} \) in equation (3) are assumed physically invariant for all \( i \) and \( k \) – everything is a “copy of the primary”. Again, this is the crucial fact underlying LSJIMP. Minimally, multiples provide a redundant constraint on the amplitude of the primaries; where no data is recorded (missing traces, near offsets), they provide additional information about the primaries. Equation (5) is a systematic way in which to exploit the multiples’ redundancy, and to integrate any additional information that they might provide.

**Model Regularization 2: Differencing across offset**

We can similarly write the model residual corresponding to the model regularization operator which applies a difference with offset to each \( m_{i,k} \) at fixed \( t \).

\[
    r_m^{[2]}(\tau, x, i) = m_{i,k}(\tau, x) - m_{i,k}(\tau, x + \Delta x). \tag{6}
\]

A similar approach is used by Prucha and Biondi (2002) to regularize prestack depth migration in the angle domain.

**Model Regularization 3: Crosstalk-boosting weighting**

NMO (for primaries or multiples) flattens events of a single order, but leaves events of other orders (crosstalk) remaining with residual curvature. To compute (for instance) the crosstalk model for first-order pegleg multiples on the primary model panel, \( m_0 \), the following steps are followed.

- Apply NMO for primaries to data: \( c_0 = N_0^T d \).
- Zero \( c_0 \) in the range \( \tau^* \geq \tau \geq 2\tau^* \), where \( \tau^*(y) \) is the zero-offset traveltime to the multiple-generating layer.
- Simulate the kinematics of the data’s total first-order pegleg by applying inverse HEMNO: \( c_1 = (N_{0,1} + N_{1,1})c_0 \).
- Apply NMO for primaries to simulate the kinematics of the first-order pegleg crosstalk event in the primary model panel, \( m_0 \): \( c = N_0^T c_1 \).
Normalize \( c \) to \([0,1]\) and clip if desired.

Figure 2 illustrates the application of the crosstalk weight for first and second-order peglegs on \( m_0 \) applied to a NMOed synthetic CMP gather. Notice how the multiples are “picked” cleanly out of the data, while strong primaries are left largely intact. Denoting the crosstalk weights for each \( m_{i,k} \) as a vector \( w_{i,k} \), we can write the model residual corresponding to the third model regularization operator:

\[
r^{[3]}_m(\tau, x, i, k) = w_{i,k}(\tau, x) m_{i,k}(\tau, x).
\]  

Figure 2: Synthetic CMP gather with and without crosstalk weights for \( k = 0 \) and \( j = 1,2,3 \) applied.

Although the crosstalk weights will likely overlap some primaries, the primaries’ flatness ensures that regularization operators (5) and (6) “spread” redundant information about the primaries from other \( m_{i,k} \) and other offsets to compensate for any losses.

**Combined Data and Model Residuals**

To solve equation (3) for the optimal set of \( m_{i,k} \), we minimize a quadratic objective function, \( Q(m) \), which consists of the sum of the weighted \( \ell_2 \) norms of a data residual [equation (4)] and of three model residuals [equations (5), (6), and (7)].

\[
\min Q(m) = \|r_d\|^2 + \epsilon_1^2 \|r^{[1]}_m\|^2 + \epsilon_2^2 \|r^{[2]}_m\|^2 + \epsilon_3^2 \|r^{[3]}_m\|^2
\]

\( \epsilon_1, \epsilon_2, \) and \( \epsilon_3 \) are scalars which balance the relative weight of the three model residuals with the data residual. In practice I suggest setting \( \epsilon_1 = 2.0, \epsilon_2 = 1.0, \) and \( \epsilon_3 = 2.0 \). I use the conjugate gradient method to minimize \( Q(m) \); the method is well-suited for large-scale least-squares optimization problems like this one.

**EXTENSION TO NON-SEABED PEGLEGS**

Reflectors other than the seabed also produce pegleg multiples, and the strength of these events sometimes rivals, and even surpasses, the strength of the seabed peglegs. The Mississippi
Canyon dataset, with its shallow tabular salt body, falls into this category. Figure 3 shows the stack of this data, with important reflectors and multiples labeled.

Computationally, the extension of this joint imaging method to include non-seabed peglegs is straightforward. We need only add an index for the multiple generating layer \( (m) \) to equation (2), where we model \( n_{surf} \) multiple-generating layers in the data.

\[
d = N_0 m_0 + \sum_{i=1}^{p} \sum_{k=0}^{n_{surf}} \sum_{m=1}^{n_{surf}} R_{i,m} N_{i,k,m} S_{i,m} G_{i,m} m_{i,k,m}
\] (9)

Physically, however, modeling non-seabed peglegs requires some additional thinking. Notice that all modeling operators in equation (9) have an extra index. Brown (2003a) discusses how to extend the HEMNO operator \( (N_{i,k,m}) \) to non-seabed peglegs, while Brown (2003b) discusses how to extend \( R_{i,m}, S_{i,m}, \) and \( G_{i,m} \). The regularization operators all remain the same, though non-seabed peglegs add additional crosstalk energy that must be modeled when the crosstalk weights are generated.

RESULTS

In 1997, WesternGeco distributed a 2-D test dataset, acquired in the Mississippi Canyon region of the Gulf of Mexico, for the testing of multiple suppression algorithms. The data contain a variety of strong surface-related multiples, and enough geologic complexity to render one-dimensional methods ineffectual. Figure 3 shows the stack of the raw data. The geologic setting of the area is a fairly well-behaved deepwater sedimentary basin, interrupted between 4000 and 16000 m by a shallow tabular salt body.

In addition to the seabed peglegs, peglegs from the top of salt and strong reflector R1 are included in the inversion. Rather than including each of these surfaces separately, the R1 and top salt events are assumed to arise from a single reflector; from midpoint 0 m to roughly 6000 m, it is R1, while from 6000 m to 20000 m it is the top of salt. Only first order multiples are included in the inversion. Thus in equation (9), \( n_{surf} = 2 \) and \( p = 1 \).

I tested LSJIMP on 750 CMP gathers of the Mississippi Canyon dataset. Figure 4 illustrates the stack of the LSJIMP result. From the difference panel, note that important peglegs (TSPL and WBM) are almost entirely removed. Primaries (like PR) are not damaged. R1PM, and to a lesser extent TSPM, are removed effectively. Salt rugosity contributes negatively to the separation, by forming diffractions which are not modeled by HEMNO, and by violating HEMNO’s small reflector dip assumption. Much deep multiple energy remains. Some of it is likely internal salt multiples. Additionally, time imaging operators (like HEMNO) are notoriously poor at imaging subsalt events.

Figures 5 and 6 show the LSJIMP results at two midpoint locations (0 m and 14400 m, respectively). From the top rows of these figures, notice that most important peglegs are separated from \( m_0 \), while the primaries are preserved. The estimated seabed and R1/top of salt peglegs seem to match the events in the data, both in term of kinematics and amplitudes.
Figures 7 and 8 illustrate the superior performance of the HEMNO operator versus 1-D NMO when used in LSJIMP. The strong top of salt seabed pegleg in Figure 7, dipping over this midpoint range, is clearly better removed when HEMNO is used. Figure 8 illustrates a more insidious problem: because the 1-D NMO operator did not focus a pegleg at the correct time, it caused a spurious event to be manufactured at a slightly smaller time.

CONCLUSIONS

I introduced many refinements to my earlier least-squares joint imaging for multiples and primaries scheme (Brown, 2002), and denoted it LSJIMP for short. Two companion papers in this report (Brown, 2003b,a) derived operators which transform pegleg multiples into “copies
Figure 4: Stack comparison of Mississippi Canyon data before and after LSJIMP. All panels windowed in time from 3.5 to 5.5 seconds. Top: Raw data stack. Center: Stack of estimated primary image, $m_0$. Bottom: Stack of the subtracted multiples. Labeled events: PR - underlying primary; WBM - first seabed multiple; R1PL - seabed pegleg of reflector R1; R1PM - R1 pure surface multiple; TSPL - seabed pegleg of TSR (top of salt); BSPL - seabed pegleg of BSR (bottom of salt); TSPM - TSR pure surface multiple. BSTSPL - TSR pegleg of BSR.
Figure 5: Mississippi Canyon CMP 1 (y = 0 m). All panels NMO’ed with stacking velocity and windowed in time from 3.5 to 5.5 seconds. Top row (L to R): Raw data; Estimated primaries ($m_0$); Estimated non-primaries (difference). Center row (L to R): Raw data; Estimated total first order seabed multiple ($\sum_{k=0}^{1} R_{1,1} N_{1,k,1} S_{1,1} G_{1,1} m_{1,k,1}$); difference. Bottom row (L to R): Raw data; Estimated total first order “salt” (R1 at this location) multiple ($\sum_{k=0}^{1} R_{1,2} N_{1,k,2} S_{1,2} G_{1,2} m_{1,k,2}$); difference.
Figure 6: Mississippi Canyon CMP 540 ($y = 14400$m). All panels NMO’ed with stacking velocity and windowed in time from 3.5 to 5.5 seconds. Top row (L to R): Raw data; Estimated primaries ($m_0$); Estimated non-primaries (difference). Center row (L to R): Raw data; Estimated total first order seabed multiple ($\sum_{k=0}^{1} R_{1,1} N_{1,k,1} S_{1,1} G_{1,1} m_{1,k,1}$); difference. Bottom row (L to R): Raw data; Estimated total first order “salt” multiple ($\sum_{k=0}^{1} R_{1,2} N_{1,k,2} S_{1,2} G_{1,2} m_{1,k,2}$); difference.
Figure 7: LSJIMP stack comparison of HEMNO versus 1-D NMO operator. All panels windowed from 4.4 to 4.8 seconds in time; 14000 to 15000 meters in midpoint. From L to R: Raw data stack; Stack of estimated $m_0$ using HEMNO; Stack of estimated $m_0$ using 1-D NMO operator; HEMNO difference; 1-D NMO difference. Seabed pegleg from top of salt reflection is outlined in all panels.

Figure 8: LSJIMP stack comparison of HEMNO versus 1-D NMO operator. All panels windowed from 4.8 to 5.2 seconds in time; 9200 to 10200 meters in midpoint. From L to R: Raw data stack; Stack of estimated $m_0$ using HEMNO; Stack of estimated $m_0$ using 1-D NMO operator; HEMNO difference; 1-D NMO difference. The ovals highlight a nonexistent event “removed” from data due to 1-D NMO’s inferior performance over nonflat structure.

of their primary”. These operators enable LSJIMP to separate peglegs and primaries by exploiting the mutual consistency of events after imaging. LSJIMP produces good multiple separation results on the 2-D Mississippi Canyon multiples test dataset. Encouragingly, this approach looks well-suited – both in terms of physics and computation – to be applied to 3-D data under the industry’s current acquisition constraints (poor crossline shot coverage), which inhibit the multiple prediction ability of methods like Verschuur et al. (1992), which work well in 2-D.

ACKNOWLEDGEMENT

WesternGeco acquired and released the Mississippi Canyon dataset for public use.

REFERENCES


Multiple attenuation in the image space

Paul Sava and Antoine Guitton

ABSTRACT
Multiples can be suppressed in the angle-domain image space after migration. For a given velocity model, primaries and multiples have different angle-domain moveout and therefore can be separated using techniques similar to the ones employed in the data space prior to migration. We use Radon transforms in the image space to discriminate between primaries and multiples. This method has the advantage of working with 3-D data and complex geology. Therefore, it offers an alternative to the more expensive Delft approach.

INTRODUCTION
The current most robust multiple attenuation techniques exploit moveout discrepancies that exist between primaries and multiples (Foster and Mosher, 1992). For instance, for relatively simple geology, our well-trusted Normal Moveout (NMO) correction efficiently flattens the primaries and leaves the multiples curved. Then the primaries and multiples can be separated in the Radon domain. However, it has been recognized that NMO and Radon transforms are not optimal when complex wavefield propagation occurs in the subsurface. The main reason is that the moveout of primaries and multiples cannot be described with simple functions (parabolic or hyperbolic) anymore (Bishop et al., 2001). Therefore, more sophisticated methods are needed to perform the multiple attenuation.

One method that takes propagation effects into account is the Delft approach (Verschuur et al., 1992). This technique has the advantage of working with the surface data only and for any type of geology. Thus, it is often the method of choice for multiple attenuation in complex geology (Miley et al., 2001). To be accurate, the Delft method requires a very dense coverage of sources and receivers. If this condition is relatively easy to meet in 2-D, it becomes much more difficult to fulfill with 3-D surveys. New multiple prediction techniques are therefore developed to circumvent this limitation (van Dedem and Verschuur, 2001; Levin and Johnston, 2001).

A powerful multiple attenuation technique would be one that first takes the wavefield propagation into account, with whatever data we have, and then uses moveout discrepancies to remove multiples. To achieve this goal, we first propose using prestack depth migration as our imaging operator. Assuming that we have the correct velocity and an accurate migration scheme, we can then handle/image any type of complex geology very accurately. In this pro-

1email: paul@sep.stanford.edu, antoine@sep.stanford.edu
cess, both primaries and multiples are migrated, after which they are transformed to angle
gathers using standard techniques. In the angle domain, primaries are flat and multiples are
curved, mimicking the situation we have after NMO for simple geology. Finally, we pro-
pose mapping the angle gather into a Radon domain where the signal/noise separation can be
achieved. This method has the potential to work with 2-D or 3-D data as long as angle gath-
ers can be estimated. It is also much cheaper than the Delft approach, but it can still handle
complicated geologic media.

**ANGLE TRANSFORM**

Angle-domain common image gathers (ADCIGs) are decompositions of seismic images in
components proportional to the reflection magnitude for various incidence angles at the re-
flector. Given correct velocities and migration algorithms, primaries map into flat gathers and
multiples map into events with moveout.

Angle-gathers can be constructed by two classes of methods: data-space methods (de
Bruin et al., 1990; Prucha et al., 1999; Mosher and Foster, 2000), with reflectivity described
function of offset ray parameter, and image-space methods (Weglein and Stolt, 1999; Sava and
Fomel, 2003; Rickett and Sava, 2001), with reflectivity described function of scattering angle.
The moveout behavior of primaries and multiples are similar, irrespective of the method used
to construct them.

Figure 1 shows on the left a CMP gather for a model with flat reflectors and \( v(z) \) velocity.
Most of the events in the gather are multiples, and just a few of the top-most events are pri-
maries. On the right, Figure 1 shows an angle-domain CIG for the data in the left panel. The
primary events are imaged correctly and are flat, but the multiples are not imaged correctly
and have strong moveout.

Angle-domain common image gathers are useful for multiple suppression for several rea-
sons. First, events imaged with the wrong velocity show substantial moveout, which allows
us to discriminate between primaries imaged with correct velocity and multiples, imaged with
incorrect velocity. Second, angle-domain common image gathers describe the reflectivity at
the reflection point, independent, in principle, from the actual structure for which they are
computed, so they capture all 3-D propagation effects at every individual CIG.

**MULTIPLE SUPPRESSION**

Multiple attenuation methods using Radon transforms (RT) are popular and robust (Foster
and Mosher, 1992). These techniques use the moveout discrepancy between primaries and
multiples for discrimination. Usually, the multiple attenuation is carried out with Common
Midpoint (CMP) gathers after NMO correction (Kabir and Marfurt, 1999). Then, the NMOed
data are mapped with a parabolic Radon transform (PRT) into a domain where primaries and
multiples are separable.
One desirable property of a Radon transform is that events in the Radon domain be well focused. This property makes the signal/noise separation much easier and decreases the transformation artifacts. These artifacts come essentially from the null-space associated with the RTs (Thorson and Claerbout, 1985). The RTs can be made sparse in the Fourier domain (Hugonnet et al., 2001) or in the time domain (Sacchi and Ulrych, 1995). The Fourier domain approach has the advantage of allowing fast computation of the Radon panel (Kostov, 1990). However, the sparse condition developed so far (Hugonnet et al., 2001) does not focus the energy in the time axis. Therefore, in our implementation of the RTs, we opted for a time domain formulation with a Cauchy regularization in order to enforce sparseness.

A generic equation for a Radon transform in the angle domain is

$$z(q, \gamma) = z_0 + q \ g(\gamma),$$

where $z_0$ is the zero-angle depth, $\gamma$ is the scattering angle, $q$ is a curvature parameter, and $g(\gamma)$ is a function that represents the moveout in the CIGs. The modeling equation from the Radon
Figure 2: Synthetic example for S/N separation in the image space: (D) data in the image domain; (P) data in the Parabolic Radon domain; (N) multiples (noise); (S) primaries (signal).

The domain to the image domain and its adjoint are

\[
d(z, \gamma) = \sum_{z_0} \sum_{q} m(z_0, q) \delta \left[ z_0 - (z - q \cdot g(\gamma)) \right], \tag{2}
\]

\[
m(z_0, q) = \sum_{z} \sum_{\gamma} d(z, \gamma) \delta \left[ z - (z_0 + q \cdot g(\gamma)) \right]. \tag{3}
\]

At first order, we can assume that \( g(\gamma) = \gamma^2 \), which shows that Equation (1) corresponds to the definition of a parabola. However, for angle-domain common image gathers, Biondi and Symes (2003) demonstrate that a better approximation is \( g(\gamma) = \tan(\gamma) \cdot \gamma^2 \).

Equation (2) can be rewritten as

\[
d = Lm, \tag{4}
\]

where \( d \) is the image in the angle domain, \( m \) is the image in the Radon domain, \( L \) is the forward RT operator. Our goal now is to find the vector \( m \) that best synthesizes in a least-squares sense the data \( d \) via the operator \( L \). Therefore, we want to minimize the objective function:

\[
f(m) = \|Lm - d\|^2. \tag{5}
\]

We also add a regularization term that enforces sparseness in the model space \( m \). High resolution can be obtained by imposing a Cauchy distribution in the model space (Sacchi and...
Ulrych, 1995):

$$f(m) = \|Lm - d\|^2 + \epsilon^2 \sum_{i=1}^{n} \ln(b + m_i^2), \quad (6)$$

where $n$ is the size of the model space, $\epsilon$ and $b$ two constants chosen a-priori: $\epsilon$ controls the amount of sparseness in the model space and $b$ relates to the minimum value below which everything in the Radon domain should be zeroed. The least-squares inverse of $m$ is

$$\hat{m} = \left[ L'L + \epsilon^2 \text{diag}(1/(b + m_i^2)) \right]^{-1} L'd, \quad (7)$$

where $\text{diag}$ defines a diagonal operator. Because the model or data space can be large, we estimate $\hat{m}$ iteratively. The objective function in Equation (6) is non-linear because the model appears in the definition of the regularization term. Therefore, we use a limited-memory quasi-Newton method (Guitton and Symes, 1999) to find the minimum of $f(m)$. 

Figure 3: Radon transform of angle-domain CIGs using the parabolic equation (left) and the tangent equation (right). Though still an approximation, the tangent equation focuses better the events in the Radon domain.
From the estimated model \( m \), we separate the multiples from primaries in the Radon domain, using their distinct \( q \) values. We transform the multiples back to the image domain by applying \( L \), and subtract them from the input data to obtain multiple-free gathers.

**EXAMPLES**

Our first example corresponds to a synthetic model with flat reflectors and \( v(z) \) velocity. The left panel in Figure 2 is a representative CMP for this model. Most of the energy in the CMP is multiples. The right panel in Figure 2 depicts a corresponding CIG. Again, most of the energy in the gather is multiples, which have non-flat moveout that distinguishes them from the flat primaries.

Figure 2 shows from left to right: (D) the data = primaries + multiples, in the image space; (P) the data transformed to the Radon domain, where the flat primaries are represented in the vicinity of \( q = 0 \), in contrast to the multiples at non-zero \( q \); (N) the multiples isolated in the Radon domain and transformed back to the image domain; (S) the primaries left after subtraction of the multiples (N) from the data (D).

Figure 3 shows a comparison between RT using the parabolic equation \( g(\gamma) = \gamma^2 \) (left), and the more accurate tangent equation \( g(\gamma) = \tan(\gamma)^2 \) (right). Not surprisingly, we observe better focusing using the tangent equation, which makes it easier to isolate the multiples.

Figure 4 corresponds to a real dataset filled with multiples over a region with mildly varying velocity. Similarly to the case of the preceding synthetic example, we show: (D) the data = primaries + multiples, in the image space; (P) the data transformed to the PRT domain; (N) the multiples isolated from the PRT panel and transformed back to the image domain; (S) the primaries left after subtraction of the multiples (N) from the data (D).

Next, we present the case of a synthetic example over a salt dome model. Figure 5 follows the pattern used in the preceding two examples: from left to right, the data (D), the PRT domain (P), the noise (multiples) (N), and the signal (primaries) (S). Figure 6 shows the corresponding stacks before (D) and after (S) multiple suppression. Most of the multiple energy is removed from the image (around a depth of 16 kft).

We also apply our technique to a Gulf of Mexico dataset from a salt dome environment. This is a more complicated example, since it illustrates many of the difficulties encountered by multiple suppression in complicated areas, around salt bodies and in the presence of notable 3-D effects. Following the pattern used in the preceding example, Figures 7 and 8 show our multiple analysis at two different locations in the data. The first figure, corresponds to an area more or less away from the salt body, while the second one corresponds to a region right under the salt. From left to right, we present the data (D), the Radon domain (P), the noise (multiples) (N), and the signal (primaries) (S).

In both cases, primaries and multiples separate remarkably well in the Radon domain. We obtain the noise model after mute in the Radon domain and inverse RT, and the signal model by subtracting the noise from the data.
Figure 4: Field data example. Signal/Noise separation in the image space: (D) signal + multiples (data); (P) data in the Parabolic Radon domain; (N) multiples (noise); (S) primaries (signal); paul1-mDSN

For comparison, in both Figures 7 and 8 we include one more panel (C) which represents the same image gather obtained by migration of the signal obtained by multiple suppression in the data space using a high resolution HRT with Cauchy regularization. The image space multiple suppression creates cleaner CIGs, compared with the data space method, although some of the inherent noise associated with RT can be still observed in the image.

Figure 9 shows the stacks of the images obtained without multiple suppression (D), with multiple suppression in the image space (S), and with multiple suppression in the data space (C). Our image space method removes more of the multiple energy than the data space method.

**DISCUSSION**

Primaries and multiples can have shapes which are neither parabolic, nor hyperbolic in the data space. All multiple suppression strategies based on PRT, HRT or similar methods approximate the data moveout and may fail in complex areas. Furthermore, primaries and multiples often have comparable shapes which are hard to discriminate. Primaries and multiples have different shapes in the image space: primaries are mostly flat and multiples are non-flat. This allows, in principle, for robust signal/noise separation strategies.
For complex geology, the multiples are better attenuated if the propagation effects are taken into account. This is why the Delft approach performs so well (Verschuur et al., 1992). For 3-D data, the latter can be rather difficult to use because interpolating the sources and receivers on a regular grid is very expensive. Alternatively, we can migrate the data and do the separation in the image space with conventional Radon transforms, as we demonstrate in this paper.

Potential pitfalls for the multiple suppression strategy in the image space include situations where our velocity model is far from the truth. We encounter the theoretical possibility that some multiples are flat and some primaries are not flat. However, even in such situations, we can still discriminate primaries from multiples, given enough separation in the Radon domain.

Finally, we would like to point out that the strategy described in this paper, which is based on Radon transforms, is not ideal. Other more sophisticated signal/noise separation methods, e.g., methods based on patterns (Guitton et al., 2001; Haines et al., 2003), can better handle the non-parabolic shapes encountered in the image space thus producing better separation results.
CONCLUSIONS

Multiples can be suppressed in the angle-domain, after migration. For a given velocity model, primaries and multiples have different moveout in the image space, and therefore they can be separated using similar techniques as the ones employed in the data space, prior to migration. We use Radon transforms, although these methods are neither unique, nor ideal.

Because we are using prestack depth migration, this method takes into account the effects of complex wavefield propagation in the same way that the Delft approach does. However, our proposed scheme has the potential to be affordable with 3-D data. Therefore, for complex geology, this method stands between multiple attenuation in the data space with Radon transforms and the Delft approach where multiples are first predicted and then subtracted.

ACKNOWLEDGMENT

We thank Biondo Biondi for useful suggestions.

REFERENCES


Figure 7: Gulf of Mexico example. S/N separation in the image space: (D) signal + multiples (data); (P) data the Parabolic Radon domain; (N) multiples (noise); (S) primaries (signal) separated in the image space; (C) primaries (signal) separated in the data space.

[CR]


Figure 8: Gulf of Mexico example. S/N separation in the image space: (D) signal + multiples (data); (P) data the Parabolic Radon domain; (N) multiples (noise); (S) primaries (signal) separated in the image space; (C) primaries (signal) separated in the data space.


Figure 9: Gulf of Mexico example. Images obtained by angle-domain stacks: (D) signal + multiples (data); (S) primaries (signal) after separation in the image space; (C) primaries (signal) after separation in the data space.


Iar în lumea asta mare, noi copii ai lumii mici,
Facem pe pamantul nostru musuroaie de furnici;
Microscopice popoare, regi, osteni și invătăți
Ne succedem generatii și ne credem minunat;
Musti de-o zi pe-o lume mica de se masura cu cotul,
In acea nemarginire ne-nvătăm uitaand cu totul
Cum ca lume asta-ntreaga e o clipa suspendată,
Ca-ndaratu-i și-nainte intuneric se arată.

**Scrisoarea I**
**Mihai Eminescu (1850-1889)**
Multiple suppression in the angle domain with non-stationary prediction-error filters

Seth Haines, Antoine Guitton, and Paul Sava

ABSTRACT
Non-stationary Prediction Error Filters (PEF’s) present an effective approach for separating multiples from primaries in the angle domain. The choice of models to be used for estimation of the PEF’s has a substantial impact on the final result, but is not an obvious decision. Muting in the parabolic radon transform (PRT) domain produces an effective multiple model, but the corresponding primary model must be massaged in order to minimize remaining multiple energy and achieve a satisfactory result.

INTRODUCTION
The separation of a desired signal from unwanted noise can take many forms, but is a central problem throughout exploration geophysics. Various methods have been developed to address the issue, and we focus on those applicable to cases where the signal and the noise can be predicted with prediction error filters (PEF’s) (Soubaras, 1994; Abma, 1995; Nemeth, 1996). In particular, we consider the problem of multiples, and focus on their suppression in the angle domain.

We use the projection filtering method (Soubaras, 1994; Abma, 1995) for this paper. Our goal is to efficiently compare the effects of various model choices, so the small loss of mathematical rigor pointed out by Guitton (2003b) is made up for by the relative computational stability. The projection filtering technique is described in detail by Guitton (2003a). We review briefly: We consider data (d) to be the sum of unknown signal (s) and noise (n):

\[ d = s + n \]  \hspace{1cm} (1)

and we estimate non-stationary PEF’s S and N to be used as whitening operators for the unknown signal and noise, respectively. The signal-noise separation is based on the following fitting goals:

\[ 0 \approx N(s - d) \]  \hspace{1cm} (2)

and

\[ 0 \approx cSs. \]  \hspace{1cm} (3)
A noise model is not explicitly estimated in the inversion. It is considered to be the difference between the data and the estimated signal \( n = d - s \).

The successful application of this method to a particular problem requires careful choices of a number of different parameters and inputs. These include the damping/weighting factor \( \epsilon \) and the dimensions of the PEF’s. A particularly important decision is the choice of models to be used for the estimation of the PEF’s \( S \) and \( N \). Sava and Guitton (2003) use muting in the parabolic radon transforms (PRT) domain to discriminate between multiples and primaries in the angle domain, and we use their results as a starting point for a PEF-based approach here. Rather than simply subtracting the multiple model output by the PRT approach from the original data, we use the primary and multiple models to estimate PEF’s which are then used in the signal/noise separation technique outlined above. A series of synthetic and real data examples illustrate the effectiveness of the technique, and enable us to test various options for PEF estimation models.

**EXAMPLE 1: A SIMPLE SYNTHETIC ANGLE GATHER**

We begin with synthetic data for the simple one-dimensional case described by Sava and Guitton (2003). Sava and Guitton (2003) use muting in the PRT domain to separate input data (Figure 1a) into a primary model (Figure 1c) and a multiple model (Figure 1b). We are able to use this multiple model for estimation of the noise PEF’s, but find that the PRT-muted primary model does not produce satisfactory results as a model for signal PEF estimation. Because multiples and primaries overlap in some parts of the PRT domain (most importantly at near offsets), the primary model still contains some multiple energy, particularly in the deeper part of the record. We tested several model options for estimating the signal PEF: the raw PRT result; \( \text{Nd} \) (the convolution of the noise PEF’s with the original data such that the signal is considered to be anything in the data that is not removed by the noise PEF’s); and a windowed version of the PRT result. We know that the lower part of the record contains no primary energy so we create a signal model (Figure 1d) by windowing and zero-padding the PRT result. Using the windowed PRT result to estimate our signal PEF, we can remove nearly all of the multiple energy, leaving Figure 1f as our final estimated signal.

While this result is clearly better than the PRT result, it is hardly surprising given the simplicity of the situation and the choice of signal model. It does, however, illustrate the important point that with careful model choice we can cleanly separate primaries from multiples in the angle domain, and encourages us to consider more complicated cases. This example also illustrates an advantage of PEF’s over the PRT approach. The multiple model from the PRT approach fails to capture some of the steeply dipping parts of the multiple energy at the larger angles (Figure 1b), while this energy is captured by the PEF approach (Figure 1e). Although this energy was not in the model used for noise PEF estimation, the same part of the signal PEF estimation model contains only random noise, so the signal/noise separation places that energy in the output noise model. The result is a more thorough separation of signal from noise.
Figure 1: (a) Input data- a synthetic angle gather. (b) Multiple model estimated with PRT approach. (c) Primary model estimated from PRT approach. (d) Signal model used for PEF estimation. (e) Multiple model estimated with PEF approach. (f) Primary model estimated with PEF approach. [ER,M]
EXAMPLE 2: A MORE REALISTIC SYNTHETIC

We next apply non-stationary PEF’s to the synthetic dataset described by Sava and Guitton (2003). The model is designed to resemble a typical Gulf of Mexico subsalt problem.

Each angle gather is processed separately, but the same parameters are used for all gathers. A typical gather is shown in Figure 2a. For the noise PEF estimation we again use the multiple model estimated with the PRT approach (Figure 2b), and for the signal PEF we use a laterally smoothed version (Figure 2d) of the primary model estimated from the PRT approach (Figure 2c). The smoothing is accomplished by averaging across 8 traces in each direction from a given trace, and helps to weaken or remove the remaining multiple energy. This improves the ability of the signal PEF’s to accurately represent the pattern of the signal. The primary model estimated by the PEF approach is shown in Figure 2f, with the corresponding multiple model in Figure 2e. The result is clearly a much better separation of the multiples from the primaries in the angle domain.

A comparison of common-angle gathers (Figure 3) highlights the effectiveness of the non-stationary PEF approach. Figure 3a shows a window of the raw data at an angle of 15 degrees. Figure 3b shows the corresponding PRT result, with improved resolution of the deeper reflectors. And Figure 3c shows the PEF result, with considerable improvement over the PRT result, particularly of the reflector at 19 kft.

The real test of effectiveness, of course, is a comparison of the stacked results. A final stack of the raw data is shown in Figure 4a, the PRT result in Figure 4b, and the final PEF stack is shown in Figure 4c. The PEF result shows improvement in the clarity of reflectors in the lower part of the record.

We chose the smoothed PRT result as the model for estimation of the signal PEF after testing several options. The PRT primary model contains sufficient multiple energy that the PEF estimation is impacted significantly, and the final estimated signal contains considerable multiple energy as well. The use of $N_d$ presents similar limitations and similar problems with the final result, due to the inability of the noise PEF $N$ to completely remove the multiple energy. The smoothed version of the PRT primary model is the most effective model that we tested. The smoothing helps to remove a considerable amount of the remaining multiple energy and generally cleans up the model so that the PEF estimation is a simpler problem. This smoothing would not be a wise choice as a processing step on raw data, but it is acceptable for cleaning up the PEF-estimation model and results in PEF’s that are well-suited to task of separating primaries from multiples in this case.

EXAMPLE 3: GULF OF MEXICO DATA

Our third example applies non-stationary PEF’s to Gulf of Mexico subsalt data (Figure 7). This Mississippi Canyon 2-D dataset was released by Western Geco in 1997 for the SEG multiples workshop. Again we find that a smoothed version of the PRT result provides a better model for signal PEF estimation than the raw PRT result, and that the raw PRT multiple model provides
Figure 2: (a) Input data- one angle gather from the synthetic. (b) Multiple model estimated with PRT approach. (c) Primary model estimated from PRT approach. (d) Signal model used for PEF estimation (smoothed version of (c)). (e) Multiple model estimated with PEF approach. (f) Primary model estimated with PEF approach.
Figure 3: Common-angle gathers, at an angle of 15 degrees. (a) Raw data. (b) PRT result. (c) PEF result, with improved clarity of reflectors at 15 kft and 19 kft.

shaines1-BP_CA [ER,M]
Figure 4: Stacked images. (a) Stack of raw data. (b) Stack of PRT result. (c) Stack of PEF result. Note improved clarity of reflector at depth of 19 kft. [shaines1-BP_stack] [ER.M]
a satisfactory noise PEF estimation model. Two angle gathers are shown in Figures 5 and 6, from the salt area and from outside the salt body, respectively. In both cases, the PEF result produces a cleaner final gather. The stacked results (Figures 7, 8 and 9) illustrate the

![Figure 5: (a) Angle gather from offset of 12000 m (see Figure 7 for reference), in the area of the salt body. (b) Multiple model from PRT approach, used as noise model for PEF approach. (c) Primary model from PRT. (d) Smoothed version of (c), used as signal model for PEF estimation. (e) Multiple model output by PEF approach. (f) Primary model output by PEF approach.](shaines1-CIG2_gom_nice)

effectiveness of the PEF technique. In addition to the raw data (Figures 7a, 8a and 9a), the PRT result (Figures 7b, 8b and 9b), and the PEF results described above (Figures 7d, 8d and 9d), we include also the result attained using Nd as the signal model (Figures 7c, 8c and 9c). The PEF results are clearly better than the raw stack, and also notably superior to the PRT result. The Nd approach is quite effective in this case, particularly in terms of maintaining higher frequency content of the data.
CONCLUSIONS

We have shown that non-stationary PEF’s provide an effective means for suppressing multiples in the angle domain. Muting in the PRT domain provides models for multiples and for primaries that can be used as models for PEF estimation. Minor adjustments, such as lateral smoothing, before PEF estimation can greatly improve the final result. The results presented here show that the use of these PRT multiple and primary models in the pattern-based approach described here achieves better results than simple subtraction of the multiple model from the original data.

The purpose of a model for PEF estimation is to produce a PEF that captures the essential aspects of the signal or the noise. In many cases, this model need not exactly resemble the signal or the noise, but instead may be more effective if it is a simplified version of that which it targets. The lateral smoothing used here would not be an acceptable treatment of the actual
Figure 7: (a) raw data. (b) Result from muting in the PRT domain. (c) PEF result, using Nd as the signal model. (d) PEF result using smoothed version of PRT primary model as signal model.
Figure 8: Windows of various final stack options, from beyond the margin of the salt body: (a) raw data. (b) Result from muting in the PRT domain. (c) PEF result, using $N_d$ as the signal model. Note higher frequency content than any of the other stacks. (d) PEF result using smoothed version of PRT primary model as signal model.

data since it alters the frequency content and removes some amplitude information. It does, however, produce a model from which an effective signal PEF can be estimated. It is worth noting that this smoothing is akin to the stacking which is the final processing to be applied to each angle gather. This application to PEF estimation models, as part of the signal/noise separation approach described here, allows us to remove significant multiple energy prior to the final stack and produces an improved image. The cost of this smoothing, however, is a loss of some higher frequency energy.

REFERENCES


Guitton, A., 2003a, Multiple attenuation with multidimensional prediction-error filters: SEP–
Figure 9: Windows of various final stack options, from beneath the salt body: (a) raw data. (b) Result from muting in the PRT domain. (c) PEF result, using Nd as the signal model. (d) PEF result using smoothed version of PRT primary model as signal model.


Multiple attenuation with multidimensional prediction-error filters

Antoine Guitton

ABSTRACT

Multiple attenuation in complex geology remains a very intensive research area. The proposed technique aims to use the spatial predictability of both the signal (primaries) and noise (multiples) in order to perform the multiple attenuation in the time domain. The spatial predictability is estimated with multidimensional prediction-error filters. These filters are time-variant in order to handle the non-stationarity of the seismic data. Attenuation of surface-related multiples is illustrated with field data from the Gulf of Mexico with 2-D and 3-D filters. The 3-D filters allow the best attenuation result. In particular, 3-D filters seem to cope better with inaccuracies present in the multiple model for short offset and diffracted multiples.

INTRODUCTION

The last decade has seen an exponential growth in the use of 3-D seismic imaging methods. Contemporaneous with this development, imaging techniques have become more complex in the effort to account for multi-pathing in complex media and to produce true amplitude migrated pictures of the subsurface. Since multiples are not accounted for in the physical model that leads to these migration methods, they can severely and adversely affect the final migration result producing erroneous interfaces or amplitude artifacts; consequently, the multiples have to be removed from the data prior to any imaging attempt.

As pointed out by Weglein (1999), the multiple attenuation techniques may be divided into two families: (1) filtering methods which exploit the periodicity and the separability (move-out discrepancies) of the multiples and (2) the wavefield methods, where the multiples are first predicted, for example by autoconvolution of the recorded data, and then subtracted (Verschhur et al., 1992; Dragoset and MacKay, 1993; Weglein et al., 1997). Traditionally, filtering techniques are the method of choice for multiple processing because of their robustness and cost. However, because these techniques are mainly 1-D methods, they do not extend their multiple attenuation properties very well to higher dimensions, i.e., 2-D or 3-D. Therefore, filtering techniques have some limitations when tackling multiples in complex media. For example predictive deconvolution in the ray parameter domain fails when the water bottom is not flat (Treitel et al., 1982).

1email: antoine@sep.stanford.edu
In this paper I present results of a multiple attenuation technique based on the spatial predictability of both primaries and multiples. The attenuation is based on the assumption that primaries and multiples have different patterns and amplitudes. The pattern is estimated with time-space domain (t-x) multidimensional prediction-error filters (PEFs). In the first section following this introduction, I present the multiple attenuation technique. In the second section I illustrate the proposed method with a Gulf of Mexico field data example. I will show that 3-D PEFs give the best noise attenuation result. More specifically, 3-D PEFs are able to attenuate diffracted multiples better and are also less sensitive to modeling inaccuracies at short offset.

THEORY OF MULTIPLE ATTENUATION AND FILTER ESTIMATION

Recently Spitz (1999) and Brown and Clapp (2000) presented a noise attenuation technique based on patterns. The main idea is simple: the noise and signal have different multidimensional spectra that PEFs can approximate (Claerbout, 1992) in order to perform the separation. In this approach, the multiple attenuation is similar to Wiener filtering (Abma, 1995). One important approximation is that the noise and signal are uncorrelated so that the cross-spectrum between these two components is not needed. In this section, I show how the separation is performed and how the PEFs are estimated.

Multiple attenuation

First, I consider that any seismic data is the sum of signal and noise as follows:

$$d = s + n,$$

where \(d\) are the seismic data, \(s\) the signal we want to preserve and \(n\) the noise we wish to attenuate. In the multiple elimination problem, the noise is the multiples and the signal the primaries.

Now, assuming that we know the multidimensional PEFs \(N\) and \(S\) for the noise and signal components, respectively, we have

$$N n \approx 0$$

$$S s \approx 0$$

by definition of the PEFs. Equations (1) and (2) can be combined to solve a constrained problem to separate signal from spatially uncorrelated noise as follows:

$$0 \approx r_n = N n$$

$$0 \approx r_s = S s$$

subject to \(d = s + n\)

We can easily eliminate \(n\) in the last equation of the fitting goal (3) by convolving with \(N\). Doing so, we end up with the following fitting goals:

$$0 \approx r_d = N s - N d$$

$$0 \approx r_s = S s$$
For some field data, it might be useful to add a masking operator on the data and signal residual $r_d$ and $r_s$ in order to perform the noise attenuation. It happens for example when the noise appears after a certain time or offset. I call $M$ this masking operator and I weight the fitting goals in equation (5) as follows:

$$
0 \approx r_d = M(Ns - Nd) \\
0 \approx r_s = MSs
$$

Solving for $s$ in a least-squares sense lead to the objective function

$$
f(s) = \|r_d\|^2 + \epsilon^2 \|r_s\|^2
$$

where $\epsilon$ is a constant to be chosen a-priori that relates to the signal/noise ratio. The least-squares inverse for $s$ becomes

$$
\hat{s} = (N'MN + \epsilon^2 S'MS)^{-1} N'MNd,
$$

where (') stands for the adjoint. Note that since $M$ is a diagonal operator of zeros and ones, we have $M'M = M^2 = M$. It is interesting to note that $N'MN$ is the inverse spectrum of the noise and $S'MS$ is the inverse spectrum of the signal where we perform the attenuation. Soubaras (1994) uses a very similar approach for random noise and more recently for coherent noise attenuation (Soubaras, 2001) with F-X PEFs. Because the size of the data space can be quite large, we estimate $s$ iteratively with a conjugate-gradient method. In the next section, I describe the PEFs estimation method I use to compute $N$ and $S$ needed in equation (7).

**Filter estimation**

The PEFs I estimate are time domain non-stationary filters to cope with the variability of seismic data with time and offset. I describe in the appendix how the PEFs are estimated. From the fitting goals in equation (A-20), we have

$$
0 \approx r_y = YKa + y \\
0 \approx r_a = Ra
$$

where $Y$ is a matrix for non-stationary combination, $K$ is a masking operator, $a$ a vector of the unknown PEFs coefficients, $y$ the data vector from which we want to estimate the PEFs and $R$ a regularization operator. I describe each of these elements in more detail in the appendix.

Often with seismic data, the amplitude varies across offset and time. These amplitude variations can be troublesome when we want to use least-squares inversion because they tend to bias the final result (Claerbout, 1992). Therefore, it is important to make sure that the amplitude variation does not affect our processing. One solution is to apply a weight to the data like Amplitude Gain Control (AGC) or a geometrical spreading correction. However, a better way is to incorporate the weight inside our inversion by weighting the residual (Guitton, 2003). Introducing a weighting function $W$ in the PEFs estimation, we have the following fitting goals:

$$
0 \approx r_y = W(YKa + y) \\
0 \approx r_a = Ra
$$
As shown by Guitton (2003), this weighting improves the signal/noise separation results. The choice of the weighting function will be discussed later for the multiple attenuation example. This weight can also be different for the noise and signal PEFs. If we want to find \( a \) in a least-squares sense, we have to minimize the objective function

\[
f(a) = \|r_y\|^2 + \epsilon^2 \|r_a\|^2
\]

which leads to the least-squares estimate of \( a \)

\[
\hat{a} = -(K'Y'W^2YK + \epsilon^2 R'R)^{-1}K'Y'W^2y.
\]

Because we have many filter coefficients to estimate, \( a \) is estimated iteratively with a conjugate-gradient method.

Now, prior to the signal estimation in equation (5), \( S \) and \( N \) need to be computed from a signal and noise model, respectively. The multiple model is derived by autoconvolving the recorded data (Verschuur et al., 1992). I then obtain a prestack model of the multiples that I use to estimate the bank of non-stationary PEFs \( N \).

It is important to keep in mind that at this stage, I assume that the relative amplitude of all order of multiples is preserved. In theory, an accurate surface-related multiple model can be derived if (1) the source wavelet is known, (2) the surface coverage is big enough, and (3) all the terms of the Taylor series that model different orders of multiples are incorporated (Verschuur et al., 1992). In practice, however, a single convolution is performed (first term of the Taylor series) which leaves us with a multiple model with erroneous relative amplitude for high order multiples. In addition, the surface coverage might not be sufficient. This leaves us with wrong amplitudes for short offset traces and complex structures. Because PEFs estimate patterns, wrong relative amplitude can affect our noise estimation. However, as we shall see later, 3-D filters seem to better cope with noise modeling inadequacies.

The signal PEFs are more difficult to estimate since the signal is usually unknown. However, Spitz (1999) shows that for uncorrelated signal and noise, the signal PEFs can be expressed in terms of 2 PEFs: the PEFs \( D \), estimated from the data \( d \), and the PEF \( N \), estimated from the noise model such that

\[
S = DN^{-1}.
\]

Equation (12) states that the signal PEFs equal the data PEFs deconvolved by the noise PEFs. This deconvolution insures that the PEFs \( S \) and \( N \) will not span similar components of the data space. To avoid the deconvolution step suggested in equation (12), I apply the noise PEFs \( N \) to the data vector \( d \) and estimate the signal PEFs from the convolution result. This will give me an approximation of \( S \) that I later use for the noise attenuation (Spitz, 2001, personal communication).

Thanks to the Helix (Mersereau and Dudgeon, 1974; Claerbout, 1998), the PEFs can have any dimension. In this paper, I use 2-D and 3-D filters and demonstrate that 3-D filters lead to the best noise attenuation result. When 2-D filters are used, the multiple attenuation is performed for one shot gather at a time. When 3-D filters are used, the multiple attenuation is performed for one macro-gather at a time. A macro-gather is a cube made of fifty consecutive
shots with all the offsets and time samples. When the multiple attenuation is done, the macro-patches are reassembled to form the final result. Note that there is an overlap of five shots between successive macro-gathers. In the next section, I show a prestack multiple attenuation example with field data from the Gulf of Mexico.

A 2-D FIELD DATA EXAMPLE

This section demonstrates that the proposed multiple attenuation method yields an efficient attenuation of surface-related multiples for complex subsurface. The modeling of the multiples is made in 2-D and does not incorporate off-plane events. This dataset has been extensively used in a special edition of The Leading Edge (January 1999) on multiple attenuation. More recently Liu et al. (2000) shows the result of a plane-wave attenuation technique with the same dataset.

The multiple attenuation is carried out in the shot domain. I display in Figure 1 a near-offset section of the data. First, I estimate a weighting function $W$ for the filter estimation and a masking operator $M$ for the noise attenuation. For the masking operator, I manually picked the first surface-related multiple and applied a mute in the offset panel with the water velocity. Figure 2a shows a masking operator for one shot gather at location 4500 m. in Figure 1. The ones in Figure 2a show data points that are affected by the signal/noise separation. The zeros show data point that will be kept unchanged. For the noise and signal PEFs weighting operators, I first applied an AGC of 0.8 second (200 samples) on the shot gather for the data and the multiple model. From these gathers, I estimated a weighting function by dividing the gathers with AGC by the gathers without AGC, making sure that no division by zero would occur. Figure 2b displays the weighting function used for the noise PEFs estimation. Figure 2c shows the weighing function used for the signal PEFs estimation. Note that the weight for the signal PEFs is estimated from the data.

Having computed the masking and the weighting operators, we can proceed to the filter estimation and the multiple attenuation steps. The first step is to estimate the noise and signal PEFs, i.e., equation (11). Then, I estimate the signal, i.e. equation (7). I show the results of the multiple attenuation for three different locations in Figures 3, 4 and 5. In Figure 3b, we notice that the multiple model has weak amplitudes at short offset. This comes from the acquisition geometry at short offset (Dragoset and Jericevic, 1998). It is interesting to notice that the weighing function in Figure 2b for the same location is boosting-up the short offset traces to balance this effect. The same problem in the multiple model can be seen in Figures 4b and 5b.

For each result in Figures 3, 4 and 5 I compare the estimated signal when 2-D and 3-D filters are utilized for the signal/noise separation. For every gather, the 3-D filters yield a better noise attenuation result. It is interesting to see that the 3-D filters can handle modeling errors and diffracted multiples very well (Figure 5d). Below the salt, e.g, Figure 4, the 3-D attenuation is very good at short and far offset. I show in Figure 6 time slices of the time/offset/shot cube in which the multiple attenuation is performed. As expected, 3-D filters (Figure 6d) attenuate the noise much better than 2-D filters (Figure 6c). It is quite remarkable that 3-D filters
perform so well in areas where the multiple model is known to be inaccurate. In particular, diffracted multiples and off-plane/3-D multiples are better attenuated (between offsets 2000 and 3000 m in Figure 6).

Now, I show the stacked section of the input data, the estimated signal with 2-D and 3-D filters in Figures 7, 8 and 9, respectively. The multiple attenuation with 3-D filters gives the best stacked results. With 3-D filters, most of the diffracted multiples are well attenuated, although they are not completely predicted in the multiple model. Therefore, having more dimensions for the filter greatly improves the multiple attenuation. It is also important to notice that as opposed to Liu et al. (2000), no internal mute or $f-k$ filtering are necessary when the 3-D filters are utilized. Figure 10 shows a close-up of the stacked sections for the data (Figure 10a), the estimated signal with 2-D filters (Figure 10b) and the estimated signal with 3-D filters (Figure 10c). The multiple attenuation with 3-D results clearly preserves the signal better than with 2-D filters.

![Figure 1: Near-offset section of the Gulf of Mexico dataset. Note the surface related multiples below three seconds.](antoine5-zero-offset.png)

**DISCUSSION-CONCLUSION**

I have presented a multiple-attenuation technique that aims to model and separate the noise and signal with time-variant, multidimensional prediction-error filters. These filters can be 2-D or 3-D thanks to the helical boundary conditions. The estimation of these filters incorporate weighting functions to cope with amplitude variations in the data and in the noise model. In addition, a masking operator is introduced in the signal/noise separation method in order to
Figure 2: (a) Masking operator $M$ used for the signal/noise separation. (b) Weighing operator used for the noise PEFS estimation. (c) Weighting operator used for the signal PEFS estimation.

preserve areas where no multiples are present. Tests with a field data example from the Gulf of Mexico prove that the multiple attenuation works much better when 3-D filters are utilized, as opposed to 2-D filters. These 3-D filters allow a better signal/separation in areas where the multiple model is known to be inaccurate, e.g., short offset traces, diffracted multiples and off-plane/3-D events.

REFERENCES


Figure 3: Multiple attenuation result at location 4500 m. (a) Input data. (b) Multiple model. (c) Signal estimated with 2-D filters. (d) Signal estimated with 3-D filters.
Figure 4: Multiple attenuation result at location 12000 m. (a) Input data. (b) Multiple model. (c) Signal estimated with 2-D filters. (d) Signal estimated with 3-D filters.
Figure 5: Multiple attenuation result at location 24000 m. (a) Input data. (b) Multiple model. (c) Signal estimated with 2-D filters. (d) Signal estimated with 3-D filters. [antoine5-comp.24000] [CR,M]
Figure 6: Time slices at 4.5 seconds. (a) Input data. (b) Multiple model. (c) Signal estimated with 2-D filters. (d) Signal estimated with 3-D filters. [antoinetime.comp.4.5] [CR,M]
Figure 7: Stack of the input data.

Figure 8: Stack of the signal estimated with 2-D prediction-error filters.


Figure 10: Close-up of the stacked section. (a) Input data. (b) Signal estimated with 2-D filters. (c) Signal estimated with 3-D filters.


APPENDIX A

ESTIMATION OF NONSTATIONARY PEFs

As shown by Margrave (1998), the generalization of stationary convolution to the non-stationary case can be done in two different ways called non-stationary convolution and non-stationary combination. As identified by Rickett (2001), our programs at the Stanford Exploration Project implement non-stationary combination. This section intends to explain in a matrix form the filter estimation step for non-stationary PEFs for both non-stationary convolution and combination.

Definitions

I call $A$ the convolution or combination operator with a bank of non-stationary filters. For the non-stationary convolution, the filters are in the column of $A_{\text{conv}}$ (one filter corresponds to one point in the input space) whereas for the non-stationary combination, the filters lie in the rows of $A_{\text{comb}}$ (one filter corresponds to one point in the output space). For the convolution matrix, I define $a_{i,j}$ as the $i^{th}$ coefficient of the filter for the $j^{th}$ data point in the input space. For the combination matrix, I define $a_{i,j}$ as the $j^{th}$ coefficient of the filter for the $i^{th}$ data point in the output space. Therefore, for the non-stationary convolution we have

$$A_{\text{conv}} = \begin{pmatrix} 1 & 0 & 0 & \vdots \\ a_{1,0} & 1 & 0 & \vdots \\ a_{2,0} & a_{1,1} & 1 & \vdots \\ a_{3,0} & a_{2,1} & a_{1,2} & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix} \quad (A-1)$$

and for the non-stationary combination we have

$$A_{\text{comb}} = \begin{pmatrix} 1 & 0 & 0 & \vdots \\ a_{1,1} & 1 & 0 & \vdots \\ a_{2,2} & a_{2,1} & 1 & \vdots \\ a_{3,3} & a_{3,2} & a_{3,1} & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix} \quad (A-2)$$

The size of both matrices is $(n \times m)$ where $n$ is the size of an output vector ($x$) and $m$ the size of an input vector ($y$) if

$$Ay = x \quad (A-3)$$

The helical boundary conditions allow to generalize this 1-D convolution to higher dimensions. We can rewrite equation (A-3) for the convolution as follows:

$$y_k = x_k + \sum_{i=1}^{\min(nf-1,k-1)} a_{i,(k-i)} x_{k-i} \quad (A-4)$$
where \( n_f \) is the number of filter coefficients, and for the combination

\[
y_k = x_k + \sum_{i=1}^{\min(n_f-1,k-1)} a_{i,k} x_{k-i}.
\]

(A-5)

In the next section, I show how the non-stationary PEFs are estimated.

Filter estimation

When PEFs are estimated, the matrix \( \mathbf{A} \) is unknown. If \( \mathbf{y} \) is the data vector from which we want to estimate the filters, we minimize the vector \( \mathbf{r}_y \) as follows:

\[
\mathbf{0} \approx \mathbf{r}_y = \mathbf{A}\mathbf{y}
\]

(A-6)

which can be rewritten

\[
\mathbf{0} \approx \mathbf{r}_y = \mathbf{Y}\mathbf{a},
\]

(A-7)

where \( \mathbf{Y} \) is the matrix representation of the non-stationary convolution or combination with the input vector \( \mathbf{y} \). The transition between equations (A-6) and (A-7) is not simple. In particular, the shape of the matrix \( \mathbf{Y} \) is quite different if we are doing non-stationary convolution or combination. For the convolution, we have

\[
\mathbf{Y}_{\text{conv}} = \begin{pmatrix} \mathbf{Y}^0_{\text{conv}} & \mathbf{Y}^1_{\text{conv}} & \mathbf{Y}^2_{\text{conv}} & \cdots \end{pmatrix}, \text{ where}
\]

(A-8)

\[
\mathbf{Y}^0_{\text{conv}} = \begin{pmatrix} y_0 & 0 & 0 & \cdots \\ 0 & y_0 & 0 & \cdots \\ 0 & 0 & y_0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \quad \mathbf{Y}^1_{\text{conv}} = \begin{pmatrix} 0 & 0 & 0 & \cdots \\ y_1 & 0 & 0 & \cdots \\ 0 & y_1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \quad \mathbf{Y}^2_{\text{conv}} = \begin{pmatrix} 0 & 0 & 0 & \cdots \\ 0 & y_2 & 0 & \cdots \\ \vdots & \vdots & \ddots & \ddots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}
\]

etc...

We see that for the convolution case, the \( \mathbf{Y}^k_{\text{conv}} \) matrices are diagonal operators, translating the need for one filter to be applied to one input point. The size of the matrix \( \mathbf{Y}_{\text{conv}} \) is \( (n \times (m \times n_f)) \) where \( n_f \) is the number of filter coefficients. Now, for the combination, we have

\[
\mathbf{Y}_{\text{comb}} = \begin{pmatrix} \mathbf{Y}^0_{\text{comb}} & \mathbf{Y}^1_{\text{comb}} & \mathbf{Y}^2_{\text{comb}} & \cdots \end{pmatrix}, \text{ where}
\]

(A-9)

\[
\mathbf{Y}^0_{\text{comb}} = \begin{pmatrix} y_0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \quad \mathbf{Y}^1_{\text{comb}} = \begin{pmatrix} 0 & 0 & 0 & \cdots \\ y_1 & y_0 & 0 & \cdots \\ 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \quad \mathbf{Y}^2_{\text{comb}} = \begin{pmatrix} 0 & 0 & 0 & \cdots \\ 0 & y_2 & y_1 & y_0 & \cdots \\ \vdots & \vdots & \vdots & \ddots & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots \end{pmatrix}
\]

etc...

We see that for the combination case, the \( \mathbf{Y}^k_{\text{combo}} \) matrices are row operators, translating the need for one filter to be constant for one output point. The size of \( \mathbf{Y}_{\text{comb}} \) is equal to the size of
\( Y_{\text{conv}} \). For the vector \( a \) in equation (A-7) we have

\[
\begin{pmatrix}
a_0 \\
a_1 \\
a_2 \\
\vdots \\
a_{n_f-1}
\end{pmatrix}
\] with

\[
a_k = \begin{pmatrix}
a_0 \\
a_1, k \\
a_2, k \\
\vdots \\
a_{n_f-1}, k
\end{pmatrix}
\]

where \( n_f \) is the number of coefficients per filter. This definition of \( a \) is independent of \( Y \). We might want to have one filter common to different input or output points instead of one filter per point. In that case, the matrix \( Y \) is obtained by adding successive \( Y^k \) matrices depending on how many points have a similar filter. Note that in the stationary case, for both the convolution and the combination case we have \( a_0 = a_1 = a_2 = \cdots = a \)

\[
Y a = (Y_0 + Y_1 + Y_2 + \cdots) a.
\]

(A-11)

Therefore, for the matrix \( Y \), we have to add all the \( Y^k \) matrices together. If we take advantage of the special structure of \( Y^k \) for the convolution and the combination, we obtain for the stationary case

\[
A y = Y a = \begin{pmatrix}
y_0 & 0 & 0 & 0 & \vdots \\
y_1 & y_0 & 0 & 0 & \vdots \\
y_2 & y_1 & y_0 & 0 & \vdots \\
y_3 & y_2 & y_1 & y_0 & \vdots \\
\vdots & \vdots & \vdots & \vdots & \vdots
\end{pmatrix}
\begin{pmatrix}
1 \\
a_1 \\
a_2 \\
a_3 \\
\vdots
\end{pmatrix},
\]

(A-12)

which is the matrix formulation of the stationary convolution.

With the definitions given in equations (A-8), (A-9) and (A-10), the fitting goal in equation (A-7) can be rewritten

\[
0 \approx r_y = Y^0_{\text{conv}} a_0 + Y^1_{\text{conv}} a_1 + Y^2_{\text{conv}} a_2 + \cdots
\]

(A-13)

or

\[
0 \approx r_y = Y^0_{\text{comb}} a_0 + Y^1_{\text{comb}} a_1 + Y^2_{\text{comb}} a_2 + \cdots
\]

(A-14)

Each vector \( a_k \) has one constrained coefficient. We can then rewrite equations (A-13) and (A-14) as follows:

\[
0 \approx r_y = y + Y^0_{\text{conv}} M a_0 + Y^1_{\text{conv}} M a_1 + Y^2_{\text{conv}} M a_2 + \cdots
\]

(A-15)

and

\[
0 \approx r_y = y + Y^0_{\text{comb}} M a_0 + Y^1_{\text{comb}} M a_1 + Y^2_{\text{comb}} M a_2 + \cdots
\]

(A-16)

with

\[
M = \begin{pmatrix}
0 & 0 & 0 & \vdots \\
0 & 1 & 0 & \vdots \\
0 & 0 & 1 & \vdots \\
\vdots & \vdots & \vdots & \vdots
\end{pmatrix}
\]

(A-17)
The definition of $\mathbf{M}$ assumes that the first coefficient of each filter is known. Note that $\mathbf{M}$ is equal for both convolution and combination methods. Having defined the matrix $\mathbf{M}$, we can now rewrite equation (A-7) as follows:

$$0 \approx \mathbf{r}_y = \mathbf{YK}a + \mathbf{y}$$

(A-18)

where the square matrix $\mathbf{K}$ is

$$\mathbf{K} = \begin{pmatrix} \mathbf{M} & 0 & 0 \\ 0 & \mathbf{M} & 0 \\ 0 & 0 & \mathbf{M} \\ \vdots & \vdots & \vdots \end{pmatrix}.$$  

(A-19)

The next step consists of estimating the filter coefficients in a least-squares sense.

**Regularization of the filter coefficients**

The number of coefficients to estimate is usually much greater than the number of data points. This makes the problem very under-determined. A solution is to introduce more equations in equation (A-18) as follows:

$$0 \approx \mathbf{r}_y = \mathbf{YK}a + \mathbf{y}$$

$$0 \approx \mathbf{r}_a = \mathbf{Ra}.$$  

(A-20)

The second term in equation (A-20) improves the conditioning of our problem and is called regularization. In the filter estimation problem, it is reasonable that $\mathbf{R}$ penalizes strong variations between filter coefficients. Hence, $\mathbf{R}$ is usually a gradient or a Laplacian. Crawley (2000) proposes smoothing the filter coefficients along radial directions. This proposal is valid for shot or common midpoint gathers only where constant dips are roughly aligned along radial segments. For instance, if $\mathbf{R}$ is a gradient operator we have

$$\mathbf{R} = \begin{pmatrix} \mathbf{I} & -\mathbf{I} & 0 \\ 0 & \mathbf{I} & -\mathbf{I} \\ 0 & 0 & \mathbf{I} \\ \vdots & \vdots & \vdots \end{pmatrix}.$$  

(A-21)

with $\mathbf{I}$ the identity matrix. Equation (A-20) can be solved for $\mathbf{a}$ in a least-squares sense. We then want to minimize the objective function

$$f(\mathbf{a}) = ||\mathbf{r}_y||^2 + \epsilon^2||\mathbf{r}_a||^2$$  

(A-22)

which gives for $\mathbf{a}$ the least-squares estimate

$$\hat{\mathbf{a}} = -(\mathbf{K}'\mathbf{Y}\mathbf{YK} + \epsilon^2\mathbf{R}'\mathbf{R})^{-1}\mathbf{K}'\mathbf{Y}'\mathbf{y}.$$  

(A-23)

Because of the number of unknowns and of the sparseness of the problem, we use a conjugate gradient method to estimate our PEFs.
Source-receiver migration of multiple reflections

Guojian Shan¹

ABSTRACT
Multiple reflections are usually considered to be noise and many methods have been developed to attenuate them. However, similarly to primary reflections, multiple reflections are created by subsurface reflectors and contain their reflectivity information. We can image surface related multiples, regarding the corresponding primaries as the sources. Traditional source-receiver migration assumes that the source is an impulse function. I generalize the source-receiver migration for arbitrary sources, and apply it to the migration of multiple reflections. A complex synthetic dataset is used to test the theory. Results show that my multiple migration algorithm is effective for imaging the multiple-contaminated data.

INTRODUCTION
Multiple reflections are traditionally regarded as noise and accordingly attenuated (Verschuur and Berkhout, 1997; Guitton et al., 2001). However, some works have treated multiple reflections as signal and tried to image them. Reiter et al. (1991) images deep-water multiples by applying a Kirchhoff scheme. Sheng (2001) migrates multiples in CDP data by applying cross-correlogram migration. Berkhout and Verschuur (1994) and Guitton (2002) image the multiples with shot-profile migration. Brown (2002) jointly images the primaries and multiples with least-square methods. In this paper, I present source-receiver migration for multiples. I calculate a pseudo-primary gather by cross-correlating the primaries with the corresponding multiples at the surface. A traditional source-receiver migration algorithm is then run without any change on the pseudo-primary data to get the image.

Biondi (2002) derived the equivalence between shot-profile migration and source-receiver migration, given the assumption that the source is an impulse function, the imaging condition is cross-correlation, and the one-way wave equation downward continuation method is used for wavefield propagation. Shan and Zhang (2003) generalized the traditional source-receiver migration for arbitrary sources, and demonstrated the equivalence between shot-profile migration and source-receiver migration. As a special case of generalized source-receiver migration, multiple migration has a complicated source—the primary reflection wavefield—and multiple migration provides a good numerical test for the equivalence between shot-profile migration and source-receiver migration.

¹email: shan@sep.stanford.edu
In this paper, I review the theory of generalized source-receiver migration and give the algorithm to create the pseudo-primary data for multiple migration. I present poststack and prestack multiple migration of a 2-D synthetic data, and compare the migration result with the migration of original data.

THEORY

Traditional source-receiver migration assumes that the source is an impulse function and the migration is based on survey sinking. Source-receiver migration downward continues the CMP gather $P(x, h, z = 0, \omega)$ into the subsurface by the Double Square Root equation

$$\frac{\partial}{\partial z} P = \left( \frac{i \omega}{v_s} \sqrt{1 + \frac{v_s^2}{\omega^2} \frac{\partial^2}{\partial x_s^2}} + \frac{i \omega}{v_r} \sqrt{1 + \frac{v_r^2}{\omega^2} \frac{\partial^2}{\partial x_r^2}} \right) P,$$

where $x$ is midpoint, $h$ is half-offset, $x_s$ is the source point, $x_r$ is the receiver point, and velocity $v_s = v(x_s, z)$ and $v_r = v(x_r, z)$. It images by extracting the wavefield at zero subsurface offset $P(x, h = 0, z, \omega)$ and adding along all frequencies. In generalized source-receiver migration, instead of using the CMP gather of the recorded data directly, the cross-correlation between the source wavefield $D(x_D, z = 0, \omega, s)$ and the receiver wavefield $U(x_U, z = 0, \omega, s)$ at the surface is extrapolated into the media (Shan and Zhang, 2003). Namely, the wavefield to be downward continued by the Double Square Root equation is

$$P(x, h, z = 0, \omega) = \sum_s U(x_U, z = 0, \omega, s) \bar{D}(x_D, z = 0, \omega, s)$$

where $x = (x_U + x_D)/2$, $h = (x_U - x_D)/2$ and $s$ means an areal shot. When the source is an impulse function, the cross-correlation between source and receiver wavefields is exactly the CMP gather of the recorded data and the generalized source-receiver migration algorithm is exactly the same as the traditional source-receiver migration.

Source-receiver migration of multiple reflections is a special case of generalized source-receiver migration, in which the source wavefield is the primary reflection and the receiver wavefield is the corresponding multiple reflection. The wavefield to be downward continued is the cross-correlation between primaries and the corresponding multiples. Since it behaves very similarly to a primary, I call it pseudo-primary data. There are two steps for source-receiver migration of multiples. First, pseudo-primary data are calculated by cross-correlating the primary reflections with the corresponding multiples at the surface. Second, a traditional source-receiver migration is run on the pseudo-primary data. Figure 1 illustrates the principle of source-receiver migration of multiples. The phase of the trace at $(x, h)$ in the pseudo-primary data is exactly the same as a trace at $(x, h)$ from the CMP gather of primary, if we would have put a source at $x_D$ and a receiver at $x_U$.

Zero-offset data is very important in amplitude work, but it is never recorded in a real survey. The zero-offset dataset can be obtained from the pseudo-primary data very easily. In equation (2), letting $x_U = x_D = x$, we can get the zero-offset surface dataset

$$P(x, h = 0, z = 0, \omega) = \sum_s U(x, z = 0, \omega) \bar{D}(x, z = 0, \omega).$$
Figure 1: Left: Two traces in originally recorded data. The trace at $x_D$ records the primary reflection traveltime $t_1$ of $x_S \to R_1 \to x_D$. The trace at $x_U$ records the multiple reflection traveltime $t_1 + t_2$ of $x_S \to R_1 \to X_D \to R_2 \to x_U$. Right: The trace of pseudo-primary data related to trace $x_D$ and $x_U$. The trace at $(x, h)$ is the cross-correlation between trace $x_D$ and trace $x_U$, where $x, h$ are mid-point and half offset of $x_D$ and $x_U$, respectively.

Figure 2: Left: One trace in original data. The trace at $x$ has two impulses. The first one records the traveltime of the primary reflection $t_1$: $x_S \to R_1 \to x$ and the second one records the traveltime of the multiple reflection $t_2$: $x_S \to R_1 \to x \to R_2 \to x$. Right: Zero-offset dataset of pseudo-primary data. The trace in the pseudo-primary data is the cross-correlation of the trace in the left figure with itself. The traveltime of the impulse in the trace is double the traveltime between $x$ and $R_2$. 
Figure 2 illustrates how to generate a zero-offset surface dataset from pseudo-primary data. The phase information of the zero-offset dataset of the pseudo-primary data is exactly the same as the zero-offset would be if we would have put a source and receiver at $x$.

**SYNTHETIC DATA EXAMPLE**

In this section, I test my theory of multiple migration on a modified version of the 2.5-D Amoco dataset (Etgen and Regone, 1998; Dellinger et al., 2000), which was also used for shot-profile migration of multiple reflections (Guitton, 2002). Figure 3 shows the velocity model of the Amoco dataset. In the survey, a 500 meter water layer is added and two finite-difference modelings with and without free surface conditions were done in order to extract the surface-related multiples (Guitton, 2002). In both of them, 32 shots are computed with a split-spread geometry.

Figure 3: The velocity model for the synthetic data.

**Pseudo-primary for multiple migration**

Figure 1 and Figure 2 show the algorithm to create the pseudo-primary data for multiple migration. I cross-correlate the primary+ multiple with the multiple at the surface and extract the zero-offset dataset and one-shot dataset for comparison.

Figure 4 shows the zero-offset dataset from both the originally recorded dataset and the pseudo-primary dataset. The zero-offset dataset is very coarse, since only 32 shots are recorded in the original dataset, while the zero-offset dataset from pseudo-primary data is continuous, since every trace in the originally recorded data can be an areal shot in the pseudo-primary data. Nevertheless, as shown in Figure 2, the zero-offset dataset from pseudo-primary data is very similar to the originally recorded data.

Figure 5 displays the comparison between one shot from originally recorded data and one from generated pseudo-primary data. Although the pseudo-primary shot is noisy, it has a similar structure to the shot from the original data.
Figure 4: Zero offset datasets. Left: The zero offset dataset of originally recorded data. Right: The zero offset dataset of pseudo-primary data.

Figure 5: One shot gather at 16,000m. Left: One shot gather of primary with multiples. Right: One shot gather of pseudo-primary data.
Migration for zero-offset multiple reflections

It is interesting that cross-correlation between primary reflections and multiple reflections can provide a real zero-offset dataset, which can be processed by poststack migration. Figure 6 displays the poststack migration of the zero-offset dataset of pseudo-primary data. The migration result is noisy because only one offset of the data has been used while usually all offsets of the data are stacked after NMO in poststack migration. But we can still interpret the subsurface structure from the image, including the top and bottom of the salt, and the flat reflector below the salt.

Source-receiver migration for multiple reflections

For comparison, I migrate the originally recorded data (primaries + multiples) with Fourier finite difference for the Double Square Root equation (Zhang and Shan, 2001). The migration result is presented in Figure 7. There are many hyperbolas because the recorded shots are very sparse. I migrated the multiple reflections using both the split-step method for the Double Square Root equation (Popovici, 1996) and the Fourier finite difference for the Double Square Root equation, in which the average velocity is used as the reference velocity. The migration results are presented in Figure 8 and Figure 9. The migration result of the multiple data (Figure 8,9) is similar to that of the primary data (Figure 7), although the latter is sharper and less noisy.

It is not easy to separate the multiples from the original data in practice. Also, separation costs a lot of computation time. Instead of cross-correlating the primary+multiple with the multiple, I cross-correlate the whole recorded data (primary+multiple) with itself and then run the source-receiver migration. Figure 10 presents the migration result. The cross-correlation between the primary reflection and itself does add some noise to the image. We can see the fake reflector, which is mainly caused by the cross-correlation between the water bottom reflection and the reflection below the water bottom in the primary. Nevertheless, the image is interpretable.
Figure 7: Migration result of originally recorded data.

Figure 8: Imaging of multiples by split-step of DSR.

Figure 9: Imaging of multiples by FFD of DSR.
The amplitude of the primary reflection at different locations and different times varies. Therefore the multiple reflections have sources of different amplitudes. It is important to do amplitude balancing for the pseudo-primary data before migration. I apply deconvolution (Claerbout, 1999) instead of cross-correlation to the surface data, namely the pseudo-primary data for the source-receiver migration are calculated by

$$P(x, h, z = 0, \omega) = \sum_s \frac{U(x_U, z = 0, \omega, s) \bar{D}(x_D, z = 0, \omega, s)}{D(x_U, z = 0, \omega, s) \bar{D}(x_D, z = 0, \omega, s) + \epsilon^2}.$$  

(4)

Figure 11 shows the migration result when deconvolution is used for creating pseudo-primary data. Besides the amplitude balancing, it also has better resolution.

**CONCLUSIONS**

I have shown the theory of source-receiver migration for multiple reflections, using the primaries as the source. The 2-D complex synthetic data test proves that multiples can be cor-
rectly imaged with the generalized source-receiver migration method and provide structural information of the subsurface. Using the original data instead of multiples as the receiver wavefield may add some noise, but the separation of the multiples from original data can be avoided. Since the source wavefield in multiple migration is not simple point source, it is a good numerical example for generalized source-receiver migration. The similar results of shot-profile migration and source-receiver migration of multiples numerically proves the equivalence between these two migration methods.

ACKNOWLEDGMENTS

I would like to thank Amoco for making the synthetic velocity model available. I would like also to thank Antoine Guitton and Morgan Brown for many useful discussions.

REFERENCES


Short Note

Prestack time imaging operator for 2-D and 3-D pegleg multiples over nonflat geology

Morgan Brown

INTRODUCTION

My Least-squares Joint Imaging of Multiples and Primaries (LSJIMP) algorithm (Brown, 2003b) separates pegleg multiples and primaries. LSJIMP computes separate images of the peglegs and primaries, and then uses the mutual consistency of the images to discriminate against unwanted noise types in each image. The images must be consistent in two respects: kinematics and amplitudes. A companion paper (Brown, 2003a) paper addresses the amplitude issue. In this paper, I address the kinematics. Kinematically, the events must be correctly positioned in time and flat with offset. To this end, I derive new time imaging operator, “HEMNO” (Heterogeneous Earth Multiple NMO Operator).

To correctly image primary or multiple reflections in a heterogeneous earth, migration reigns supreme, although its computational cost may be nontrivial over complex geology, in 3-D, or when a “true amplitude” image is required. My previously presented (Brown, 2002a) joint imaging technique used normal moveout (NMO), because of NMO’s speed and its amplitude predictability, despite its inability to account for nonflat geology. HEMNO correctly accounts for moderate subsurface structure but retains the speed and amplitude advantages of NMO. I show that the 2-D and 3-D pegleg moveout equations for two dipping planes derived by Levin and Shah (1977) and Ross et al. (1999), respectively, reduce to HEMNO in the small dip angle limit.

When used in conjunction with LSJIMP, I show that HEMNO produces improved results on the moderately complex Mississippi Canyon 2-D multiples test dataset, relative to a flat-earth NMO operator. Nevertheless, the real value of HEMNO lies in 3-D. Guitton (2003) demonstrates that the “Delft method” (Verschuur et al., 1992) (plus advanced multiple subtraction technology), almost perfectly separates surface-related multiples from 2-D data with complex 2-D structure. However, in real-world 3-D situations, acquisition and computational constraints diminish the method’s applicability. I demonstrate that in a simple, yet realistic, 3-D synthetic example that HEMNO can accurately image pegleg multiples from a seabed that dips in both directions. In the future, HEMNO should prove a useful advance in 3-D multiple separation.

1email: morgan@sep.stanford.edu
A first-order pegleg multiple consists of two unique arrivals: the event with a multiple bounce over the source, and the event with a bounce over the receiver. Figure 1 shows that in a flat earth, both “legs” of a pegleg (denoted S102G and S201G) arrive simultaneously; when the reflector geometry varies with position, they generally do not. In some cases, pegleg multiples are actually observed to “split”, though humans rarely observe the phenomenon unambiguously in field data, unless the reflector geometry is uniformly dipping over a large distance (see (Morley, 1982), for a good example).

The practical non-observation of split peglegs notwithstanding, geologic heterogeneity is a first-order effect in the accurate modeling of their kinematic and amplitude behavior. Mild variations in reflector depth over a cable length can introduce significant destructive interference between the legs of a pegleg multiple at far offsets – interference impossible to predict with a 1-D theory.

Levin and Shah (1977) deduced analytic kinematic moveout equations for 2-D pegleg multiples arising from two dipping layers, and Ross et al. (1999) extended the work to 3-D. Both approaches assume constant velocity and locally planar reflectors – assumptions which, depending on local geology, may be unrealistic in practice. In this section, I present “HEMNO” (short for Heterogeneous Earth Multiple NMO Operator), a simplified moveout equation based upon a more practically realizable conceptual model. In Appendix A, I show that the HEMNO equation is equivalent to Levin and Shah’s in the limit of small dip angle.

Brown (2002a) derived an extension to the conventional NMO equation which images pegleg multiples at the zero-offset traveltine of the target reflector:

\[
t = \sqrt{\left(\tau + j\tau^*\right)^2 + x^2/V_{\text{eff}}^2},
\]

where \(j\tau^*\) is the two-way traveltime of a \(j\)th-order pegleg in the multiple-generating layer. \(V_{\text{eff}}(\tau)\) is the effective RMS velocity of the equivalent primary shown in Figure 1.

\[
V_{\text{eff}}(\tau) = \frac{(j\tau^*V^*(\tau^*) + \tau V(\tau))}{(j\tau^* + \tau)}
\]

Figure 2 graphically illustrates HEMNO in a constant-velocity earth. The solid line in panel d) is the final result. It has the equation of a hyperbola with zero-offset traveltime \(\tau^*(y_0 - \Delta y)\)
Figure 2: Raypath for single-CMP approximate imaging operator. Panel a): True raypath in constant-velocity earth. The zero-offset traveltimes to the seabed and subsea reflector are $\tau^*(y_0)$ and $\tau(y_0)$, respectively. Panel b): Assumed reflection points under flat-earth assumption. $x_p$ is defined in equation (4). Panel c): The approximation. Stretch legs of raypath vertically to match measured $\tau^*(y_0-x_p/2)$ and $\tau(y_0+(x-x_p)/2)$. Panel d): Final step. Connect legs of raypath. The solid line that connects the reassembled raypath is the final result.

$$x_p/2 + \tau(y_0 + (x - x_p)/2)$$

and the same offset, $x$:

$$t^2 = \left[ \tau(y_0 - (x - x_p)/2) + \tau^*(y_0 - x_p/2) \right]^2 + \frac{x^2}{V_{\text{eff}}^2}$$

(3)

The following expression for $x_p$, the width of the pegleg’s primary leg in an assumed 1-D earth, is derived in (Brown, 2003a).

$$x_p^2 = \frac{x^2 \tau^2 V^4}{(\tau + j \tau^*)^2 V_{\text{eff}}^4 + x^2 (V_{\text{eff}}^2 - V^2)}$$

(4)

Although equation (3) was derived by seemingly *ad hoc* means, I show in Appendix A that the HEMNO raypath is a first-order approximation of Levin and Shah’s (1977) true raypath. The $V_{\text{eff}}$ shown in equation (3) is modified relative to that in equation (2): $\tau^*(y_0 - x_p/2)$ is substituted for $\tau^*$ and $\tau(y_0 - (x - x_p)/2)$ is substituted for $\tau$. 
Practical implementation of the HEMNO equation

To implement equation (3) on a computer, we must obtain two quantities. The first, the zero-offset traveltime of the seabed, \( \tau^*(y) \), may be obtained by hand- or auto-picking. Unfortunately, the second quantity, the zero-offset traveltime to an arbitrary subsea reflector at \( y = y_0 \pm (x - x_p)/2 \), cannot realistically be picked. Panel c) of Figure 2 motivates the problem; starting at \( \tau(y_0) \), the subsea reflector must be followed to \( y = y_0 + (x - x_p)/2 \). My approach to the problem is similar to Lomask and Claerbout’s (2002) algorithm for automatically flattening seismic data. It requires a smooth, unambiguous estimate of reflector dip. I summarize this approach (in 2-D; extension to 3-D is more involved but conceptually similar) in pseudocode:

Obtain zero offset section, \( d(\tau, y) \), by stacking input data.
Use \( d(\tau, y) \) to compute smooth reflector dip, \( p(\tau, y) \), using technique of Fomel (2002).
Set \( k = 1 \), \( y = y_0 \).
\[ \text{do while } \left( y + \Delta y \leq y_0 + (x - x_p)/2 \right) \]
\[ \text{Set } y = y_0 + (k - 1)\Delta y \]
\[ \text{Set } \tau(y + \Delta y) = \tau(y + \Delta y) + p(\tau(y), y) \]
\[ \text{Set } k = k + 1 \]
end do

This approach suffers from some possible pitfalls.

- In the “\( \tau^* \) to \( 2\tau^* \) zone”, \( d(\tau, y) \) consists of primary events only, and to a (normally) lesser extent, other coherent noise modes (e.g. locally-converted shear waves, internal multiples). For \( \tau > 2\tau^* \), \( d(\tau, y) \) contains both peglegs and primaries, as well as other possibly strong unmodeled multiple reflection modes, and the dip of these events is unlikely to be equal.

Previous authors (Fomel, 2001; Brown, 2002b) have developed methods to simultaneously estimate two crossing dips. The problem is inherently nonlinear, and thus highly dependent on initial guesses for the two dips. Assuming weak non-primary/pegleg noise, a possible strategy might be to seed the pegleg dip with primary dip from above.

In current implementation, however, I ignore this problem by setting the dip to zero in the \( \tau > 2\tau^* \) zone. I justify this assumption by noting that as \( \tau \) increases, the distance from \( y_0 \) to \( y_0 \pm (x - x_p)/2 \) shrinks asymptotically to zero, so unless the reflector dip is severe (inconsistent with the derivation of equation(3)), this omission will not present a large problem.

- Faults and other event discontinuities in \( d(\tau, y) \) are inconsistent with the “smooth” \( p(\tau, y) \) mentioned above. The current implementation ignores any possible faulting, although the dip estimation algorithm used can handle steeply folding reflectors.
EXAMPLES

In 1997, WesternGeco distributed a 2-D test dataset, acquired in the Mississippi Canyon region of the Gulf of Mexico, for the testing of multiple suppression algorithms. The data contain a variety of strong surface-related multiples, and enough geologic complexity to render one-dimensional methods ineffectual. Figure 3 shows the stack of the raw data.

Figures 4 and 5 illustrate the superior performance of the HEMNO operator versus 1-D NMO when used in LSJIMP. This figure is shown in better context in Brown (2003b). The strong top of salt seabed pegleg in Figure 4, dipping over this midpoint range, is clearly better removed when HEMNO is used. Figure 5 illustrates a more insidious problem: because the 1-D NMO operator did not focus a pegleg at the correct time, it caused a spurious event to be manufactured at a slightly smaller time.

Figure 6 compare the results of LSJIMP at a single midpoint (0 m), with and without HEMNO. The differences are fairly subtle, but illustrative. Although the dip in this region is mild, it is strong enough to cause the R1 pegleg multiple with $\tau = 4$ sec to defocus at far offsets. The thinly layered reflection is particularly susceptible to destructive interference with dip. On the center row of the figure, notice that in the HEMNO result, the defocusing effect is modeled, while in the 1-D result it is not. The difference panel further shows that HEMNO better models this effect.

Figure 7 illustrates the HEMNO operator applied to a synthetic 3-D dataset. The dataset consists of a seabed reflection and multiples only, with constant velocity and inline and crossline dips of 4° and 2°, respectively. Figure 7 shows a near-offset section after conventional NMO (left) and HEMNO for first-order multiples (right). The seabed multiple doesn’t “split”, but it is nonetheless treated as a pegleg by HEMNO, so this example does test HEMNO’s effectiveness in 3-D. Notice that over the entire survey area, the multiple and primary are mutually focused.

CONCLUSIONS

I introduced a new time-imaging operator for pegleg multiples, called HEMNO. HEMNO operates on single CMP gathers, making it computationally efficient and predictable in terms of amplitude behavior. I demonstrate that HEMNO can correctly account for moderate variation in reflector geometry with position, and that in a joint imaging context (LSJIMP), HEMNO accurately models both the vertical positioning of peglegs and some of the focusing effects with offset.

ACKNOWLEDGMENT

WesternGeco acquired and released the Mississippi Canyon dataset for public use.
APPENDIX A

Proof of Equivalence with Levin and Shah’s Traveltime Equations for Small Dips

In the following appendix, I show in that the approximate traveltime equation for pegleg multiples in a nonflat earth, equation (3), is a valid first-order approximation of Levin and Shah’s (1977) true raypath.
Figure 4: LSJIMP stack comparison of HEMNO versus 1-D NMO operator. All panels windowed from 4.4 to 4.8 seconds in time; 14000 to 15000 meters in midpoint. From L to R: Raw data stack; Stack of estimated $m_0$ using HEMNO; Stack of estimated $m_0$ using 1-D NMO operator; HEMNO difference; 1-D NMO difference. Seabed pegleg from top of salt reflection is outlined in all panels.

Figure 5: LSJIMP stack comparison of HEMNO versus 1-D NMO operator. All panels windowed from 4.8 to 5.2 seconds in time; 9200 to 10200 meters in midpoint. From L to R: Raw data stack; Stack of estimated $m_0$ using HEMNO; Stack of estimated $m_0$ using 1-D NMO operator; HEMNO difference; 1-D NMO difference. The ovals highlight a nonexistent event “removed” from data due to 1-D NMO’s inferior performance over nonflat structure.

Levin and Shah’s Traveltime Equations

Levin and Shah (1977) show that for a flat subsea reflector and dipping seabed, the moveout equation of the “S102G” (source-seabed-surface-reflector-receiver) pegleg multiple is:

$$(tV)^2 = \left[ \tau^* \cos \theta + \tau \cos \phi \right]^2 + \left[ \frac{x \cos (\phi + \theta)}{V} - \tau^* V \sin \theta - \tau V \sin \phi \right]^2,$$  \hspace{1cm} (A-1)

where $\phi$ and $\theta$ are the dip angle (in radians) of the seabed and subsea reflector, respectively. For small dip angles (i.e., less than 5 degrees), we can make the small angle approximation for angles $\phi$, $\theta$, and $\phi + \theta$ to update equation (A-1) accordingly:

$$(tV)^2 = \left[ \tau^* + \tau \right]^2 + \left[ x - \tau^* V \theta - \tau V \phi \right]^2.$$ \hspace{1cm} (A-2)

Multiplying out the squares in equation (A-2) and collecting terms gives:

$$t^2 = \left[ \tau^* + \tau \right]^2 + \frac{x^2}{V^2} - 2 \frac{\theta \tau^* x}{V} - 2 \frac{\phi \tau x}{V} + (\tau^* \theta)^2 + (\tau \phi)^2.$$ \hspace{1cm} (A-3)

The $\theta^2$ and $\phi^2$ terms are negligible for small angles, so we can ignore these terms and further simplify equation (A-3):

$$t^2 = \left[ \tau^* + \tau \right]^2 + \frac{x^2}{V^2} - 2 \frac{(\theta \tau^* + \phi \tau) x}{V}.$$ \hspace{1cm} (A-4)
Notice that equation (A-4) is equivalent to the previously derived equation (1) for a first-order pegleg, with the addition of an offset-dependent correction term for the dipping layers.

Although explicit seabed and subsea reflector dip angles, \( \phi \) and \( \theta \), are contained in equation (A-10), they were introduced only to show equivalence to equation (A-4). Locally-planar reflectors are not required to implement equation (3).

**Approximation to Levin and Shah’s Traveltime Equations for mild reflector dip**

In a constant-velocity medium, the expression for \( x_p \) derived earlier, equation (4), simplifies to:

\[
x_p = \frac{\tau}{\tau + \tau^*} x
\]  

(A-5)

Then \( x - x_p \), which will be needed later, simplifies to:

\[
x - x_p = \frac{\tau^*}{\tau + \tau^*} x
\]  

(A-6)

Using equations (A-5) and (A-6), we can directly write the (two-way) zero offset traveltime of the seabed and subsea reflection as a function of midpoint location, \( y_0 \):

\[
\tau^*(y_0 - x_p/2) = \tau^*(y_0) - \frac{x_p \sin \phi}{V}
\]

\[
\approx \tau^*(y_0) - \frac{\phi \tau(y_0) x}{V(\tau(y_0) + \tau^*(y_0))}
\]  

(A-7)

\[
\tau(y_0 - (x - x_p)/2) = \tau(y_0) - \frac{(x - x_p) \sin \theta}{V}
\]

\[
\approx \tau(y_0) - \frac{\theta \tau^*(y_0) x}{V(\tau(y_0) + \tau^*(y_0))}
\]  

(A-8)

where the small angle approximation was employed as before. Utilizing equations (A-5)-(A-8), we can now manipulate equation (3) to make it resemble Levin and Shah’s moveout equation (A-4):

\[
t^2 = \left[ \tau(y_0) + \tau^*(y_0) - \frac{(\phi \tau(y_0) + \theta \tau^*(y_0)) x}{V(\tau(y_0) + \tau^*(y_0))} \right]^2 + \frac{x^2}{V^2}
\]

\[
\approx \left( \tau(y_0) + \tau^*(y_0) \right)^2 - 2 \frac{(\phi \tau(y_0) + \theta \tau^*(y_0)) x}{V} + \frac{x^2}{V^2}
\]  

(A-9)

Equation (A-10) is equivalent to equation (A-4). Therefore, we have proven the equivalence of the moveout equations of the true and approximate raypaths shown in Figure 2, subject to the small dip angle approximation. As before, \( \phi^2 \) and \( \theta^2 \) terms were dropped in going from equation (A-9) to equation (A-10).

**REFERENCES**


Figure 6: Mississippi Canyon CMP 1 (y = 0 m). All panels NMO’ed with stacking velocity and windowed in time from 3.5 to 5.5 seconds. Top row (L to R): Raw data; Estimated HEMNO primaries; Estimated 1-D primaries; HEMNO difference; 1-D difference. Center row (L to R): Raw data; Estimated HEMNO total first order seabed multiple ($\sum_{k=0}^{1} R_{1,1} N_{1,k,1} S_{1,1} G_{1,1} m_{1,k,1}$); Estimated 1-D total first order seabed multiple; HEMNO difference; 1-D difference. Bottom row (L to R): Raw data; Estimated HEMNO total first order “salt” (R1 at this location) multiple ($\sum_{k=0}^{1} R_{1,2} N_{1,k,2} S_{1,2} G_{1,2} m_{1,k,2}$); Estimated 1-D total first order salt multiple; HEMNO difference; 1-D difference.
Figure 7: HEMNO demonstration on synthetic 3-D dataset. 8 swaths of 43 shots were acquired over a seabed reflector with 4 degrees of inline dip and 2 degrees of crossline dip. Acquisition was 3 streamers with 300 m crossline separation. Left panel shows the zero offset section after conventional NMO for primaries. Right panel shows zero offset section after HEMNO for multiples. Notice how, over all midpoints, the seabed reflections in each panel are coincident.

[morgan1-syn3d.comp] [CR,M]
Short Note

Amplitude modeling of pegleg multiple reflections

Morgan Brown

INTRODUCTION

My Least-Squares Joint Imaging of Multiples and Primaries (LSJIMP) algorithm (Brown, 2003a) separates pegleg multiples and primaries. LSJIMP computes separate images of the peglegs and primaries, and then uses the mutual consistency of the images to discriminate against unwanted noise types in each image. The images must be consistent in two respects. Kinematically, the events must be correctly positioned in time and flat with offset. This is accomplished by an improved normal moveout operator (HEMNO) introduced in a companion paper in this report (Brown, 2003b). This paper addresses the second aspect of image consistency: amplitudes.

LSJIMP requires that the amplitudes of the pegleg images be correct not in absolute terms, but relative to the primary image. Correct modeling of the relative amplitudes of pegleg multiples requires more work than a scaling by a reflection coefficient. For one, the reflection coefficient of the multiple-generating layer generally varies with position. Multiples also have longer raypaths through the earth, and thus suffer greater geometric spreading and anelastic attenuation losses than do primaries with the same offset. Furthermore, multiples and primaries have different reflection angles for a fixed offset, so the multiples and primaries will exhibit different amplitude-versus-offset (AVO) behavior.

In this paper, I present three innovations which transform pegleg multiples into events which are directly comparable to their corresponding primary reflections.

SNELL RESAMPLING REMOVES AVO/ATTENUATION DIFFERENCES

If we are modeling seabed pegleg multiples, Figure 1 illustrates the fact that (ignoring the seabed reflection) in a $v(z)$ medium, there exists a single offset $x_p$ such that a pegleg with offset $x$ and primary with offset $x_p$ are physically invariant with respect to AVO behavior and anelastic attenuation (water is assumed perfectly elastic). Ottolini (1982) introduced the concept of “Snell Traces” – a resampling of multi-offset reflection data along curves of constant time dip, or “stepout”. I adopt a similar line of reasoning to infer $x_p$ as a function of $x$. Since

---

1email: morgan@sep.stanford.edu
Figure 1: A primary and pegleg multiple with the same emergence angle ($\theta$) and midpoint ($y$). Note different offsets ($x$ and $x_p$) and a shift ($\Delta y$) in reflection point.

the pegleg multiple and primary in Figure 1 have the same emergence angle, $\theta$, the stepout of the two events is the same at $x$ and $x_p$. This fact is the basis of the derivation in Appendix A, which obtains the following result for $x_p$ as a function of $x$.

$$x_p^2 = \frac{x^2 \tau^2 V^4}{(\tau + j\tau^*)^2 V_{eff}^4 + x^2 (V_{eff}^2 - V^2)}, \text{ where}$$  

$$V_{eff}(\tau) = \frac{j\tau^* v^*(\tau^*) + \tau V(\tau)}{j\tau^* + \tau} \quad (1)$$

$j\tau^*$ is the two-way traveltime of a $j$th-order pegleg multiple in the top layer. Equation (1) defines a time-variable compression of the offset axis. In constant velocity, $V_{eff} = V$, and equation (1) reduces to the radial trace resampling used by Taner (1980) for long-period deconvolution of peglegs. Figure 2 demonstrates the Snell resampling on the first- and second-order pegleg multiples of a synthetic dataset.

Graphically (Figure 1), we infer that the shift in midpoint, $\Delta y$, of the reflection points of the primary and pegleg is

$$\Delta y = (x - x_p)/2. \quad (3)$$

As a function of time, $\Delta y$ decreases asymptotically to zero from a maximum of $x/4$ at the seabed. The deeper the reflector, the smaller $\Delta y$ becomes.

Oftentimes, non-seabed pegleg multiples (e.g. top of salt) are strong enough to merit modeling. In a $v(z)$ earth, the results derived in this section are equally valid, with one exception: attenuation. In this case, the effects of attenuation are encoded – in a possibly non-linear way – in the effective reflection coefficient that we estimate in a subsequent section.

**DIFFERENTIAL GEOMETRICAL SPREADING FOR PEGLEG MULTIPLES**

The Snell resampling transformation derived in the previous section renders pegleg multiples and primaries invariant with respect to AVO and attenuation. However, because of their longer raypaths, multiples suffer greater geometric spreading losses than the corresponding primary. Following previous authors (Ursin, 1990; Lu et al., 1999), I write offset-dependent geometric
Figure 2: Snell resampling demonstration. A synthetic dataset with flattened primaries and resampled first- and second-order peglegs. The transformation warps the vertical black lines horizontally, according to equation (1). Notice that the raw data has five unrecorded near offset traces and two dead traces at medium offsets. Snell resampling spreads information from the multiples into these no-data zones. The multiples provide a direct, additional constraint on the amplitude of the primaries where no data is recorded.

spreading corrections for a primary \( g_{\text{prim}} \) and its pegleg multiples \( g_{\text{mult}} \), respectively:

\[
\begin{align*}
  g_{\text{prim}} &= v^* t_{\text{prim}}(x) = \sqrt{(\tau v^*)^2 + \left(\frac{x v^*}{V}\right)^2} \\
  g_{\text{mult}} &= v^* t_{\text{mult}}(x) = \sqrt{[\tau v^*]^2 + \left(\frac{x v^*}{V_{\text{eff}}}\right)^2}.
\end{align*}
\]  

\( V_{\text{eff}} \) is defined in equation (2). After scaling by the ratio \( g_{\text{mult}}/g_{\text{prim}} \) and Snell resampling, a pegleg multiple should be an exact copy of its associated primary, to within a scaling by the appropriate reflection coefficient.
ESTIMATION OF SEABED REFLECTION COEFFICIENT

The reflection coefficient of the multiple-generating layer can be estimated directly, by comparing the amplitude of the primary reflection with its first multiple, after the latter has undergone Snell resampling (normalized to account for the amplitude increase due to compression of the offset axis) and differential geometric spreading correction. The following procedure is utilized to obtain a smooth estimate the reflection coefficient as a function of midpoint.

- Pick zero-offset traveltime to reflector for all midpoints: \( \tau^*(y) \).
- Loop over midpoints \((y')\):
  - Apply NMO to data. Extract small time window around \( \tau^*(y') \). Output is \( p_0(t, x, y') \).
  - Apply NMO for first-order multiple (Brown, 2003b), normalized Snell resampling, and differential geometrical spreading to data. Extract time small window around \( 2\tau^*(y') \). Output is \( m(t, x, y') \).
  - Optional: Align \( p_0(t, x, y') \) with \( m(t, x, y') \) on a trace-by-trace basis using (for example) the crosscorrelation technique of Rickett and Lumley (2001). Output is \( p(t, x, y') \).
  - Optional: Compute residual weight which reflects “quality” of the data at this midpoint. Output is \( w(y') \).
- Minimize the following quadratic functional for unknown seabed reflection coefficient vector \( r(y) \):

\[
\sum_{k=1}^{ny} \sum_{j=1}^{nx} \sum_{i=1}^{nt} w(k)^2 (r(k) p(i, j, k) - m(i, j, k))^2 + \epsilon^2 \sum_{k=2}^{ny} (r(k) - r(k-1))^2
\]  

Minimization of equation (6) is the equivalent to solving a regularized least-squares problem. To minimize the first term, we adjust \( r(y) \) to force the scaled primary to match the multiple. To minimize the second (regularization) term, we force \( r(y) \) to vary slowly across midpoint. The scalar term \( \epsilon \) balances data fitting with model smoothness.

Figure 3 shows the stack of the raw Mississippi Canyon 2-D test dataset, acquired and distributed by WesternGeco. The LSJIMP method is tested on this data by Brown (2003a). In addition to the seabed peglegs, peglegs from the top of salt and strong reflector R1 are included in the inversion. Rather than including each of these surfaces separately, the R1 and top salt events are assumed to arise from a single reflector; from midpoint 0 m to roughly 6000 m, it is R1, while from 6000 m to 20000 m it is the top of salt.

Figure 4 illustrates the reflection coefficient estimation procedure applied to 750 midpoints of the Mississippi Canyon data. The geology is quite complex in some areas, and the data looks decidedly incoherent. For this reason, the residual weight \( w(y) \) is quite important. I eyeballed Figure 4 and heuristically picked the residual weights shown therein. The seabed
reflection coefficient is nearly constant (-0.12) across the entire spread. The rugosity of the top of salt severely limited the sections of data considered “good”, but since salt is generally somewhat homogeneous in material properties, we can confidently assume some degree of spatial similarity.

Figure 3: Stacked Mississippi Canyon 2-D dataset (750 midpoints), annotated with important horizons and multiples. Labeled events: R1 - strong reflection; TSR - top of salt; BSR - bottom of salt; WBM - first seabed multiple; R1PL - seabed pegleg of R1; R1PM - R1 pure surface multiple; TSPL - seabed pegleg of TSR; BSPL - seabed pegleg of BSR; TSPM - TSR pure surface multiple.

APPLYING THE SEABED REFLECTION COEFFICIENT IN PRACTICE

Figure 5 illustrates that a first-order pegleg multiple consists of two unique arrivals, each with the same traveltime in a $v(z)$ medium. If the material properties of the water bottom and target reflector do not vary across midpoint, then the two arrivals also have the same strength. In this
Figure 4: a) Stack of time window around seabed reflection. b) Stack of time window around first seabed multiple. c) Seabed residual weight (either 0 or 1). d) Estimated seabed reflection coefficient. e) Stack of time window around top of salt reflection. f) Stack of time window around first top of salt multiple. g) Top of salt residual weight (either 0 or 1). h) Estimated top of salt reflection coefficient.

In the idealized case, the expression for total pegleg multiple amplitude derived by Backus (1959),

\[ r_{tot} = r_p \cdot (j + 1)r^j, \]  

holds, where \( j \) is the order of the multiple, \( r \) is the seabed reflection coefficient, and \( r_p \) is the target reflector reflection coefficient. In real data, sedimentary bedding layers always exhibit at least some “texture”, or variation of reflection strength with midpoint (Claerbout, 1985). If \( r(y) \) and \( r_p(y) \) are space-variable reflection coefficients of the seabed and subsea reflector, respectively, then the total strength of the first-order pegleg is

\[ r_{tot} = r(y_{m,1}) \cdot r_p(y_{p,1}) + r(y_{m,2}) \cdot r_p(y_{p,2}). \]  

In practice, a tractable compromise between the simplistic Backus model of pegleg amplitude [equation (7)] and the complicated model of equation (8) is justified. Let us assume that the
total reflection strength of a first-order pegleg can be modeled as

\[ r_{tot} = \left[ r(y_{m,1}) + r(y_{m,2}) \right] r_p(y). \]  

(9)

In other words, we ignore variations in reflection strength of the target reflector, but not the seabed. Notice from Figure 5 that the two pegleg “splits” impinge upon the seafloor at \( y_{p,1} \) and \( y_{p,2} \), and that the primary reflection occurs at \( y \). Therefore, if \( r_p \) varies linearly in the neighborhood of \( y \), then the average of \( r_p(y_{p,1}) \) and \( r_p(y_{p,2}) \) is \( r_p(y) \), and we can safely ignore the variation in reflection strength of the target. Equation (3) shows that \( \Delta y_p \) is always less or equal to than \( |y_{m,1} - y_{m,2}| \), making local linearity in reflection strength more likely.

Figure 5: In a 1-D earth, both pegleg multiple events shown here arrive at the same time. At fixed offset, the multiple legs of the two events impinge on the multiple-generating layer at \( y_{m,1} \) and \( y_{m,2} \), and on the target reflector at \( y_{p,1} \) and \( y_{p,2} \). \( \Delta y_p \), which we ignore in equation (9), goes asymptotically to zero as \( \tau \) increases.

EXTENSION TO 3-D

All operators derived in this paper extend fairly readily to 3-D. In equation (1), \( x_p \) becomes a vector quantity (\( x_p = [x_{p,1}, x_{p,2}]^T \)), though the derivation is the same. In equations (4) and (5) for the differential geometric spreading correction, \( x \) becomes \( \sqrt{x_1^2 + x_2^2} \), and \( \tau^* \) becomes a function of two midpoint axes. The reflection coefficient estimation scheme extends easily to 3-D, provided that HEMNO correctly aligns the primary and multiple.

CONCLUSIONS

In this short note I derived three linear operators that transform a pegleg multiple into an event that is directly comparable to its primary. Snell Resampling normalizes multiples to primaries with respect to AVO response and attenuation. Differential geometric spreading accounts for the longer raypaths of multiples through the earth. Finally, I introduce a flow for computing and applying the reflection coefficient of the multiple-generating layer. These operators extend naturally to three dimensions.
ACKNOWLEDGMENT

WesternGeco acquired and released the Mississippi Canyon dataset in 1997.

APPENDIX A

Derivation of Snell Resampling Operator

In the following appendix, I derive the Snell resampling operation, equation (1). The graphical basis for the derivation is Figure 1. Since the pegleg multiple and primary in the figure have the same emergence angle, $\theta$, the stepout of the two events is the same at $x$ and $x_p$. First we compute the stepout of the primary event (standard NMO equation):

$$
\begin{align*}
    t_p^2 &= \tau + \frac{x_p^2}{V^2} \quad \text{(A-1)} \\
    \frac{d}{dx_p}(t_p^2) &= 2t_p \frac{dt_p}{dx_p} = \frac{2x_p}{V^2} \quad \text{(A-2)} \\
    \frac{dt_p}{dx_p} &= \frac{x_p}{t_p V^2}. \quad \text{(A-3)}
\end{align*}
$$

Brown (2002) derived an extension to the conventional NMO equation which flattens peglegs to the zero-offset traveltime of the reflector of interest.

$$
    t_m = \sqrt{(\tau + j \tau^*)^2 + \frac{x^2}{V_{eff}^2}} \quad \text{(A-4)}
$$

Using equations (A-4) and (2), we can similarly compute the stepout of the corresponding $j^{th}$-order pegleg multiple:

$$
    \frac{dt_m}{dx} = \frac{x}{t_m V_{eff}^2}. \quad \text{(A-5)}
$$

Finally, we compute $x_p$ as a function of $x$. We square equations (A-3) and (A-5), set them equal, then substitute equation (A-4) for $t_m$ and $t_p$.

$$
\begin{align*}
    \frac{x_p^2}{t_p^2 V^4} &= \frac{x^2}{t_m^2 V_{eff}^4} \quad \text{(A-6)} \\
    x^2[V^4 \tau^2 + x_p^2 V^2] &= x_p^2[V_{eff}^4(\tau + j \tau^*)^2 + x^2 V_{eff}^2] \quad \text{(A-7)} \\
    x_p^2 &= \frac{x^2 \tau^2 V^4}{(\tau + j \tau^*)^2 V_{eff}^4 + x^2(V_{eff}^2 - V^2)} \quad \text{(A-8)}
\end{align*}
$$
REFERENCES


Narrow-azimuth migration of marine streamer data

Biondo Biondi

ABSTRACT
I introduce a new migration method that overcomes the limitations of common-azimuth migration while retaining its computational efficiency for imaging marine streamer data. The method is based on source-receiver downward-continuation of prestack data with a narrow range of cross-line offsets. To minimize the width of the cross-line offset range, while assuring that all the recorded events are correctly propagated, I define an “optimal” range of cross-line offset dips. To remove the effects of the boundary artifacts I apply a coplanarity condition on the prestack image. This process removes from the image cube the events that are not correctly focused at zero offset. Tests of the proposed method with the SEG-EAGE salt dataset show substantial image improvements in particularly difficult areas of the model and thus confirm the capability of the new method to overcome the limitations of common-azimuth migration in complex areas.

INTRODUCTION
Common-azimuth (Biondi and Palacharla, 1996) is an attractive alternative to shot-profile migration for wave-equation 3-D prestack migration. For 3-D marine streamer data, it is computationally more efficient than shot-profile migration and thus it has been implemented in different migration algorithms (Jin et al., 2002) and applied to several datasets (Fliedner et al., 2002; Le Rousseau et al., 2002). In addition to the computational efficiency, common-azimuth migration has the substantial advantage of enabling migration velocity analysis (Clapp and Biondi, 2000; Liu et al., 2001) by generating high-quality Angle-Domain Common Image Gathers (ADCIG) (Prucha et al., 1999) without additional computations. However, in the presence of arbitrary velocity functions, common-azimuth migration is not exact. In this paper, I propose a method for generalizing common-azimuth migration that is accurate in presence of arbitrary velocity variations but retains computational advantages with respect to shot-profile migration.

Common-azimuth migration is based on the principles of source-receiver (survey-sinking) migration (Claerbout, 1985). Source-receiver migration is theoretically equivalent to shot-profile migration based on downward continuation (Wapenaar and Berkhout, 1987; Biondi, 2002; Shan and Zhang, 2003). (notice, not shot-profile migration based on time extrapolation), and thus the proposed generalization of common-azimuth migration has the potential to

1email: biondo@sep.stanford.edu
produce as high-quality images as the more computationally demanding shot-profile migration.

At the basis of common-azimuth computational efficiency is the exploitation of the narrow azimuthal range of typical marine data acquired by towed streamers. This is made possible by a crucial characteristic of source-receiver migration: during source-receiver downward continuation the offset range shrinks with depth. In most practical situations the offset range shrinks monotonically with depth, but this property is not guaranteed in arbitrarily heterogeneous media. At the limit, the cross-line offset can be assumed to be zero and all recorded events can be propagated with the same azimuth (e.g. common azimuth) at every depth level. The assumption of no cross-line offset provides the computational efficiency of common-azimuth migration, but also causes its accuracy limitations. In this paper I remove this assumption by downward continuing the data on a narrow, but finite, cross-line offset range. To achieve computational efficiency, the cross-line offset range must be as narrow as possible and still “capture” all the useful propagation paths and avoid boundary artifacts. I propose to accomplish this goal by applying two complementary procedures: 1) definition of an “optimal” range of cross-line offset dips for the downward continuation, and 2) application of a coplanarity condition on the prestack image that enhances the events that are well focused at the imaging point (zero offset). I presented a method to perform 1) in (Biondi, 2001). In this paper I introduce a method to perform 2).

NARROW-AZIMUTH MIGRATION

In (Biondi, 2001) I describe a method for reducing the cost of full prestack downward continuation in the midpoint-wavenumber domain by defining an “optimal” range of cross-line offset dips. The method exploits the information provided by the common-azimuth equation to define a range of cross-line offset wavenumbers. To illustrate the need for applying the coplanarity condition on the prestack image for enhancing the events that are well focused at the imaging point (zero offset), I will now review some of the results presented in my previous report. In that report I analyzed in detail the kinematics of the migration results of a 3-D synthetic data set (Vaillant and Biondi, 2000). The reflectivity field consists of a set of five dipping planes, from zero dip to 60 degrees dip. The azimuth of the planes is 45 degrees with respect to the direction of the acquisition. This reflection geometry (i.e. dipping reflectors oriented at 45 degrees with respect to the acquisition direction) is known to be the most challenging for common-azimuth migration. The velocity was \( V(z) = 1.5 + 0.5 z \) km/s, which corresponds to the upper limit among the typical gradients found in the Gulf of Mexico. The maximum source-receiver offset was 3 km. Figure 1 shows the geometry of the reflectors.

Figure 2 shows a subset of the image cube obtained by common-azimuth migration. The front face of the cube displayed in the figure is an in-line section through the stack. The other two faces are sections through the prestack image as a function of the offset ray parameter \( p_{xh} \). The three events in the ADCIG (right panel) correspond to the planes dipping at 30, 45 and 60 degrees. Notice that the events are almost perfectly flat as a function of the offset ray parameter \( p_{xh} \), except for the reflections from the 60-degree plane with large offset ray
Figure 1: Geometry of the set of slanted planes, dipping at 0°, 15°, 30°, 45° and 60° toward increasing x and y, at 45° with respect to the in-line direction.

parameters (i.e. large reflection angle). This slight smiling in the ADCIG is caused by the common-azimuth approximation.

The small error visible in the common-azimuth migration can be completely corrected by using full source-receiver migration. Figure 3 shows an ADCIG extracted at the same location as the ADCIG shown in Figure 2, but from the migrated image obtained by a full source-receiver migration. For these data, 16 cross-line offsets were necessary to obtain an accurate image when using full source-receiver migration. In contrast, only 4 cross-line offsets are necessary to obtain an accurate image when using the narrow-azimuth downward continuation described in (Biondi, 2001). Figure 4 shows an ADCIG extracted at the same location as the ADCIG shown in Figure 2, but from the migrated image obtained by narrow-azimuth migration with $N_{y_b} = 4$.

Coplanarity condition

The previous results show that as few as four cross-line offsets might be sufficient to “capture” all the useful propagation paths, when the range of cross-line offset dips for the downward continuation is properly defined. However, boundary artifacts caused by either reflecting or circular boundary conditions are unavoidable when such a narrow range of offsets are used.
Figure 3: ADCIG extracted at the same location as the ADCIG shown in Figure 2, but from the migrated image obtained by a full source-receiver migration with $N_{y_h} = 16$.

Figure 4: ADCIG extracted at the same location as the ADCIG shown in Figure 2, but from the migrated image obtained by narrow-azimuth migration with $N_{y_h} = 4$.

To propagate the data. The effects of boundary artifacts are more obvious in the full source-receiver migration results (Figure 3) than in the narrow-azimuth migration results (Figure 4), but, as I discuss in (Biondi, 2001), they might also become a problem when narrow-azimuth downward-continuation is used with constant sampling in the cross-line offset wavenumber ($dk_{y_h}$). A potential solution to the problem caused by boundary effects can be the use of absorbing boundary conditions. However, effective absorbing boundary conditions require the addition of several grid points, and consequently can cause a substantial increase in the computational cost.

Fortunately, the boundary artifacts can be effectively attenuated by applying a post-processing filter on the prestack image that preserves only the events for which the source and receiver rays are coplanar at the imaging point. This condition must be fulfilled by all the events that are correctly focused at zero offset because two lines passing through the same point are coplanar. The coplanarity condition can be easily applied on the prestack image after transformation into the Fourier domain, possibly at the same time that ADCIGs are computed using a 3-D generalization of the method described by Sava and Fomel (2003), as presented in Tisserant and Biondi (2003).

The coplanarity condition can be derived by simple geometric considerations starting from
the common-azimuth condition expressed as (Biondi and Palacharla, 1996):

\[
\hat{k}_{yh} = k_{ym} \frac{\sqrt{\frac{\omega^2}{v^2(g,z)} - \frac{1}{4} (k_{xm} + k_{yh})^2} - \sqrt{\frac{\omega^2}{v^2(s,z)} - \frac{1}{4} (k_{xm} - k_{yh})^2}}{\sqrt{\frac{\omega^2}{v^2(g,z)} - \frac{1}{4} (k_{xm} + k_{yh})^2} + \sqrt{\frac{\omega^2}{v^2(s,z)} - \frac{1}{4} (k_{xm} - k_{yh})^2}}.
\]

(1)

where \( \omega \) is the temporal frequency, \( k_{xm} \) and \( k_{ym} \) are the midpoint wavenumbers, \( k_{xh} \) is the in-line offset wavenumbers.

As for the common-azimuth condition, the coplanarity condition can be expressed as a relationship that links the cross-line offset wavenumber \( k_{yh} \) to the other wavenumbers in the image. For events with azimuth aligned along the in-line direction (\( x_m \) in my notation), the expression of the coplanarity condition is:

\[
k_{yh} = \frac{k_{ym} k_{xm} k_{xh}}{k_z^2 + k_{ym}^2},
\]

(2)

where \( k_z \) is the vertical wavenumber.

The condition expressed in equation (2) can be easily generalized to be valid for an arbitrary azimuthal direction \( \beta \) (Tisserant and Biondi, 2003). The wavenumber axes are rotated by \( \beta \) in both the midpoint wavenumber plane \( (k_{xm}, k_{ym}) \) and the offset wavenumber plane \( (k_{xh}, k_{yh}) \).

As discussed in (Tisserant and Biondi, 2003), in a prestack image each event fulfills the coplanarity condition for one value of the azimuth. However, streamer data illuminate the reflectors only within a fairly narrow azimuthal range. Therefore, the coplanarity condition can be used to remove from the prestack image all the events that are imaged outside a given azimuthal range. For example, if we image only events within a range of \( \pm 22.5 \) degrees, we will remove the 75 % of the events in the image that are least likely to be real reflection events.

**Five-planes synthetic data set migration results**

We can gain a intuitive understanding of the effects of applying the coplanarity condition, and of the trade-off when setting the parameters for narrow-azimuth migration, by analyzing the results of two full source-receiver migrations of the synthetic data set with five dipping planes, which we introduced in the previous section. I used 8 cross-line offsets for both migrations; for the first migration the cross-line sampling was 100 meters whereas for the second one the cross-line sampling was 50 meters. There is a trade-off between the coarser and finer cross-line offset sampling. With the finer sampling we expect stronger artifacts caused by the circular boundary condition because the offset range is narrower (only 400 meters vs. 800 meters). On the other hand, with the finer offset sampling the cross-line dip range is wider than with the coarse offset sampling and thus we expect better imaging of the events reflected with wide-aperture angles from the steeply dipping planes.

These “theoretical” predictions are confirmed by the zero offset images (panels a) and the ADCIGs (panels b) displayed in Figure 5 (\( \Delta y_h = 100 \) meters) and Figure 6 (\( \Delta y_h = 50 \)
Figure 5: (a) Stack (i.e. image at zero offset) and (b) CIG produced by full prestack migration with $N_{y_h} = 8$ and $\Delta y_h = 100$ meters.

The ADCIG shown in Figure 6b shows stronger artifacts than the ADCIG shown in Figure 5b. Even the “stacked” image (i.e. zero-offset image) shown in Figure 6a has strong artifacts, at least in the shallow part of the section. On the other hand, the finer offset sampling allows a slight better imaging of the wide-aperture reflection from the 60-degree plane, as the comparison of the deepest event in Figure 5b and Figure 6b demonstrates.

Figure 7 shows the same section and ADCIG as in Figure 5 but after applying the coplanarity condition for zero azimuth (i.e. $\beta = 0$). As expected, the events from the flattish reflectors are preserved since their azimuth at the reflection point is close to zero. In contrast, the reflections from the steeper reflectors are attenuated because their azimuth at the reflection point is larger than zero.

An interesting side-benefit of the capability of selecting reflections with a given azimuthal direction from full prestack migration, is the possibility to demonstrate the differences between the zero azimuth image shown in Figure 7, and the result obtained by common-azimuth migration shown in Figure 8. In constant velocity, the image produced by common-azimuth migration is equivalent to the zero-azimuth image. However, in variable velocity the two images are substantially different. The zero azimuth image (Figure 7) contains only the events that were close to zero azimuth at the reflection point. In contrast, the common-azimuth mi-
Narrow-azimuth migration

Figure 7: (a) Stack (i.e. image at zero offset) and (b) CIG for reflections with 0 degrees azimuth at the image point. This image cube was produced starting from the image cube obtained by full prestack migration with $N_y = 8$ and $\Delta y = 100$ meters.

Common-azimuth downward continuation propagates all the events assuming that they are coplanar along the zero azimuth. In variable velocity this assumption is incorrect for some of the events, which are therefore slightly mispositioned in the image. As the common-azimuth migration image illustrates, the challenge of this data set is to image properly the wide-aperture reflections from the 60-degree plane. Simple ray-tracing modeling indicates that those reflections occur along an azimuth oriented approximately at 18 degrees with respect to the acquisition geometry. Figure 9 shows the image obtained by selecting the reflections with 18-degrees azimuth from the results of full prestack migration with the coarser offset sampling ($\Delta y = 100$ meters). Figure 10 shows the image obtained by selecting the reflections with 18-degrees azimuth from the results of full prestack migration with the finer offset sampling ($\Delta y = 50$ meters). Both images show significantly weaker artifacts than the corresponding images with full azimuth (Figure 5 and Figure 6). Figure 9 has weaker artifacts than Figure 10. The reflections for the 60-degree plane is flat as a function of the reflection angle for both ADCIGs (panels b), but Figure 10b has broader angular bandwidth (up to 40 degrees for as compared with up to 35 degrees) than Figure 9b.

Figure 8: (a) Stack (i.e. image at zero offset) and (b) CIG produced by common-azimuth migration.
Figure 9: (a) Stack (i.e. image at zero offset) and (b) CIG for reflections with 18 degrees azimuth at the image point. This image cube was produced starting from the image cube obtained by full prestack migration with $N_{y_h} = 8$ and $\Delta y_h = 100$ meters.

The best-quality image can be obtained by stacking the images corresponding to a range of azimuths. This range can be fairly narrow because of the narrow-azimuth nature of streamer data. In this example, I stacked the image corresponding to azimuths within the 0–30 degrees range. Figure 11 and Figure 12 are the result of this averaging process. Notice the further attenuation of the artifacts as compared with both the full-azimuth images (Figure 5 and Figure 6) and the 18-degrees azimuth images (Figure 9 and Figure 10). As before, there is a trade-off between the better signal-to-noise in Figure 11, and the wider angular bandwidth in Figure 12.

The last two figures show that the stacking over azimuth decreases the amplitude of the reflections with wide reflection angles relatively to the narrow reflection angles. The intuitive explanation of this phenomenon is that the narrow reflection angles are enhanced by the stacking over azimuth because they are more stationary as a function of azimuth than the wide reflection angles. I believe that this effect can be compensated by applying an appropriate...
Figure 11: (a) Stack (i.e. image at zero offset) and (b) CIG for reflections with azimuth within the 0–30 degrees range. This image cube was produced starting from the image cube obtained by full prestack migration with $N_{y_h} = 8$ and $\Delta y_h = 100$ meters.

jacobian during the integration over azimuth, but I have not derived such a factor yet.

**SEG-EAGE salt data set migration results**

The improvement in image quality achieved by applying source-receiver migration on a narrow range of cross-line offsets in conjunction with the coplanarity condition is demonstrated in the following results obtained from the SEG-EAGE salt data set. Figure 13a shows the in-line section of the velocity model taken at cross-line location of 5,770 meters. Figure 13b shows the corresponding migrated image obtained by common-azimuth migration. The section is well imaged everywhere, with the exception of the bottom of the salt around in-line location of 4,000 meters. This inaccuracy in the image is likely to be caused by the common-azimuth approximation.

Figure 14 compares the results of common-azimuth migration and full source-receiver
migration with 8 cross-line offsets and the application of the coplanarity condition. It shows three zooms into the area of interest. Figure 14a shows the velocity model. Figure 14b shows the image obtained by common-azimuth migration. Figure 14c shows the result of stacking the images obtained by applying the coplanarity condition on the azimuthal range defined by $|\beta| \leq 16$ degrees. Notice the improved definition of the bottom of the salt in Figure 14c compared to Figure 14b.

Similar improvements are visible in the corresponding depth slices. Figure 15 compares the slices taken at a depth of 2,600 meters. Figure 15a shows the velocity model. Figure 15b shows the image obtained by common-azimuth migration. Figure 15c shows the result of stacking the images obtained by applying the coplanarity condition on the azimuthal range defined by $|\beta| \leq 16$ degrees. Now the salt bottom boundary located between in-line locations of 4,000 and 4,500 meters is well defined. Notice that the portion of the salt boundary that is not well delineated by the image (between in-line locations of 3,000 and 3,500 meters) is not properly illuminated by the data.

**CONCLUSIONS**

I presented a “narrow-azimuth” generalization of common-azimuth migration that overcomes the accuracy limitations and retains the computational efficiency of the original method. The new method is based on: 1) the definition of an “optimal” range of cross-line offset dips for the downward continuation, and 2) application of a “coplanarity” condition on the prestack image for enhancing the correctly focused events. The migration examples show that the new method has the potential of correcting the inaccuracy introduced by common-azimuth migration even in challenging situations such as the one presented by the SEG-EAGE salt data set.

**ACKNOWLEDGMENTS**

I would like to thank Bob Clapp and Paul Sava for developing the WEI library that allowed me to run wave-equation prestack migration using MPI. Without WEI I could not have tested narrow-azimuth migration on the SEG-EAGE data set.

**REFERENCES**

Biondi, B., and Palacharla, G., 1996, 3-D prestack migration of common-azimuth data: Geophysics, 61, no. 6, 1822–1832.


Figure 13: In-line sections ($y_m = 5770$ m): (a) the velocity model, (b) common-azimuth migration.

Figure 14: Zooms of the in-line sections ($y_m = 5770$ m): (a) the velocity model (b) common-azimuth migration, (c) full source-receiver migration with 8 cross-line offsets and the application of the coplanarity condition.
Figure 15: Zooms of the depth slices ($z = 2,600$ m): (a) the velocity model (b) common-azimuth migration, (c) full source-receiver migration with 8 cross-line offsets and the application of the coplanarity condition.


Equivalence between shot-profile and source-receiver migration

Guojian Shan and Guanquan Zhang

ABSTRACT

Shot-profile migration and source-receiver migration seem different, but the image and Common Image Gather they obtain is the same. In this paper, we prove that shot-profile migration and source-receiver migration are equivalent, assuming that the imaging condition is cross-correlation and the method for propagating the source and receiver wavefields is a one-way wave equation. This is achieved after generalizing source-receiver migration to arbitrary sources.

INTRODUCTION

At first glance, shot-profile migration and source-receiver migration seem to be completely different. They are performed on different data geometry and use different equations to extrapolate wavefields. Shot-profile migration downward continues source and receiver wavefields independently, and produces an image through a cross-correlation between these two wavefields along the time axis. Source-receiver migration extrapolates the CMP gathers with the Double Square Root equation, and creates an image by extracting the wavefield at zero time and zero subsurface offset.

However, shot-profile migration and source-receiver migration obtain the same migration result. Wapenaar and Berkhout (1987) proves that identical stacked images will be obtained from these two methods. Biondi (2002) proves that an equivalent image cube will be obtained, given the assumptions that the source is an impulse function, the imaging condition is cross-correlation, and the source and receiver wavefields are downward propagated by a one-way wave equation. The cross-correlation imaging condition and one-way wave equation downward continuation are the key points for the equivalence between shot-profile migration and source-receiver migration. In this paper, we give a new proof for the equivalence between shot-profile migration and source-receiver migration for an arbitrary source.

Before demonstrating the equivalence between them, we first present an overview of shot-profile migration and source-receiver migration.

1email: shan@sep.stanford.edu, zgq@lsec.cc.ac.cn
SHOT-PROFILE MIGRATION

In shot-profile migration, each shot is treated as an independent physics experiment. Each shot is migrated separately and the images of all the shots are then stacked to generate the final image. The source for shot-profile migration is not necessarily an impulse function. Actually, it is a wavelet for a point shot in practice. The source can also be a plane wave, a primary reflection (Guitton, 2002), or some other configurations. We assume that the source wavefield is a down-going wavefield and the receiver wavefield is an up-going wavefield. These two wavefields are downward continued independently. Let \( U(x_U, z = 0, \omega, s) \) be the receiver wavefield and \( D(x_D, z = 0, \omega, s) \) be the source wavefield at the surface for shot \( s \). The source wavefield at the subsurface \( U(x_U, z, \omega, s) \) can be obtained by extrapolating \( U(x_U, z = 0, \omega, s) \) with the up-going wave equation

\[
\frac{\partial}{\partial z} U(x_U, z, \omega, s) = \frac{i \omega}{v(x_U, z)} \sqrt{1 + \frac{v^2(x_U, z)}{\omega^2}} \frac{\partial^2}{\partial x_U^2} U(x_U, z, \omega, s),
\]

and the source wavefield at subsurface \( D(x_D, z, \omega, s) \) can be obtained by extrapolating \( D(x_D, z = 0, \omega, s) \) with the down-going wave equation

\[
\frac{\partial}{\partial z} D(x_D, z, \omega, s) = -\frac{i \omega}{v(x_D, z)} \sqrt{1 + \frac{v^2(x_D, z)}{\omega^2}} \frac{\partial^2}{\partial x_D^2} D(x_D, z, \omega, s).
\]

where \( \sqrt{1 + \frac{v^2(x_D, z)}{\omega^2}} \frac{\partial^2}{\partial x_D^2} \) is a pseudo-partial differential operator (Zhang, 1993). The image for shot \( s \) is formed by cross-correlating the source wavefield \( D(x_D, z, \omega, s) \) and receiver wavefield \( U(x_U, z, \omega, s) \) along the time axis at all depths and evaluating this at zero time lag (Claerbout, 1971). Stacking the images of all the shots, we can get the image of frequency \( \omega \)

\[
I(x, z, \omega) = \sum_s U(x_U = x, z, \omega, s) \bar{D}(x_D = x, z, \omega, s),
\]

and the image of all frequencies

\[
I(x, z) = \sum_\omega I(x, z, \omega).
\]

To perform a velocity analysis, Rickett and Sava (2002) developed the Common Image Gather (CIG) for shot-profile migration, calculated by

\[
I(x, h, z) = \sum_\omega I(x, h, z, \omega) = \sum_\omega \sum_s U(x_U = x + h, z, \omega, s) \bar{D}(x_D = x - h, z, \omega, s).
\]

SOURCE-RECEIVER MIGRATION

Traditional source-receiver migration is based on the concept of survey sinking (Claerbout, 1985). It sorts the recorded data into CMP gathers \( P(x, h, z = 0, \omega) \) and propagates the CMP
gather to the subsurface $P(x,h,z,\omega)$ with the Double Square Root (DSR) equation

$$
\frac{\partial}{\partial z} P(x,h,z,\omega) = \left( \frac{i\omega}{v(x_r,z)} \sqrt{1 + \frac{v^2(x_s,z)}{\omega^2} \frac{\partial^2}{\partial x_s^2}} + \frac{i\omega}{v(x_r,z)} \sqrt{1 + \frac{v^2(x_r,z)}{\omega^2} \frac{\partial^2}{\partial x_r^2}} \right) P(x,h,z,\omega),
$$

(6)

where $x_s = x - h$ is the shot location, $x_r = x + h$ is the receiver location. The wavefield $P(x,h,z,\omega)$ at each depth $z$ is equivalent to the data that would have been recorded when the shots and receivers were located at that depth. Source-receiver migration produces an image by extracting the wavefield at zero subsurface offset $P(x,h = 0,z,\omega)$, and stacking over all frequencies. Correspondingly, the stack of $P(x,h,z,\omega)$ along the frequencies is its CIG.

Traditional source-receiver migration assumes that the source is an impulse function at the source location $x_s$. But in fact, source-receiver migration works for arbitrary sources, such as wavelets, plane waves, and primary reflections as well. For arbitrary sources, the surface wavefield $P(x,h,z = 0,\omega)$ is not a simple CMP gather of the recorded data, but the stack of the cross-correlation between the source wavefield and the receiver wavefield at the surface

$$
P(x,h,z = 0,\omega) = \sum_s U(x_U = x + h,z = 0,\omega,s) \bar{D}(x_D = x - h,z = 0,\omega,s).
$$

(7)

Then $P(x,h,z = 0,\omega)$ is downward continued to the subsurface with the DSR equation (6) and the image is formed by the same method as the traditional source-receiver migration. In fact, traditional source-receiver migration is a special case of source-receiver migration when the source is an impulse function at the source location, and the CMP gather of the recorded data at the surface is the cross-correlation between the impulse function source and the receiver wavefield, which is the recorded data for each shot.

**DEMONSTRATION OF EQUIVALENCE**

Although shot-profile migration and source-receiver migration look totally different, they obtain both the same image and CIG. In this section, we prove that the mono-frequency image $I(x,z,\omega)$ and CIG $I(x,h,z,\omega)$ in the shot-profile migration are exactly the same mono-frequency image $P(x,h = 0,z,\omega)$ and CIG $P(x,h,z,\omega)$ in the source-receiver migration, respectively.

We define a new wavefield $Q_s(x_U,x_D,z,\omega)$, which is the cross-correlation between the source wavefield $D(x_D,z,\omega,s)$ and the receiver wavefield $U(x_U,z,\omega,s)$ in the shot-profile migration for shot $s$, that is

$$
Q_s(x_U,x_D,z,\omega) = U(x_U,z,\omega,s) \bar{D}(x_D,z,\omega,s),
$$

(8)

and the wavefield $Q(x_U,x_D,z,\omega)$ is the stack of $Q_s(x_U,x_D,z,\omega)$ along all the shots,

$$
Q(x_U,x_D,z,\omega) = \sum_s Q_s(x_U,x_D,z,\omega).
$$

(9)
Obviously, from equation (7), \( Q_s(x_U, x_D, z = 0, \omega) \) is the surface data in source-receiver migration. We will demonstrate that the wavefield \( Q_s(x_U, x_D, z, \omega) \) satisfies the DSR equation,

\[
\frac{\partial}{\partial z} Q_s = \left( \frac{i \omega}{v(x_U, z)} \sqrt{1 + \frac{v^2(x_U, z)}{\omega^2} \frac{\partial^2}{\partial x_U^2}} + \frac{i \omega}{v(x_D, z)} \sqrt{1 + \frac{v^2(x_D, z)}{\omega^2} \frac{\partial^2}{\partial x_D^2}} \right) Q_s,
\]

where \( Q_s = Q_s(x_U, x_D, z, \omega) \). By extension, \( Q(x_U, x_D, z, \omega) \) also satisfies the DSR equation. Thus shot-profile migration and source-receiver migration are two different ways to obtain wavefield \( Q \) at the subsurface. In shot-profile migration, source and receiver wavefields are downward continued into the subsurface with the one-way wave equation, and the wavefield \( Q(x_U, x_D, z, \omega) \) is formed by cross-correlating the source wavefields and receiver wavefields and stacking over all shots at all depths. But in source-receiver migration, the wavefield \( Q(x_U, x_D, z, \omega) \) at the surface \( Q(x_U, x_D, z = 0, \omega) \) is obtained by cross-correlating the source wavefield and the receiver wavefield at the surface, and \( Q(x_U, x_D, z, \omega) \) is formed by extrapolating \( Q(x_U, x_D, z = 0, \omega) \) to all depths with the DSR equation.

From the Leibniz rule, we have

\[
\frac{\partial Q_s}{\partial z} = \frac{\partial U}{\partial z} \cdot \bar{D} + U \frac{\partial \bar{D}}{\partial z},
\]

where \( U = U(x_U, z, \omega, s) \) and \( D = D(x_D, z, \omega, s) \). Since \( \bar{U}(x_U, z, \omega, s) \) is an up-going wavefield, it satisfies the up-going wave equation (1), so we have

\[
\frac{\partial U}{\partial z} \cdot \bar{D} = \left( \frac{i \omega}{v(x_U, z)} \sqrt{1 + \frac{v^2(x_U, z)}{\omega^2} \frac{\partial^2}{\partial x_U^2}} U \right) \cdot \bar{D}.
\]

\( \bar{D}(x_D, z, \omega, x) \) is not dependent on \( x_U \), so it is constant with respect to the operator \( \sqrt{1 + \frac{v^2(x_U, z)}{\omega^2} \frac{\partial^2}{\partial x_U^2}} \), and we have

\[
\left( \sqrt{1 + \frac{v^2(x_U, z)}{\omega^2} \frac{\partial^2}{\partial x_U^2}} U \right) \cdot \bar{D} = \sqrt{1 + \frac{v^2(x_U, z)}{\omega^2} \frac{\partial^2}{\partial x_U^2}} (U \bar{D}) = \sqrt{1 + \frac{v^2(x_U, z)}{\omega^2} \frac{\partial^2}{\partial x_U^2}} Q_s (13)
\]

Summarizing equation (12) and (13), we have

\[
\frac{\partial U}{\partial z} \cdot \bar{D} = \frac{i \omega}{v(x_U, z)} \sqrt{1 + \frac{v^2(x_U, z)}{\omega^2} \frac{\partial^2}{\partial x_U^2}} Q_s.
\]

It is easy to prove that

\[
\frac{\partial \bar{D}}{\partial z} = \bar{D} \frac{\partial \bar{D}}{\partial z}.
\]

So the second term of equation (11) changes to

\[
U \frac{\partial \bar{D}}{\partial z} = U \bar{D} \frac{\partial \bar{D}}{\partial z}.
\]
Since $D(x_D, z, \omega, s)$ is a down-going wavefield, it satisfies the down-going wave equation (2), and we have

$$U \cdot \frac{\partial D}{\partial z} = U \cdot \left( \frac{-i \omega}{v(x_D, z)} \sqrt{1 + \frac{v^2(x_D, z)}{\omega^2} \frac{\partial^2}{\partial x_D^2} D} \right)$$

$$= U \left( \frac{i \omega}{v(x_D, z)} \sqrt{1 + \frac{v^2(x_D, z)}{\omega^2} \frac{\partial^2}{\partial x_D^2} (U \bar{D})} \right).$$

Again, $U(x_U, z, \omega, s)$ does not depend on $x_D$, so we have

$$U \left( \frac{i \omega}{v(x_D, z)} \sqrt{1 + \frac{v^2(x_D, z)}{\omega^2} \frac{\partial^2}{\partial x_D^2} D} \right) = \frac{i \omega}{v(x_D, z)} \sqrt{1 + \frac{v^2(x_D, z)}{\omega^2} \frac{\partial^2}{\partial x_D^2} Q_s}.$$  

Summarizing equations (15-20), we have

$$U \frac{\partial \bar{D}}{\partial z} = \frac{i \omega}{v(x_D, z)} \sqrt{1 + \frac{v^2(x_D, z)}{\omega^2} \frac{\partial^2}{\partial x_D^2} Q_s}.$$  

Finally, from equation (11), equation (14) and equation (21), we know $Q_s$ satisfies the DSR equation (10). $Q_s$ is the stack of $Q_s$ over all shots, so by extension $Q_s$ satisfies the DSR equation also.

It is obvious that the image of shot-profile migration in equation (3) is

$$I(x, z, \omega) = Q(x_U = x, x_D = x, z, \omega),$$

and the corresponding CIG in equation (5) is

$$I(x, h, z, \omega) = Q(x_U = x + h, x_D = x - h, z, \omega).$$

In traditional source-receiver migration, $Q(x_U = x + h, x_D = x - h, z = 0, \omega)$ is the stack of the cross-correlation between the impulse source and the recorded data along shots, which is the CMP gather $P(x, h, z = 0, \omega)$ of the recorded data at the surface. Since both $Q(x_U = x + h, x_D = x - h, z, \omega)$ and $P(x, h, z, \omega)$ are obtained by propagating $Q(x_U = x + h, x_D = x - h, z = 0, \omega)$ to the subsurface with the DSR equation (10), they are equivalent. If the source in source-receiver migration is not an impulse function, $Q(x_U = x + h, x_D = x - h, z = 0, \omega)$ is the stack of the cross-correlation between the source wavefield and the receiver wavefield, and the same conclusion is reached. Thus we have

$$I(x, z, \omega) = P(x, h = 0, z, \omega) = Q(x_U = x, x_D = x, z, \omega),$$

and

$$I(x, h, z, \omega) = P(x, h, z, \omega) = Q(x_U = x + h, x_D = x - h, z, \omega).$$
CONCLUSION

In this paper, we generalize the conception of source-receiver migration. Generalized source-receiver migration works for arbitrary sources. It is not restricted to a point source and it is not survey-sinking, but rather the downward continuation of the cross-correlation between the source and the receiver wavefield. We prove the equivalence between shot-profile migration and source-receiver migration for arbitrary sources, given the assumption that the imaging condition is cross-correlation and the wavefields are propagated by one-way wave equation downward continuation.

REFERENCES


Claerbout, J. F., 1971, Toward a unified theory of reflector mapping: Geophysics, 36, no. 03, 467–481.


Multichannel deconvolution imaging condition for shot-profile migration

Alejandro A. Valenciano and Biondo Biondi

ABSTRACT
A significant improvement of seismic image resolution is obtained by framing the shot-profile migration imaging condition as a 2-D deconvolution in the shot position/time ($x_s, t$) domain. This imaging condition gives a better image resolution than the crosscorrelation imaging condition and is more stable than the “more conventional” 1-D deconvolution imaging condition. A resolution increment is also observed in common image gathers (CIGs) computed with the 2-D deconvolution imaging condition, thus allowing a more accurate velocity analysis.

INTRODUCTION
Shot-profile migration is a method used to construct an image of the earth’s interior from seismic data. This technique is implemented in two steps. The first step consists of constructing the source and the receiver wavefields for each shot position. This is done by downward propagation of an impulsive source and upward propagation of the data recorded at the surface. The second step consists of applying the imaging condition. The imaging step is based on Claerbout’s imaging principle (Claerbout, 1971).

A practical way to implement Claerbout’s imaging principle is to use match filters (cross-correlation of the shot and receiver wavefields). Thereby, for each shot position, a partial image is obtained by matching the source and the receiver wavefields along the time dimension. Then, the image is formed by stacking the partial images at each subsurface location.

We propose a different imaging condition that also satisfies Claerbout’s imaging principle. It consists of deconvolving the receiver wavefield by the source wavefield in the shot position/time ($x_s, t$) domain. This 2-D deconvolution imaging condition has the advantage of improving the image resolution while keeping the resulting image stable.

In this paper we show the advantages of the 2-D deconvolution over crosscorrelation and 1-D deconvolution imaging conditions. To do this, we use a synthetic model with five dipping reflectors. Using this model, we demonstrate the advantages of 2-D deconvolution not only for image resolution but also for estimating the correct moveout of reflectors in the angle-domain (Sava and Guitton, 2003).

1email: valencia@sep.stanford.edu, biondo@sep.stanford.edu
DATA AND WAVEFIELD DIMENSIONALITY

To implement a better imaging condition that is also feasible, it is important to understand 3-D prestack data and wavefield dimensionality. 3-D prestack seismic data are defined in a 5-D continuum \((t, x_s, y_s, x_g, y_g)\) (Biondi, 1998), where \(t\) is time, \(x_s, y_s\) and \(x_g, y_g\) are the surface coordinates of the sources and the receivers, respectively.

After applying the first step of shot-profile migration (source and receiver wavefield construction) the dimensionality of the data increases. Thus, for each shot position \((x_s, y_s)\) the source and the receiver wavefields \(u(x, y, z, t)\) and \(d(x, y, z, t)\) are defined in four dimensions (Figure 1), where \((x, y, z)\) are the image space dimensions.

![Figure 1: Data dimensionality. The source and receiver wavefields are 6-D datasets.](image)

In the following analysis, we explain how to combine the source and the receiver wavefields to obtain an image. For simplicity, we restrict our analysis to 2-D prestack data. In this case, the source and the receiver wavefields are 4-D datasets defined in \((x, z, t)\) for each shot position \(x_s\).

1-D IMAGING CONDITIONS

Claerbout's imaging principle

According to Claerbout’s (1971) imaging principle, a reflector exists where the source and the receiver wavefields coincide in time and space. Claerbout expresses the imaging condition as follows:

\[
    r(x, z) = \frac{u(x, z, t_d)}{d(x, z, t_d)},
\]

where \(x\) is the horizontal coordinate, \(z\) is the depth, and \(t_d\) is the time at which the source wavefield \(d(x, z, t_d)\) and the receiver wavefield \(u(x, z, t_d)\) coincide in time and space. This principle states that the reflectivity strength \(r(x, z)\) depends only on the source and the receiver.
wavefields at time $t_d$. The time $t_d$ is not known a priori, therefore we need a practical way to locate the reflector position in the $(x,z)$ plane and compute its strength.

**1-D Crosscorrelation**

A practical way to compute the reflectivity strength in equation (1) is discussed in Claerbout (1971). He computes the reflector strength and position as the zero lag of the crosscorrelation of the source and the receiver wavefields in the time dimension (Figure 2).

![Figure 2: Source and receiver wavefields to be matched in the time dimension.](image)

The previous concept is expressed in the formula:

$$r(x,z) = \sum_{x_s} \sum_{\omega} U(x,z,\omega,x_s)D^*(x,z,\omega,x_s),$$

(2)

where $r(x,z)$ is the zero lag coefficient of the crosscorrelation, which is computed by summation over the frequencies and $U(x,z,\omega)$ and $D(x,z,\omega)$ are the one-dimensional Fourier Transforms of the receiver and the source wavefields, respectively. The contribution of each shot (located at $x_s$) is added to form the final image.

**1-D Deconvolution**

The division in equation (1) is better approximated by implementing a 1-D deconvolution imaging condition in the time dimension. It adds more complexity and potential instability in the computation of the image, but better approximates the definition of the reflection coefficient (ratio between incoming and reflected wave amplitude).

In practice, however, the 1-D deconvolution can be computed in the Fourier domain as a polynomial division. The zero lag coefficient is computed as the sum over frequencies:

$$r(x,z) = \sum_{x_s} \sum_{\omega} \frac{U(x,z,\omega,x_s)D^*(x,z,\omega,x_s)}{D(x,z,\omega,x_s)D^*(x,z,\omega,x_s)} + \epsilon^2(x,z,x_s).$$

(3)

In equation (3), $U(x,z,\omega)$ and $D(x,z,\omega)$ are the one-dimensional Fourier transforms of the receiver and the source wavefields respectively. The contribution of each shot (located at $x_s$) is added to form the final image.
Notice that the regularization parameter \( \varepsilon(x, z, x_s) \) can vary spatially. Jacobs (1982) discuss the difficulties of choosing a spatially variable \( \varepsilon \). In practice, defining \( \varepsilon \) as a function of a dimensionless parameter \( \lambda \) makes its selection easier (Claerbout, 1991). We define this dependence as

\[
\varepsilon^2(x, z, x_s) = \lambda < D(x, z, \omega, x_s)D^*(x, z, \omega, x_s) >,
\]

where \(< >\) indicates the mean in the frequency dimension. \( \lambda \) can be set constant for the whole image because it is independent of the data scale.

A 1-D imaging conditions implicitly make the assumption that each shot contributes to the image with the same weight [equations (2) and (3)]. This assumption is far from reality, since even when the subsurface geometry is not complex, reflectors are illuminated in a different way according to the source position.

### 2-D Imaging Conditions in the \((X_S, T)\) Plane

We can apply an imaging condition that does not assume a uniform contribution of the shots to the image. This can be done by framing the imaging condition as either a 2-D crosscorrelation or a 2-D deconvolution in the shot position/time dimensions \((x_s, t)\). Figure 3 shows a cartoon of the domain where this imaging condition is computed. As showed in Figure 3, we can extract a plane \((x_s, t)\) from the source and the receiver wavefields at each \((x, z)\) position in the image.

![Source and receiver wavefields to be matched in the shot position/time plane.](alejandrol-match2)

#### 2-D Crosscorrelation

The zero lag of the 2-D crosscorrelation of the source and the receiver wavefields in \((x_s, t)\) is the same as the zero lag of the 1-D crosscorrelation in the time domain with a stack across the shot position axis. A simple example with matrices illustrates the concept. If we take the 2-D crosscorrelation of 2 matrices,

\[
\begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{bmatrix} ** \begin{bmatrix} 4 & 1 \\ 2 & 5 \end{bmatrix} = \begin{bmatrix} 12 & 14 & 4 \\ 11 & 25 & 12 \\ 2 & 11 & 5 \end{bmatrix},
\] (5)
the zero lag coefficient is 25. If instead we crosscorrelate the columns,

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} \ast \begin{bmatrix} 4 \\ 2 \end{bmatrix} = \begin{bmatrix} 8 \\ 2 \end{bmatrix}$$

(6)

$$\begin{bmatrix} 2 \\ 3 \end{bmatrix} \ast \begin{bmatrix} 1 \\ 5 \end{bmatrix} = \begin{bmatrix} 3 \\ 17 \\ 10 \end{bmatrix}$$

(7)

take the zero lag, and stack over the rows, the result is also 25. This illustrates the relationship that exists between the 1-D and the 2D crosscorrelation.

### 2-D deconvolution

In the case of deconvolution, the relation between 2-D and 1-D (plus stacking), if it exists, is not straightforward to illustrate. This is because deconvolution in the \((x_s,t)\) plane should be implemented by recursive filtering. However, we know that 2-D deconvolution in the \((x_s,t)\) plane compresses the information in both the shot and the time dimensions. This is a better way of compressing information than 1-D deconvolution followed by a stack in the shot dimension.

In practice, however, 2-D deconvolution can be computed as a polynomial division in the Fourier domain. But in this case, the wavefields that are divided, \(U(x,z,\omega,k_{xs})\) and \(D(x,z,\omega,k_{xs})\), are the two-dimensional Fourier transforms of the receiver and the source wavefields, respectively.

The zero lag coefficient of the 2-D deconvolution is computed as:

$$r(x,z) = \sum_{k_{xs}} \sum_{\omega} \frac{U(x,z,\omega,k_{xs})D^*(x,z,\omega,k_{xs})}{D(x,z,\omega,k_{xs})D^*(x,z,\omega,k_{xs}) + \varepsilon^2(x,z)}$$

(8)

Note that whereas we sum over the shot positions \((x_s)\) in equation (3), we now sum over the shot position spatial frequency \((k_{xs})\) in equation (8).

Also notice that the regularization parameter \(\varepsilon(x,z)\) is spatially variable but constant in the \((\omega,k_{xs})\) plane. It is defined as

$$\varepsilon^2(x,z) = \lambda \ <D(x,z,\omega,k_{xs})D^*(x,z,\omega,k_{xs})>$$

(9)

where < > is the mean in the \((\omega,k_{xs})\) plane.

Analogous to Rickett and Sava (2000), we extend the 2-D deconvolution imaging condition to compute a range of offsets. This is done by shifting the source and the receiver wavefields in the \(x\) dimension. Then, the nonzero-offset reflectivity can be computed as follows:

$$r(x,z,h) = \sum_{k_{xs}} \sum_{\omega} \frac{U(x-h,z,\omega,k_{xs})D^*(x+h,z,\omega,k_{xs})}{D(x+h,z,\omega,k_{xs})D^*(x+h,z,\omega,k_{xs}) + \varepsilon^2(x+h,z)}$$

(10)

where \(h\) is the subsurface offset.

In the following section we show the advantages of 2-D deconvolution imaging condition over the crosscorrelation and the 1-D deconvolution imaging conditions.
Figure 4: Comparison of 3 different imaging conditions. (a) Image obtained with the crosscorrelation imaging condition. (b) Image obtained with the 1-D deconvolution imaging condition in the time dimension. (c) Image obtained with the 2-D deconvolution imaging condition in the \((x,t)\) dimensions.
RESULTS WITH SYNTHETIC DATA

Zero offset image resolution

To compare the result of the different imaging conditions, we build a constant velocity model of five dipping layers pinching-out to the right of the model. The deepest reflector has the steepest dip angle ($\approx 63^\circ$) and the shallowest has zero dip. Figure 4 shows a comparison of three different imaging conditions. Figure 4a uses the crosscorrelation imaging condition, Figure 4b the 1-D deconvolution imaging condition along the time dimension ($\lambda = 0.05$) and Figure 4c the 2-D deconvolution imaging condition in the ($x_s, t$) plane ($\lambda = 0.05$). Notice the better resolution of the 2-D deconvolution image.

The better stability of the 2-D deconvolution image is demonstrated by comparing Figure 5 and Figure 6. They display the result of the 1-D deconvolution and the 2-D deconvolution for different $\lambda$ values at a fixed $x$ position (1.96 km in Figure 4).

Notice that when $\lambda$ decreases, the 1-D deconvolution result presents low frequency noise. The stacking across the shot position [equation (3)] reduces the high frequency noise but might increase some low spatial frequency noise that is periodic with the shot positions. In the case of 2-D deconvolution, when $\lambda$ decreases, the noise contaminates the image in all the spatial frequency bandwidth. However, the signal to noise ratio for this imaging condition is much higher than for the 1-D deconvolution imaging condition.

From this results, we can conclude that the 2-D deconvolution imaging condition in the ($x_s, t$) plane gives a better resolution than the other imaging conditions. The 2-D deconvolution final image is less sensitive to the choice of $\lambda$ and is less affected by the low frequency noise visible in the 1-D deconvolution result. The less sensitivity to $\lambda$ choice is important, since deconvolution major handicap is the selection of the regularization parameter.
CIGs and velocity analysis resolution

Not only the zero-offset image resolution is important. Having a better image resolution for a range of offsets and angles can lead to better estimation of the reservoir properties and better velocity analysis. The angle-domain Common Image Gather (CIG) is the domain where AVA analysis is performed, because it has information about the reflectivity variation with angle (Shuey, 1985). Angle-domain CIGs have also information about how well events are focused at depth, thus providing a natural domain for migration-focusing velocity analysis (Biondi and Sava, 1999).

Following equation (10), we compute offset-domain CIGs with the 2-D deconvolution imaging condition. Figure 7a shows the offset-domain CIGs computed with the 1-D cross-correlation imaging condition. As expected, they show less resolution than the offset-domain CIGs computed with the 2-D deconvolution imaging condition (Figure 7c).

We transform offset-domain CIGs into angle-domain CIGs following the method presented by Sava and Fomel (2000). Figure 7b shows the angle-domain CIGs computed with the 1-D crosscorrelation imaging condition. They also show less resolution than the angle-domain CIGs computed with the 2-D deconvolution imaging condition (Figure 7d).

We now compare the previous gathers with the gathers resulting from the migration of the data with a 3% lower velocity. Figures 8a to 8d display the offset-domain and the angle-domain CIGs computed with the 1-D crosscorrelation imaging condition and the 2-D deconvolution imaging condition for the low velocity, respectively. Notice that with the 2-D deconvolution the curvature of the events, due to the low migration velocity, appears more clear.

Finally, we use the angle-domain Radon transform (ART) methodology described in Sava and Guittton (2003) to estimate the curvature of the reflectors in the angle-domain CIGs. Since the curvature of the reflectors in the ART domain is related to velocity errors, a better curvature estimation can lead to a better velocity estimation. A flat reflector in the angle-domain CIG maps at zero \( q \) (curvature index) in the ART CIGs. Conversely, a reflector with moveout in the angle-domain CIG should map away from zero \( q \) in the ART CIG.
Figure 7: Comparison between 1-D crosscorrelation and 2-D deconvolution imaging conditions. (a) 1-D crosscorrelation offset-domain CIGs. (b) 1-D crosscorrelation angle-domain CIGs. (c) 2-D deconvolution offset-domain CIGs. (d) 2-D deconvolution angle-domain CIGs.
Figure 8: Comparison between 1-D crosscorrelation and 2-D deconvolution imaging conditions for a 3% lower velocity. (a) 1-D crosscorrelation offset-domain CIGs. (b) 1-D crosscorrelation angle-domain CIGs. (c) 2-D deconvolution offset-domain CIGs. (d) 2-D deconvolution angle-domain CIGs.
In Figure 9 we compare the ART CIGs with the correct velocity (Figure 9a and Figure 9b) and the low velocity (Figure 9c and Figure 9d). In both cases we calculated the angle-domain CIGs with 1-D crosscorrelation (Figure 9a and Figure 9c) and 2-D deconvolution (Figure 9b and Figure 9d).

Notice that 2-D deconvolution imaging condition gives ART CIGs that are easier to interpret. Also notice that due to the lack of resolution, ART CIGs computed with 1-D crosscorrelation and the correct velocity Figure (9a) show events that can be misinterpreted as reflectors with the wrong velocity. In addition, ART CIGs computed with 1-D crosscorrelation and the wrong velocity Figure (9c) show events that can be misinterpreted as reflectors with the correct velocity. This problem is reduced with the 2-D deconvolution imaging condition (Figure 9b and Figure 9d).

Figure 9: Comparison between the 1-D crosscorrelation and the 2-D deconvolution imaging conditions for the correct and the low velocity. (a) Angle-domain and ART CIGs computed with the 1-D crosscorrelation and the correct velocity. (b) Angle-domain and ART CIGs computed with the 2-D deconvolution and the correct velocity. (c) Angle-domain and ART CIGs computed with the 1-D crosscorrelation and the low velocity. (d) Angle-domain and ART CIGs computed with the 2-D deconvolution and the low velocity.
CONCLUSIONS

By framing the shot-profile migration imaging condition as a 2-D deconvolution in the \((x_s,t)\) plane the final image resolution is enhanced. The 2-D deconvolution gives better resolution than the crosscorrelation and the 1-D deconvolution imaging conditions.

The 2-D deconvolution imaging condition gives also a more robust result than the 1-D deconvolution imaging condition because it is less dependent on the regularization parameter and it is less affected by the low frequency noise. This is an important property of the 2-D deconvolution, since one deconvolution application major handicap is the selection of the regularization parameter.

In addition, the angle-domain CIGs computed with the 2-D deconvolution imaging condition are less affected by low resolution effects. Consequently, we can estimate better velocities with migration-focusing velocity analysis.

ACKNOWLEDGMENTS

The authors thank Guojian Shan and Brad Artman for providing the programs to construct the source and the receiver wavefields. We also thank Paul Sava, Antoine Guitton, and Gabriel Alvarez for their important suggestions.

REFERENCES


Operator aliasing in wavefield continuation migration

Brad Artman, Jeff Shragge, and Biondo Biondi

ABSTRACT

With the widespread adoption of wavefield continuation methods for prestack migration, the concept of operator aliasing warrants revisiting. While zero-offset migration is unaffected, prestack migrations reintroduce the issue of operator aliasing. Some situations where this problem arises include subsampling the shot-axes to save shot-profile migration costs and limited cross-line shot locations due to acquisition strategies. These problems are overcome in this treatment with the use of an appropriate source function or band-limiting the energy contributing to the image. We detail a synthetic experiment that shows the ramifications of subsampling the shot axis and the efficacy of addressing the problems introduced with our two approaches. Further, we explain how these methods can be tailored in some situations to include useful energy residing outside of the Nyquist limits.

INTRODUCTION

Imaging the subsurface by migration of seismic data may give rise to aliasing problems that can generate artifacts in the final image. In Kirchhoff migration, effective solutions to the problem of operator aliasing have been developed and widely implemented by practitioners (Silva, 1992; Lumley et al., 1994; Sun and Bernitsas, 1999). However, wavefield continuation migration is often considered unaffected by operator aliasing, which is only true for zero-offset applications. Accordingly, with the increasing use of prestack wave-equation migrations, it is important to understand the ramifications of data-space sampling choices on the potential aliasing of the image-space.

Aliasing can arise in wave-equation migration during the application of the imaging condition even when propagating wavefields are alias free. Imaging condition aliasing, as discussed by Zhang et al. (2003), occurs when the image space is inappropriately discretized. We address an aliasing problem that arises when, during the application of the imaging condition, wavenumbers improperly map into the image-space due to sampling changes of the data axes. This situation is analogous to the operator aliasing problem in Kirchhoff migration (Biondi, 2001).

We identify two methods of removing aliased energy from the image-space. We derive the criteria for determining the appropriate Nyquist boundaries for the image due to subsampling a

\[^{1}\text{email: brad@sep.stanford.edu, jeff@sep.stanford.edu, biondo@sep.stanford.edu}\]
data axis. To illustrate these ideas we present a simple case where aliased energy is introduced into the image due to subsampling the shot axis, and show the efficacy of our methods in removing these artifacts. Two important applications are readily identified: 1) calculating the dip information sacrificed when migrating a decimated data set to save computational costs; and 2) optimizing acquisition direction over complex targets in marine towed array surveys.

METHODOLOGY

To demonstrate how operator aliasing arises, it is informative to study an example where improper sampling generates artifacts in the image. When analyzing this problem, it is common to superpose the shot-receiver, $sr$, and location-offset, $xh$, axes. The definition of the transformation between the coordinate pairs dictates how the movement of energy in one frame is related to changes in the other. Even though there is only rotation between the two coordinate systems, the act of overlaying them on the same Cartesian grid inappropriately applies a stretch of $\sqrt{2}$ and gives rise to confusion especially when trying to interpret the Fourier duals of these dimensions. Therefore, these axes are treated separately in this analysis.

Figure 1 explains how we will visualize distributions of energy in the image-space. By transforming the surface location and CIG-offset axes, $(x, h)$, to the Fourier domain variables, $(k_x, k_h)$, we can view the spatial energy components of the image at a reflector depth. The shaded box in the center of each diagram represents the energy delimited by the Nyquist wavenumbers. The thick line at $k_x = 0$ is the 2-D Fourier transform in the $(x, h)$-coordinate system of the image at the depth of a flat reflector. Outside of the shaded region the periodicity of Fourier spectrum, due to discrete sampling, is depicted by two of an infinite number of possible Nyquist replicas (dashed lines). Note that the juxtaposition of all of the replications gives rise to the lines of continuous horizontal energy (dot-dash lines).

In panels a and b, which are the data-space and image-space coordinate representations of energy spectra, sampling requirements are honored and no aliasing occurs. The first set of horizontal energy replications will be the first aliased energy to enter the shaded region if an axis is compressed by decimating the data. In our experiments, we have focused on subsampling the shot-axis as indicated by the filled arrows pointing toward the origin. Panels c and d depict the effects of subsampling the shot axis by a factor of two. The replicated energy bands are compressed toward the origin by a factor of two relative to the overlaying panels. The energy in the right and left columns moves to the same wavenumber values on their respective axes, despite the rotation of the Nyquist boundaries. Although aliased energy has now moved below Nyquist on the $(k_x, k_r)$-plane, the replicated $k_x = 0$ energy is now coincident with the Nyquist boundaries on the $(k_x, k_h)$-plane. However, in the presence of dipping energy ($k_x \neq 0$) aliasing will arise with this degree of subsampling.

We consider two approaches to control the aliasing problems associated with the acquisition and subsampling situations mentioned above. First, wavenumbers from the source and receiver wavefields are band-limited to prevent the entry of aliased duplications into the imaging condition. This does not require eliminating these components from the propagating wavefields, as we can save appropriate portions of the wavefields in disposable buffers for imaging.
Figure 1: Cartoon illustrating a sample energy distribution in the image-space and the hypothetical position of Fourier domain replications (dotted lines) outside of Nyquist boundaries (shaded region). Dot-dashed lines represent replicated energy from the \( k_x = 0 \) band of energy corresponding to a single flat reflector at this image depth. Panels (a) and (c) depict the energy components of the data-space. Panels (b) and (d) depict the energy components of the image-space. The bottom row shows the result of a compression along the \( k_s \)-axis (indicated by the filled arrows) as a result of migrating every other available shot-location.
Second, a band-limited source function is propagated for the duration of each shot-gather migration which effectively zeros energy in the aliased band during imaging. This method manufactures a source function with a wavenumber spectrum limited to the cutoff frequencies imposed by the resampled data axis. There is no additional computational overhead when anti-aliasing with a thick source function, though anti-aliasing by restricting the wavenumbers requires two additional Fourier transforms.

The analytical band-limit is required to appropriately delimit non-aliased wavenumbers for either method. After compression of either data axis, the stretch factor to show the movement of wavenumbers in the \((k_x,k_h)\)-plane is,

\[
\text{BandLimit}(k_x) = \frac{\text{max}(\Delta r, \Delta s)}{\text{max}(a \Delta r, b \Delta s)} \text{min}(N_{y_r}, N_{y_s})
\]

where \((\Delta r, \Delta s)\) are the original receiver and source increments, \((a,b)\) are the down-sampling ratios from the original to the new receiver and source grids, and \((N_{y_r}, N_{y_s})\) are the original Nyquist limits of the data-space. The \text{max} functions in the expression are required because the maximum grid-spacing along either shot or receiver axis alone dictates the aliasing criteria for the \(k_x\)-axis.

The band-limit value calculated with equation (1) is the criteria for eliminating all energy external to the new rigorously defined Nyquist limit. However, if the dipping energy in the data is limited to less than Nyquist, the maximum dip, \(|p|_{\text{max}}\), may be used to relax the band-limiting criteria resulting in the definition:

\[
\text{BandLimit}(k_x, |p|_{\text{max}}) = \frac{N_{y_x} - |p|_{\text{max}}/2}{\text{max}(a \Delta r, b \Delta s)}
\]

Note that if the energy distribution is asymmetric (i.e. \(p_{\text{min}} \neq -p_{\text{max}}\)), two one-sided limits can be implemented. In the case of only zero dip, the band-limit wavenumber restriction is twice the rigorous definition from equation (1).

**EXPERIMENT**

A synthetic data set was generated to test these methods of preventing aliasing. Shot-gathers were modeled over a 2000\(m/s\) earth model with one flat reflector at 1000\(m\). Nominal shot, \(\Delta s\), and receiver, \(\Delta r\), spacing is 10\(m\). Dominant frequency of the wavelet is 30\(Hz\). These experiments were performed using a shot-profile algorithm. However, these analyses and conclusions are valid for both shot-geophone and shot-profile migrations due to their mathematical equivalence (Biondi, 2003),(Shan and Zhang, 2003).

The introduction of aliasing artifacts is seen by comparing the panels in Figure 2 constructed at the depth of the reflector. Panels share the characteristics of Figure 1\(c\), though a frequency axis is now introduced by postponing the summation of frequencies normally required by wave-equation migration imaging conditions. Migrating shot-profiles at every receiver location (left panel) shows marked difference to the wavenumber components that make
up the image produced by migrating shots at every tenth shot (center panel). As the $k_x = 0$ energy moves into the image from the replicated Fourier spectra, bands of aliased energy appear at multiples of $10m^{-1}$ on the $k_x$ axis. Both images were constructed with standard shot-profile migration using a single trace source function.

![Figure 2](image)

Figure 2: Panels show image wavenumbers from migrations of different selections of available shots from the data set. Left panel was created by migrating all available shots, while the center and right panels include only every tenth shot. Note the replication of the flat reflector at $k_x = 2i Ny/10$ in the center panel. The right panel is the result of restricting the migrated energy with a fat source function. Notice that the length of the reflector energy along the $k_x = 0$ axis has been limited from $10m^{-1}$ to $5m^{-1}$ indicating an imposed restriction on the number of offsets contributing to the image at any given dip. Selective energy imaging yields an identical result to the fat wavelet result. [brad1-allv10vfat10] [ER]

The right panel, also produced using every tenth shot, illustrates the effectiveness of our methods in eliminating the aliased energy introduced by subsampling the shot axis. The image does not suffer from aliased energy due to the use of a spatially band-limited (fat) source function. It also shows a diminished $k_h$ bandwidth as a result of removing the aliased energy.

Selective energy imaging, our second proposed method, uses band-limited versions of the source and receiver wavefields in the imaging condition. The resultant image is identical to the right panel and is accordingly not shown.

**CONCLUSIONS**

We conclude that operator aliasing artifacts are indeed introduced during prestack wavefield continuation migration when $s$- or $r$-axes are allowed to deviate from a regular grid. This problem is thus manifest both when we choose to subsample the number of shots available from the data or design a survey without shots at every receiver location (or in reciprocal cases
like OBC surveys). We have shown two methods for eliminating aliased contributions to the image.

We feel that it is more appropriate to use selective energy imaging conditions rather than a fat source function. In practice, lateral velocity variation will cause energy constituents in the \( f_k \)-plane to move around. To allow for any beneficence from this movement, it would be unwise to propagate a band-limited source function, or worse, to eliminate energy from the source and receiver wavefields during propagation steps. Therefore we recommend migrating individual shots on the fine grid, and accounting for image aliasing in the imaging step rather than propagating data decimated to match the grid of the subsampled shot axis (or vice-versa).

With some knowledge of the dip content of the data it is possible to extend the boundaries of anti-aliasing limit criteria. Using both positive and negative one-sided band-limits can allow for the inclusion of appropriate energy into the image that resides outside of the rigorously defined Nyquist boundaries. Therefore, when challenged with imaging important steeply dipping targets, decisions concerning acquisition design or the level of decimation along different directions for migration can be made with a better understanding of the consequences to the final product.

REFERENCES


Phase-shift migration of approximate zero-offset teleseismic data

Jeff Shragge

ABSTRACT

A hybrid of traditional survey-sinking migration is derived that is applicable to teleseismic wavefields. To reconfigure teleseismic data to an approximate equivalent of zero-offset, an adjoint linear moveout shift is applied. This transformation enables the straightforward development of phase-shift operators to downward continue the modified teleseismic data. This method also affords an opportunity for imaging earth structure with a variety of forward- and backscattered modes through appropriate choices of wavefield velocities. This method is applied to a synthetic teleseismic data set, and several migration results are presented to demonstrate its effectiveness.

INTRODUCTION

The use of idealized plane-wave sources in exploration seismic imaging is well documented (Claerbout, 2001). One situation where plane-wave sources are often realized is in imaging with earthquake waves recorded at large epicentral distances (i.e. teleseismic distances). Many advances in teleseismic imaging have stemmed from adapting existing seismic exploration techniques to the teleseismic context. In particular, much recent effort has been centered on reconfiguring multi-dimensional Kirchhoff-based migration/inversion formulations for teleseismic plane-wave sources and acquisition geometry (Bostock et al., 2001).

Multi-dimensional wave-equation migration methods are not applied widely in teleseismic imaging. Although the advantages of wave equation-based methods over their Kirchhoff counterparts are known to explorationists, two issues persistently hinder their use in the teleseismic community. One problem is that Fourier-based methods are difficult to apply to data of non-uniform spacing. In practice, this is often observed because of practical limitations in acquisition. A second issue is that the geometry of teleseismic sources and data needs to be reconciled with the geometry of traditional pre- and post-stack migration formulations. Hence, both of these issues must be addressed before wave-equation migration algorithms are applied.

The aim of this paper is to show that, with only minor adjustments, traditional wave-equation migration is readily applicable to teleseismic wavefields (i.e. by accounting for the effects of upward propagating plane-wave sources). This paper specifically addresses the second issue by demonstrating that the application of an adjoint linear moveout (LMO) operator successfully reconfigures teleseismic data to an approximate equivalent of traditional zero-offset
data. Consequently, migration of data with a hybrid of traditional survey-sinking methodology becomes possible. The method can use different source and receiver wavefield velocities, which permits imaging with converted waves present in most three-component earthquake records. Although this paper does not address the first issue directly, there is at least one method to successfully re-grid irregularly spaced teleseismic data to a regular mesh (Curry, 2003).

I begin with a general discussion of what source and receiver wavefields mean in the context of teleseismic imaging. I develop modified double square root operators appropriate for downward continuing teleseismic data, and discuss the choice of velocities required to image different converted modes. The method is then applied to a synthetic data set generated by finite differencing plane wave sources through a simplified lithospheric subduction zone model.

**METHODOLOGY**

The first stage in generating the migration algorithm is to develop an analytic expression for the propagation of source wavefields through the earth. To do this, two approximations are employed. First, I assume that earthquake wavefronts at teleseismic distances (between 30° and 103° in epicentral distance from source) can be approximated by plane waves. This approximation holds because at these distances and array lengths there is little curvature in the earthquake wavefront. The second approximation is that the source propagates through a vertically-stratified, $v(z)$ medium. This approximation is consistent with the notion that earth structure ‘visible’ at teleseismic frequencies (i.e. between 0.005 and 4 Hz) is predominately $v(z)$. Accordingly, the source wavefront is parameterized with constant horizontal ray parameters in a Cartesian system. This enables an analytic calculation of the planar source wavefield at all times. When discussing the source wavefield henceforth, I am referring to its analytic expression.

To enable the use of the survey-sinking algorithm, teleseismic data need to be reconfigured to an approximate equivalent of zero-offset geometry. One way that this can be achieved is to exploit the fact that planar teleseismic wavefields sweep across recording arrays in a linear fashion. This leads to a LMO of first arrivals in the recorded teleseismic section, with the degree of moveout being dependent on the incident wave’s orientation to the strike of the array, as shown in Figure 1. Thus, as an initial approximation, I apply an adjoint LMO operator to remove the LMO recorded in the data. (The processing flow is depicted in Figure 2).

In this new reference frame, zero time is defined by the arrival of the teleseismic source at each station. Since all arrivals now occur at zero-time, this may be considered as a zero-offset experiment. Note that reflections and mode conversions (i.e. P- to S-wave) from horizontal discontinuities are flattened by this transformation. One drawback, though, is that diffracted hyperbolas symmetric in the initial reference frame are now skewed to one side of the hyperbola apex. The degree to which they are skewed is dependent on the angle of the applied adjoint LMO (see Figure 3). While this may not be an optimal physical representation, two observations support the use of this approximation. First, coherent events observed in the re-
receiver wavefield predominately consist of planar reflected and converted events. These types of scattered energy are properly migrated by a zero-offset wave-equation method. Second, little-to-no LMO shift is applied to plane waves arriving from directions nearly or exactly coincident with the strike-axis of the recording array. In these cases planar events (plus diffractions if present) are properly migrated by a zero-offset algorithm.

Double square root equations for teleseismic geometry

The theory developed here is similar to the traditional survey-sinking migration approach where the source and receiver wavefields are downward continued into the earth (Claerbout, 2001). However, in the traditional application of zero-offset survey sinking downward continued source wavefields do not have an intrinsic initial horizontal velocity. In the teleseismic case, planar teleseismic sources nearly always have non-zero horizontal velocities. This discrepancy is reconciled by deriving modified survey-sinking equations.

To help conceptualize the equations required for this context, consider a reference frame
moving with a horizontal velocity equal to that of the source wavefield. In this reference frame a dipping plane wave propagates vertically. Hence, the downward continuation operators applied to recorded teleseismic data must transform to the double square root equations in a reference frame moving with a horizontal velocity equal to that of the source. To enable source horizontal propagation, wavefields downward continued to the next depth step are now dependent on horizontal position \((x)\) in addition to depth \((z)\). That is,

\[
dt = dt(x_g, z_g, x_s, z_s),
\]

(1)

where subscripts \(s\) and \(g\) correspond to source and receiver wavefields, respectively. To begin the derivation the differential is expressed explicitly,

\[
dt = \frac{\partial t}{\partial x_g}dx_g + \frac{\partial t}{\partial z_g}dz_g + \frac{\partial t}{\partial x_s}dx_s + \frac{\partial t}{\partial z_s}dz_s,
\]

(2)

where partial derivatives with respect to the \(x\) and \(z\) coordinates may be readily associated with wavefield parameters. In a \(v(z)\) medium the constant horizontal slowness of the source wavefield, \(p\), is defined by

\[
\frac{\partial t}{\partial x_s} = p,
\]

(3)

the vertical slowness of source wavefield by

\[
\frac{\partial t}{\partial z_s} = \sqrt{\frac{1}{v_s^2} - p^2},
\]

(4)

the variable horizontal slowness of the scattered receiver wavefield, \(p_{scat}\), by

\[
\frac{\partial t}{\partial x_g} = p_{scat},
\]

(5)

and the vertical slowness of receiver wavefield by

\[
\frac{\partial t}{\partial z_g} = \sqrt{\frac{1}{{v_g}^2} - {p_{scat}}^2},
\]

(6)

where \(v_s\) and \(v_g\) represent the source and receiver wavefield velocities, respectively. Differential pairs \(dx_g\) and \(dz_g\), and \(dx_s\) and \(dz_s\) are linked geometrically by

\[
\begin{align*}
  dx_g &= \tan \theta_g dz_g, \\
  dx_s &= \tan \theta_s dz_s,
\end{align*}
\]

(7, 8)

where \(\theta_s\) and \(\theta_g\) represent angles of the source and receiver propagation direction with respect to the \(z\)-axis, respectively. Replacing trigonometric expressions in equations (7) and (8) with ray parameter equivalents, and by downward continuing the source and receiver wavefields by equal depth steps (i.e. \(dz = dz_g = dz_s\)) one may rewrite equation (2) as

\[
\frac{\partial t}{\partial z} = \frac{p^2 v_s}{\sqrt{1 - p^2 v_s^2}} + \frac{\sqrt{1 - p^2 v_g^2} v_s}{v_s} + \frac{p_{scat}^2 v_g}{\sqrt{1 - p_{scat}^2 v_g^2}} + \frac{\sqrt{1 - p_{scat}^2 v_g^2} v_g}{v_g}.
\]

(9)
Equation (9) demonstrates that the effects of the propagating source wavefield are transferred to the receiver wavefield. Accordingly, using the chain rule and algebraic manipulation, the partial derivative of the receiver wavefield, \(U\), with respect to depth is

\[
\frac{\partial U}{\partial z} = \frac{\partial U}{\partial t} \frac{\partial t}{\partial z} = \left( \frac{1}{v_s \sqrt{1 - p^2 v_s^2}} + \frac{1}{v_g \sqrt{1 - p_{scat}^2 v_g^2}} \right) \frac{\partial U}{\partial t}. \tag{10}
\]

Relating Fourier domain variables to the angular components of the individual plane waves

\[
p_{scat}(v_g(z)) = \frac{\sin \theta_g}{v_g} = \frac{k_x}{\omega}, \tag{11}
\]

and applying a Fourier Transform over the temporal coordinate in equation (10) yields

\[
\frac{\partial U}{\partial z} = -i\omega \left( \frac{1}{v_s \sqrt{1 - p^2 v_s^2}} + \frac{\omega}{v_g \sqrt{\omega^2 - k_x^2 v_g^2}} \right) U. \tag{12}
\]

The solution to differential equation (12) expressed in a discrete sense is,

\[
U(z + \Delta z, \omega, p) = U(z, \omega, p) \exp \left[ -i\omega \Delta z \left( \frac{1}{v_s \sqrt{1 - p^2 v_s^2}} + \frac{\omega}{v_g \sqrt{\omega^2 - k_x^2 v_g^2}} \right) \right]. \tag{13}
\]

With this expression, the receiver wavefield is downward continued and the image of the reflector at each model point \((x,z)\) constructed through the usual imaging condition (Claerbout, 1971),

\[
I(x, z) = \sum_{\omega} U(s = x, g = x, z, \omega). \tag{14}
\]

Although the method derived here is strictly for 2-D geometry, it may be extended to account for out-of-plane propagation by assuming an underlying 2-D medium and accounting for the effects of 2.5-D propagation through incorporation of the invariant cross-line horizontal slowness. This method is also potentially applicable in 3-D, but this is not explored here.

**Apex correction due to LMO shift**

The LMO operation leads to vertical time shifts in the data that incorrectly position the scattering point horizontally. Figure 3 illustrates the kinematics of the LMO operation. In Figure 3, a light gray line connects the energy scattered from the discontinuous structure shown in Figure 1 with the associated source point on the incident wavefront. As illustrated in Figure 3b, after the LMO operation the scattering point must be shifted horizontally so that it falls beneath the corresponding source point on the source wavefront. This is a requirement of zero-offset migration. Noting that the relationship between the horizontal change in apex location, \(dx_{apex}\), as a function of depth, \(dz\) is,

\[
dx_{apex} = \tan \theta_s dz, \tag{15}\]
the required horizontal shift operator is derived by evaluating a Fourier transform ($\mathcal{F}$) of the vertical component of velocity from depth $z$ to the surface,

$$\mathcal{F} (\delta(x - x_0)) = \mathcal{F} \left( \delta(x - \int_0^z \text{d}z \frac{pv_s}{\sqrt{1 - p^2v_s^2}}) \right) = \exp \left( -ik_x \int_0^z \text{d}z \frac{pv_s}{\sqrt{1 - p^2v_s^2}} \right).$$ (16)

Note that equation (16) is dependent solely on the $v = v(z)$ velocity profile and can be applied either pre- or post-migration. I have chosen to apply this shift post-migration (Figure 2).

Figure 3: Diagram of LMO transform to approximate equivalent zero-offset section. a) Before LMO correction hyperbola is symmetric about the scattering point. The light gray line links points of scattered energy to the location on the wavefront causing scattering. b) After the LMO correction the hyperbola and ‘Imaging box’ are skewed. In-filled area illustrates the travel-time error introduced by LMO operation. Note that the hyperbola apex has zero error. The tilted light gray line shows the non-centered location of apex hyperbola. This line needs to be straightened out through a horizontal shift so that it falls beneath the scattering point.

**Velocities**

In the theory developed above, there is an allowance for differing source and receiver wavefield velocities ($v_s$ and $v_g$). This occurs with different wavefield scattering modes. Figure 4 illustrates the primary elastic scattering modes arising in teleseismic imaging for an incident planar P-wave source. In the forward-scattering scenario, two modes arise when the incident wavefront refracts/converts from a given scattering point. The same incident P-wavefront also reflects/converts downward from the free-surface and then reflects/converts upward from the same scattering point. Accordingly, backscattering gives rise to four in-plane scattering modes (plus one out-of-plane SV to SH scattering mode; see Table 1).

To represent the difference in propagation direction of source wavefronts in the forward- and backscattering scenarios, a negative velocity is used. That is, for forward- and backscattering, velocities of opposite and like sign are used, respectively. This is the equivalent of
Figure 4: Diagram of primary scattering elastic scenarios in teleseismic imaging. Incident P-wavefront directly refracts/converts from a scattering point, but also may reflect/convert downward from the free-surface and then reflect/convert upward from the scattering point. Legs spent as P- and S-waves are drawn in black and light gray, respectively.

Choosing to take the complex conjugate of both the source and receiver wavefields in the forward-scattering scenario (Claerbout, 1971). Table 1 in the Appendix A gives the velocities used for all six primary elastic scattering modes.

SYNTHETIC MODEL AND DATA

The idealized lithospheric suture model, shown in Figures 5a and 5b, is defined on a 300×120 km² section and consists of three materials with differing elastic properties. A low-velocity crustal layer (white) overlies a faster upper mantle (gray). (See Table 2 in the Appendix A for model velocities and densities). At the location of the suture, crustal material from the lithospheric block to the left bifurcates, with the lower segment descending into the mantle. At a depth of ∼40 km, this relict (black) converts to velocities and density higher than the surrounding mantle (note the proportionally greater increase in S-velocity) and thereafter folds and thins to the right of the model.

Several sets of two-component seismograms were computed through the lithospheric model using a 2-D, elastic pseudo-spectral code (Kosloff et al., 1990). The seismograms comprise a suite of plane P-wave sources interacting with the model over a range of incident horizontal slowness, \( p = [0.05, -0.05, 0.06, -0.06, 0.07, -0.07] \text{ s km}^{-1} \). The output seismogram sections consist of 120 traces computed at 3km intervals at the free surface.

One preprocessing requirement is that a reasonable zero time mark is computed for all traces. The method employed here consists of transforming raw data sections, \( U=[U_1, U_3] \), into up-going P- and S-wavefield sections, \( w=[P, SV] \), via the free-surface transfer matrix (Kennett, 1991). Multi-channel cross-correlation (VanDecar and Crosson, 1990) it then applied to a window about the direct P-arrival to allow optimal alignment of wavefield sections. A representative set of synthetics seismograms is presented in Figures 5c and 5d. Generally, the two sections are characterized by a combination of spectral sub-planar reflectors, and diffractions from the higher (spatial frequency) wavenumber model structure.
Figure 5: Idealized, three material collisional suture model and representative finite difference data. a) P-velocity model (Velocities and densities of each material are given in Table 2); b) S-velocity model; c) Sample processed synthetic section from model in a) and b) rotated into P wave vector; d) As in c) but rotated SV wavefield.

**EXPERIMENTS**

The results of applying the hybrid survey-sinking algorithm to the synthetic data set are presented in this section. The $v(z)$ velocity profiles used in these migrations consist of a 40-km-thick crust overlying an upper mantle half-space. The values of the P- and S-wave velocities employed are given in the first two rows of Table 2.

Figure 6a presents the forward-scattered P-S migration result for the SV component shown in Figure 5c where the source is incident from the right. Both the crust-mantle (white-to-gray in model) and relict crust-mantle (black-to-gray in model) discontinuities are well imaged, but the crust-relict crust (white-to-black in model) boundary is notably absent due to weak amplitudes of incidence-angle dependent P-S conversions. Figure 6b presents the backscattered P-P migration result for the P-wave component presented in Figure 5d. The structure imaged in this case is predominantly the crust-mantle discontinuity. This is expected since the magnitude of the P-P reflection coefficient for the mantle-relict crust discontinuity is very weak as
the change in P-wave velocity is less than two percent. Also note the vertical mis-positioning of structure in Figures 6a and 6b arising due to use of a $v(z)$ reference model.

Figure 6: Results for applying migration algorithm to data shown in Figures 5c and 5d. (a) Forward-scattered P-S image; (b) Backscattered P-P image.

Figure 7 presents the results of stacking the migrations of all 6 generated events for 4 of the 6 primary elastic scattering modes. Results show good reconstruction of the crust-mantle discontinuity. Panels (a), (c) and (d) also exhibit recovery of the relict crust, though with differing relative amplitudes. Forward-scattered P-S and backscattered P-P (Panels a) and b)) do not suffer from cross-mode contamination to the same degree as modes in panels c) and d). This is attributed to these two modes being the first scattered arrivals in their respective wave sections. These modes are themselves separated by the orthogonality of waves ending in P- and S-phases.

Forward-scattered P-P and backscattered S-P images are not shown since they are severely degraded. Forward-scattered P-P energy has little-to-no sensitivity to travel-time variations since the scattered wavefield moves at the same velocity and in the same direction as the source wavefield. This yields small-to-zero time delays insufficient for acceptable resolution of model structure. Backscattered S-P energy is twice converted and, over the range of angles considered here, has a much lower amplitude relative to the earlier arriving P-P energy.

CONCLUSIONS

The application of an adjoint LMO operator to teleseismic data successfully transforms recorded teleseismic wavefields to an approximate equivalent to traditional zero-offset. This transformation enables the use of zero-offset migration techniques. The hybrid survey-sinking method developed here downward continues all primary elastic scattering modes, and produces interpretable images.
Figure 7: Stack of all events for given scattering modes. a) P-S forward-scattering; b) P-P backscattering; c) P-S backscattering; and d) S-S backscattering.
Migration results for the lithospheric test model show a good recovery of the model structure. The absence of some model structure in the images is attributed to a combination of poor illumination and weak scattering amplitudes. Imaged discontinuities are slightly displaced from their true locations due to the $v(z)$ velocity model used. In addition, diffractions do not collapse perfectly back to point-like scatterers. The general success of the derived zero-offset survey-sinking methodology and the possibility of undertaking more advanced analysis of angle-dependent scattering behavior further motivate the study of wave-equation imaging of teleseismic data (Shragge and Artman, 2003).

REFERENCES


Claerbout, J. F., 1971, Toward a unified theory of reflector mapping: Geophysics, 36, no. 03, 467–481.


APPENDIX A

Table 1. Velocities used for different scattering modes ($\alpha$ and $\beta$ are P- and S-wave velocities, respectively).

<table>
<thead>
<tr>
<th>Scattering Mode Index</th>
<th>Scattering Mode</th>
<th>$\tilde{v}$</th>
<th>$\hat{v}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FS P-P</td>
<td>$\alpha$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>2</td>
<td>FS P-S</td>
<td>$\alpha$</td>
<td>$\beta$</td>
</tr>
<tr>
<td>3</td>
<td>BS P-P</td>
<td>$-\alpha$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>4</td>
<td>BS P-S</td>
<td>$-\alpha$</td>
<td>$\beta$</td>
</tr>
<tr>
<td>5</td>
<td>BS S-P</td>
<td>$-\beta$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>6</td>
<td>BS S-S</td>
<td>$-\beta$</td>
<td>$\beta$</td>
</tr>
</tbody>
</table>

Table 2. Lithospheric Model Parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>P-wave velocity (km s$^{-1}$)</th>
<th>S-wave velocity (km s$^{-1}$)</th>
<th>Density (g cm$^{-3}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crust</td>
<td>6.2</td>
<td>3.6</td>
<td>2.8</td>
</tr>
<tr>
<td>Mantle</td>
<td>8.0</td>
<td>4.5</td>
<td>3.2</td>
</tr>
<tr>
<td>Relict Crust</td>
<td>8.1</td>
<td>4.9</td>
<td>3.3</td>
</tr>
</tbody>
</table>
Imaging with buried sources

Jeff Shragge and Brad Artman

ABSTRACT

Because the shot-profile migration algorithm largely mimics the data acquisition process, simple thought experiments may extend its utility to image the subsurface with less conventional geometries and/or sources. Imaging with the forward- and backward-scattered wavefields in an elastically modeled earth from buried sources is easily implemented without the development of new tools. With this potential in mind, we identify several novel applications of this wave-equation imaging technique, detail the requirements and processing required for its success, and give an example of the process and results by applying these concepts to a crustal-scale imaging experiment using emergent teleseismic plane-waves as sources.

INTRODUCTION

Conventional seismic reflection surveys are largely constrained to sources detonated on the surface of the earth. Correspondingly, most of the advanced imaging techniques of proven value have been developed to migrate these types of data sets. One subset of sources generally excluded from the purview of seismic imaging is those originating at depth, including both active (e.g. VSP) and passive (e.g. micro-fracture) generated energy. The predominant reason for their omission is because of a lower applicability during the exploration stage. This is largely due to longer acquisition times, lower average source S/N ratios, the need for three-component data, and the irregularity of source distributions. However, the generality of wave-equation methods allows for the adaptation of exploration-oriented imaging techniques to novel non-exploration geometries. This fact provides the motivation for identifying a number of candidate experiments where buried sources may be used to image the subsurface using adapted imaging techniques.

Shot-profile migration shows an impressive adaptivity to acquisition geometries incorporating subsurface sources. One of this method’s strengths is that reconstructions of source wavefields originating in the subsurface easily honor the geometry of individual experiments. These wavefields are used, in turn, to image components of both the forward- and back-scattered wavefields. This is accommodated by a proper selection of the sign of the exponential used to propagate the wavefields. Further, the ability to choose different velocity models by which to propagate each wavefield adds the ability to image with converted energy modes.
We propose the application of seismic imaging methods to a suite of problems centered on the utilization of energy originating from buried sources. Geometries potentially suitable for imaging include both natural and induced fracture point sources, teleseismic plane waves, downhole VSP, similar to Harwijanto et al. (1987), and borehole tomography. All of these applications require a minor amount of pre-processing of the scattering wavefields contained in recorded data, the construction of appropriate source wavefields, and the generation of appropriate compressional and shear velocity models for use in the shot-profile migration algorithm.

To these ends, we present an overview of the processing flow, and point of some of the difficulties likely to be encountered when imaging with buried sources. After a general discussion of the methodology, we then apply the processing flow to the case of teleseismic imaging for a demonstration and to provide a proof of concept.

METHODOLOGY

In this section, we outline the steps for preparing data from buried sources for imaging. Figure 1 illustrates the work flow. Central to each of these new applications is the careful attention to the geometry and characteristics of the source wavefield. To be able to utilize these alternative sources, in many cases the first stage is determining when and where an event has occurred. The primary energy mode (i.e., P or S-wave) and the direction of first motion must then be determined to enable characterization of the available scattered phases. After this information is ascertained, an appropriate source wavefield is constructed for input to the migration.

Figure 1: Flow chart illustration the steps to prepared data from buried sources for multi-mode migration. Solid line shows overall direction of processing flow. Dashed lines show dependancies. [Jeff2-Flow][NR]

After identifying source parameters, time zero and the total length of the record to be migrated is determined. Three-component receiver arrays are necessary to exploit the full range of wavefield scattering combinations in these experiments. With the introduction of vector displacements, a rotation of data components is defined that maps energy from the recording geometry \([Z,N,E]\) to an alignment with source wave vector axes \([P,SV,SH]\).
The Table below summarizes all of the information required to generate images from various combinations of scattering modes for a source with an initial P-wave polarization. In the first column, FS and BS represent forward- and backscattering, respectively. Forward-scattered wavefields are introduced in a manner akin to the exploding reflector model of seismic imaging. The forward-scattered source wavefield propagates through the model space in the same direction as the data wavefield. This introduces the only substantiative modification to the shot-profile algorithm. This is implemented by a change in the sign of the exponential in the SSR equation (column 2). Additionally, the interaction of the primary source with the free-surface gives rise to downward reflected P- and converted S-wavefields that are, in turn, back-scattered in a manner similar to conventional reflection experiments. The second field in the first column indicates the various mode transitions available for use in migration. These are abbreviated by their source and receiver phases.

The third column contains the source and receiver velocities required to produce the desired image. The final column identifies the receiver component in which the individual scattering components are expected. Notice that there are a number of different modes in the $P$ and $SV$ sections and, accordingly, cross-mode contamination will occur.

<table>
<thead>
<tr>
<th>Scattering Mode</th>
<th>Source Prop. Dir.</th>
<th>S. Velocity</th>
<th>R. Velocity</th>
<th>Rec. Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>FS P-P</td>
<td>-</td>
<td>$P$</td>
<td>$P$</td>
<td>$P$</td>
</tr>
<tr>
<td>FS P-S</td>
<td>-</td>
<td>$P$</td>
<td>$S$</td>
<td>$SV$</td>
</tr>
<tr>
<td>BS P-P</td>
<td>+</td>
<td>$P$</td>
<td>$P$</td>
<td>$P$</td>
</tr>
<tr>
<td>BS P-S</td>
<td>+</td>
<td>$P$</td>
<td>$S$</td>
<td>$SV$</td>
</tr>
<tr>
<td>BS S-P</td>
<td>+</td>
<td>$S$</td>
<td>$P$</td>
<td>$P$</td>
</tr>
<tr>
<td>BS S-S</td>
<td>+</td>
<td>$S$</td>
<td>$S$</td>
<td>$SV$</td>
</tr>
<tr>
<td>BS S-S</td>
<td>+</td>
<td>$S$</td>
<td>$S$</td>
<td>$SH$</td>
</tr>
</tbody>
</table>

The ability to realize these conditions is highly dependent on the nature of the candidate buried source. Specifically, imaging with local earthquakes and induced micro-fractures requires a prior inversion for a source’s nucleation location and rupture time. Additional issues are the correct identification of dominant energy mode and polarization, and perhaps a deconvolution of source functions of potentially significant complexity. In applications to reservoir monitoring or the imaging of a major fault zone (e.g., San Andreas), a proliferation of 3-C receivers at the surface and incorporation of existing velocity model provides background information to locate and characterize micro-tremor sources. Down-hole geometries make realizing these conditions somewhat easier since the spatial and temporal location of the source are known.

**EXPERIMENTS**

In this section, we present an example of the application of the shot-profile migration algorithm to a synthetic teleseismic earthquake data set. (For additional information regarding the synthetic data set refer to Shragge (2003)). The first step in imaging process is identifying the probing energy, and separating from all other phases emanating directly from the source. In teleseismic imaging, this is simplified by the fact that P- and S-waves are well separated in
time at large epicentral distances due to differences in P- and S-wave velocity magnitude. In this case, we use P-waves because the provide a better S/N ratio than latter arriving phase. To characterize the teleseismic source, we exploit the fact that teleseismic arrivals are planar after traversing large distances (Shragge, 2003). Knowing the location of the earthquake, precise dip and azimuthal orientation of these plane waves can be calculated using standard 1-D reference earth models. Having thus characterized the source, we can construct a wavefield for use in the migration to come. An example of a modeled teleseismic source wavefield is presented in Figure 2a.

![Figure 2](image)

Figure 2: a) Constructed source wavefield; b) SV component of the (rotated) receiver wavefields. The data are immersed in a large field of zeros to allow for the correct migration of back-scattered energy that does not reflect from the free-surface within the length of the receiver spread.

The first stage in preprocessing the data is to identify zero time. Although this generally involves time-windowing about the estimated arrival time, with synthetic data it is a straightforward task. The dip and azimuth parameters from source characterization are used to define the appropriate receiver wavefield rotation from [Z,N,E] to [P,SV,SH]. Figure 2b presents one component of the rotated data.

We present the results of two different migrations in Figure 3. Figure 3a and 3c present the forward-scattered P-S converted mode and the backscattered P-P mode images, respectively. The model used to generate the finite-difference data is presented for reference in Figure 3b. The appropriate exponent and velocity models for are given in Table 1. The incorporation of $v(x,z)$ velocity models in the wave-equation migration leads to better positioning of model structure relative to other teleseismic imaging techniques (Shragge et al., 2001; Shragge, 2003). This method also affords all the common advantages of wave-equation depth imaging in complex media.

**CONCLUSIONS**

Imaging the subsurface with wave-equation migration algorithms can be extended in exciting ways to encompass new experiments. Combinations of natural or induced seismicity, and sur-
Figure 3: a) Forward-scattered P-S mode image; b) Synthetic Model; c) Backscattered P-P mode image.
face and down-hole sources and 3-C receivers has the potential to greatly extend the scope of imaging techniques. To demonstrate the generalized processing flow, we have used a synthetic buried source to generate two complementary images of model structure. By extension, the same methodology can be used to produce five additional images using other scattering modes extant in the teleseismic section. Buried sources closer to our array undergo a transition from plane-waves to point-like sources. We speculate that these can also be utilized for imaging by following the procedures outlined above. The application of these ideas would be most readily accomplished in cross-well tomography and VSP experimental geometries. The potential for reservoir monitoring using pressure-change induced micro-fracture sources during exploitation or $CO_2$ sequestration is also extremely intriguing.

**REFERENCES**


Improving the amplitude accuracy of downward continuation operators

Ioan Vlad, Thomas Tisserant, and Biondo Biondi

ABSTRACT

While wave-equation downward continuation correctly accounts for traveltimes, the amplitude and phase of the image can be improved. We show concrete ways of implementing a previously proposed improvement using both mixed-domain and finite-difference extrapolators. We apply the corrections to constant velocity, constant vertical velocity gradient and \( v(x, z) \) cases and show that the correction brings amplitudes closer to the theoretical values.

INTRODUCTION

While accurate kinematics are the first requirement of a useful seismic image, its amplitude and phase behavior are nonetheless important. The amplitude and phase information can be useful in wave-equation migration velocity analysis or as direct hydrocarbon indicators. The one-way wavefield extrapolators (Claerbout, 1971) employed in migration produce correct traveltimes. However, their amplitude and phase characteristics can be improved.

Zhang (1993) brings theoretical insight into the problem of improving the amplitudes of downward continuation operators. Zhang et al. (2001) compare them with Kirchhoff inversion and provide formulas for amplitude-preserving shot-profile migration that are accurate in the case of constant velocity. Along the same lines, Zhang et al. (2002) give an additional term for the case of velocity varying only with depth. Zhang et al. (2003a) extend the argument to heterogeneous media. Besides that, Zhang et al. (2003b) provide a more accessible description of the basic theory of the method. In this paper, we describe implementations of this amplitude-preserving downward continuation operators in the \( \omega - x \) domain (finite-difference), as well as in the mixed domain.

ANALYTICAL FORMULATION

Denote the surface position vector by

\[ \vec{x} = (x_1, x_2), \]

\(^1\text{email: nick@sep.stanford.edu, thomas@sep.stanford.edu, biondo@sep.stanford.edu}\)
the temporal frequency by $\omega$, and the velocity by

$$v = v(\vec{x}, z).$$

(2)

We define the operator $\Lambda$ as

$$\Lambda = i \sqrt{\frac{\omega^2}{v^2} + \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2}},$$

(3)

and the operator $\Gamma$ as

$$\Gamma = \frac{1}{2v} \frac{\partial v}{\partial z} + \frac{1}{\omega^2} \left( \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} \right).$$

(4)

Without the amplitude correction, shot profile migration proceeds by computing the upward going and the downward going wavefields at all depths:

$$\left( \frac{\partial}{\partial z} - \Lambda \right) U = 0,$$

(5)

$$\left( \frac{\partial}{\partial z} + \Lambda \right) D = 0.$$

(6)

To solve these equations, we need boundary conditions (inputs into the downward continuation). The recorded shot gather is taken as the upper boundary condition for the upward going wavefield, and a Dirac, approximated by a small wavelet, is taken as the upper boundary condition for the downward going wavefield:

$$U(\vec{x}, z = 0, \omega) = \text{FFT} \left[ \text{Shotgather}(\vec{x}, t) \right]_{(\omega)},$$

(7)

$$D(\vec{x}, z = 0, \omega) = \text{FFT} \left[ \text{Shot}(\vec{x}, t) \right]_{(\omega)} = \text{FFT} \left[ \delta(\vec{x} - \vec{x}_{\text{shot}}, t) \right]_{(\omega)}.$$  

(8)

The reflectivity image is produced using the imaging condition:

$$R(\vec{x}, z) = \sum_{\omega} \frac{U(\vec{x}, z, \omega)}{D(\vec{x}, z, \omega)}.$$

(9)

According to Zhang et al. (2003a), to improve the amplitude and phase of the image $R$, we only need to add the $\Gamma$ operator into the downward continuation equations:

$$\left( \frac{\partial}{\partial z} - \Lambda - \Gamma \right) p_{U} = 0,$$

(10)

$$\left( \frac{\partial}{\partial z} + \Lambda - \Gamma \right) p_{D} = 0.$$

(11)

and while taking the same boundary conditions for the upward going wavefield, to apply a correction to the downward going wavefield boundary condition:

$$p_{U}(\vec{x}, z = 0, \omega) = \text{FFT} \left[ \text{Shotgather}(\vec{x}, t) \right]_{(\omega)},$$

(12)

$$p_{D}(\vec{x}, z = 0, \omega) = \frac{1}{2} \Lambda^{-1} \text{FFT} \left[ \text{Shot}(\vec{x}, t) \right]_{(\omega)}.$$  

(13)
The imaging condition remains the same:

\[ R(\vec{x}, z) = \sum_\omega \frac{p_U(\vec{x}, z, \omega)}{p_D(\vec{x}, z, \omega)}. \] (14)

The only differences between the amplitude-preserving algorithm and the traditional one are the use of the \( \Gamma \) operator in downward continuation and the application of the \( \Lambda^{-1} \) operator to the shot boundary condition. In the following sections, we will describe implementations of the two operators in the \( \omega - x \) domain (finite-difference), as well as in the mixed domain. We will also show 2-D synthetic examples of applying the operators.

**IMPLEMENTING THE \( \Lambda^{-1} \) OPERATOR**

**Finite differences**

We denote the source wavefield by \( q(x, \omega) = FFT[\text{Shot}(\vec{x}, t)](\omega) \). Equation (13) can be written:

\[ p_D = \frac{1}{2i} \left( \frac{\omega^2}{v^2} + \frac{\partial^2}{\partial x^2} \right)^{-\frac{1}{2}} q. \] (15)

We then take a Taylor expansion of \( \Lambda^{-1} \) by considering \( k_x \) small compared to \( \frac{\omega}{v} \):

\[ p_D \simeq \frac{iv}{2\omega} \left( -1 + \frac{v^2}{2\omega^2} \frac{\partial^2}{\partial x^2} \right) q. \] (16)

Explicit finite differences are used to obtain a numerical solution:

\[ p^x_D = \frac{iv}{2\omega} \left[ -q^x + \frac{v^2}{2\omega^2} \frac{q^{x-1} - 2q^x + q^{x+1}}{\Delta x^2} \right] \] (17)

\[ = \frac{iv}{2\omega} \left[ \Delta_3 q^{x-1} - (1 + 2\Delta_3) q^x + \Delta_3 q^{x+1} \right], \] (18)

where \( \Delta_3 = \frac{1}{2} \left( \frac{v}{\omega \Delta x} \right)^2 \).

**Mixed domain**

In a homogeneous media, \(-i \Lambda = k_z = k_{z0}\). Hence the \( \Lambda^{-1} \) correction is simply a division by \( k_{z0} \) of the source wavefield at the surface. If a split-step formulation of \( k_z = k_{z0} + \omega \left( \frac{1}{v} - \frac{1}{v_0} \right) \) is considered, then the corrections must be carried out both in the space and wavenumber domain. One solution consists of doing the following expansion:

\[ \frac{1}{k_z} = \frac{1}{k_{z0} + \omega \left( \frac{1}{v} - \frac{1}{v_0} \right)} \simeq \frac{1}{k_{z0}} \left[ 1 - \frac{\omega}{k_{z0}} \left( \frac{1}{v} - \frac{1}{v_0} \right) \right]. \] (19)

It is now possible to perform the correction in a few stages in the space and wavenumber domain.
IMPLEMENTING THE $\Gamma$ OPERATOR

Finite differences

Each of Equations (10) or (11) can be split in two parts:
\[
\frac{\partial P}{\partial z} = \pm ik_z P
\]
(20)
\[
\frac{\partial P}{\partial z} = \frac{1}{2v} \frac{\partial v}{\partial z} \left( \frac{\partial^2}{\partial z^2} + \frac{v_z^2}{\omega^2} \frac{\partial^2}{\partial x^2} \right) P.
\]
(21)

Equation (20) can be written in the general form of the 45° equation (Claerbout, 1999) in which $\alpha$ and $\beta$ are two coefficients to be set:
\[
\frac{\partial P}{\partial z} = \pm \frac{i\omega}{v} \frac{\partial^2 P}{\partial z^2} + \frac{\beta v_z^2}{\omega^2} \frac{\partial^2 P}{\partial x^2}.
\]
(22)

Equation (21) resembles (22) enough for us to use the same scheme to solve it. We present some details of the implementation in Appendix A.

Mixed domain

One solution of (10) is:
\[
\begin{align*}
P'_z &= P_z e^{-\Gamma \Delta z} \\
P_{z+\Delta z} &= P'_z e^{+\Lambda \Delta z}.
\end{align*}
\]
(23)
(24)

We implemented (24) with the split-step method (Stoffa et al., 1990). To implement (23) we take a Taylor expansion of the $\Gamma$ term:
\[
\Gamma = \frac{v_z}{2v} \left[ 1 - \left( \frac{\frac{\partial v}{\partial z}}{\omega} \right)^2 \right] \approx \frac{v_z}{2v} \left[ 1 + \left( \frac{\frac{\partial v}{\partial z}}{\omega} \right)^2 + \left( \frac{\frac{\partial v}{\partial z}}{\omega} \right)^4 \right].
\]
(25)

Equation (23) becomes:
\[
P'_z = P_z e^{-\frac{v_z \Delta z v}{2v^2}} e^{-\frac{v_z \Delta z \frac{k_x^2}{2\omega^2}}{2\omega^2}} e^{-\frac{v_z \Delta z \frac{k_x^4}{2\omega^4}}{2\omega^4}}.
\]
(26)

The last exponentials cannot be computed in a single step because the velocity terms and the horizontal wavenumber do not belong to the same domain. One solution consists of expanding the exponentials in a Taylor series and keeping only the terms of order smaller or equal to 4 in $k_x$:
\[
e^{-\frac{v_z \Delta z \frac{k_x^2}{2\omega^2}}{2\omega^2}} e^{-\frac{v_z \Delta z \frac{k_x^4}{2\omega^4}}{2\omega^4}} \approx 1 - \frac{v_z \Delta z v}{2} \left( \frac{k_x}{\omega} \right)^2 + \frac{v_z \Delta z v^2}{2} \left( \frac{v_z \Delta z v}{4} - v \right) \left( \frac{k_x}{\omega} \right)^4.
\]
(27)

The $\Gamma$ correction is:
\[
P'_z = P_z e^{-\frac{v_z \Delta z}{2v}} \left[ 1 - \frac{v_z \Delta z v}{2} \left( \frac{k_x}{\omega} \right)^2 + \frac{v_z \Delta z v^2}{2} \left( \frac{v_z \Delta z v}{4} - v \right) \left( \frac{k_x}{\omega} \right)^4 \right].
\]
(28)

We give an algorithm of the mixed-domain implementation in Appendix B.
RESULTS

To test the effect of the boundary condition correction operator we used a constant velocity (1000\(m/s\)) model, in which the \(\Gamma\) correction is zero. Because the correction only applies to the source wavefield, we did not go through the imaging condition. We generated a shot at the surface, downward propagated it until \(z = 375m\), and then picked the maximum amplitudes at each \(x\) location at that depth. Figure 1 shows that the operator \(\frac{1}{\Lambda} - 1\) brings the amplitudes much closer to the analytically computed curve both for mixed domain and for finite difference. In the case of the mixed-domain method (top and middle panels), the amplitudes are practically as good as those obtained with two-way wave equation modeling. In the case of finite difference (bottom panel), artifacts interfere constructively and destructively with the true events and lead to amplitude variations along the hyperbola. However, even with artifacts, the corrected amplitudes are closer to the analytical ones than the uncorrected ones.

The \(\Gamma\) correction was tested the same way: by comparing the amplitudes from a shot, propagated until \(z = 1125m\) in a \(v = 1000m/s + 2z\) medium, for which amplitude curves could be computed analytically (Appendix C). The most meaningful comparison is with the WKBJ amplitude correction (Clayton and Stolt, 1981):

\[
\text{WKBJ Correction Factor} = \sqrt{\frac{k_z(z = z_{max})}{k_z(z = 0)}} = \left(\frac{\frac{\omega^2}{v_{z=0}^2} - k_x^2}{\frac{\omega^2}{v_{z=0}^2} - \frac{k_x^2}{k_z(z = 0)}}\right)^{\frac{1}{4}}. \tag{29}
\]

In \(v(z)\) the \(\Gamma\) correction has the same effect as the WKBJ correction, but unlike it, it can also be applied in a \(v(x, z)\) medium. The differences at large angles between the \(\Gamma\) and the WKBJ result can be attributed to the truncations in the Taylor expansions in the \(\Gamma\) term.

We then downward continued a shot until \(z = 2000m\) through the velocity model shown in Figure 3. Downward continuation was performed with both split-step (left panels) and finite differences (right panels), without applying any correction (upper panels), with a \(\Lambda^{-1}\) boundary condition only (middle panels) and with both \(\Lambda^{-1}\) and \(\Gamma\) corrections (lower panels). The corrections result in phase changes and in a much more uniform repartition of amplitudes.

CONCLUSIONS

We have shown ways of implementing the amplitude corrections proposed by Zhang (1993) both by mixed-domain and by finite-difference methods. We applied them to media of constant velocity, constant vertical velocity gradient, and of laterally varying velocity. In constant velocity, and especially for split-step modeling, the boundary condition correction makes amplitudes comparable to the analytical ones, or to those obtained by two-way wave equation modeling. In the constant vertical velocity gradient medium, the \(\Gamma\) correction has the same effect as the WKBJ correction, but unlike it, it can also be applied in a \(v(x, z)\) medium. Future work will include application of the same procedure to other laterally varying velocity models.
Figure 1: Amplitudes computed in the following ways: **Top panel** - Curve 0: analytically; Curve 1: mixed domain with $\Lambda^{-1}$ correction; Curve 2: two-way wave equation modeling; Curve 3: mixed domain without $\Lambda^{-1}$ correction. **Middle panel** - normalized to the analytical curve: Curve 0: analytically; Curve 1: mixed domain with $\Lambda^{-1}$ correction; Curve 2: two-way wave equation modeling. **Bottom panel** - Curve 0: analytically; Curve 1: finite-difference with $\Lambda^{-1}$ correction; Curve 2: finite-difference without $\Lambda^{-1}$ correction.
Figure 2: Amplitudes of a shot propagated by split-step, with the following corrections: Curve 0: $\Lambda^{-1}$ correction only; Curve 1: Both $\Lambda^{-1}$ and $\Gamma$ corrections; Curve 2: $\Lambda^{-1}$ and WKBJ corrections

Figure 3: $V(x,z)$ velocity model

and to real data and validation of $v(x,z)$ results by comparison with results of ray-tracing with amplitudes.

ACKNOWLEDGMENTS

We thank Guojian Shan for his advice regarding finite differencing and for the two-way wave equation modeling program used for curve 2 in the top and middle panels of Figure 1, to Norm Bleinstein for bringing this topic to our attention during a recent visit to SEP, and to Bill Symes for the velocity model shown in Figure 3.

REFERENCES


Figure 4: Wavefields of a shot propagated through the velocity model in Figure 3. **Left** panels: split-step; **Right** panels: finite difference; **Upper** panels: no amplitude correction; **Middle** panels: boundary condition correction ($\Lambda^{-1}$) correction only; **Lower** panels: both boundary condition ($\Lambda^{-1}$) and $\Gamma$ correction.


APPENDIX A

Solving equation 20  We define \( n = \frac{\omega}{v} \). Equation 20 can be written:

\[
\partial_z = \pm i \frac{\alpha n \partial_{xx}}{1 + \beta n^2 \partial_{xx}}, \quad \text{(A-1)}
\]

or

\[
\partial_z + \beta n^2 \partial_{xx} \mp i \alpha n \partial_{xx} = 0. \quad \text{(A-2)}
\]

We apply the finite difference method:

\[
\partial_z p = \frac{1}{\Delta z} (p_{z+1}^x - p_z^x) \quad \text{(A-3)}
\]

\[
\partial_{xx} p = \frac{1}{\Delta x^2} (p_{z+1}^x - p_z^x) + \frac{-1}{2 \Delta x^2} (p_{z+1}^x - p_z^x) + \frac{1}{\Delta x^2} (p_{z+1}^x - p_z^x), \quad \text{(A-4)}
\]

\[
\partial_{xx} p = \frac{1}{\Delta x^2} (p_{z+1}^x - p_z^x) + \frac{-1}{2 \Delta x^2} (p_{z+1}^x - p_z^x) + \frac{1}{\Delta x^2} (p_{z+1}^x - p_z^x). \quad \text{(A-5)}
\]

After some little rearrangements, equation (A-2) can be written:

\[
\Delta p_{z+1}^{x-1} + (1 - 2\Delta) p_z^x + \Delta_{z} p_{z+1}^x = \Delta_{z} p_z^{x-1} + (1 - 2\Delta) p_z^x + \Delta_z p_z^x. \quad \text{(A-6)}
\]

where

\[
\Delta = \frac{\beta n^2}{\Delta x^2} \mp i \frac{\alpha n \Delta z}{2 \Delta x^2} \quad \text{(A-7)}
\]

For downward continuing the receiver (upgoing) wavefield, we choose the positive sign in the relation above; for downward continuation of the source (downgoing) wavefield we take the sign in \( \Delta \) to be negative. To solve the 15° equation, \((\alpha, \beta) = (0.5, 0)\). For the 45° equation, \((\alpha, \beta) = (0.5, 0.25)\), and for the 65° equation, \((\alpha, \beta) = (0.478242060, 0.376369527)\) (Lee and Suh, 1985). In our implementation, we have used the 65° equation with the “\( \frac{1}{6} \) trick” (Claerbout, 1985) for improving the accuracy of the second derivative. This changes (A-2) and (A-7) into

\[
\partial_z + \left( \beta n^2 + \frac{\Delta x^2}{s} \right) \partial_{xx} \mp i \alpha n \partial_{xx} = 0 \quad \text{(A-8)}
\]

\[
\Delta = \left( \frac{\beta n^2}{\Delta x^2} + \frac{1}{s} \right) \mp i \frac{\alpha n \Delta z}{2 \Delta x^2} \quad \text{(A-9)}
\]

where we took \( s = 8.13 \) (Fomel and Claerbout, 1997). Equation (A-6) with the new \( \Delta \) gives a tridiagonal system for each frequency and depth.

Solving equation 21

Equation (21) can be written:

\[
\partial_z = \frac{v_z}{2v} \frac{1}{1 + n^2 \partial_{xx}}. \quad \text{(A-10)}
\]
As for Equation (A-2), we use a Crank-Nicolson scheme for the second derivative in \( x \). Equation (A-10) becomes:

\[
\Delta_1 p_{z+1}^{x-1} + (1 - 2\Delta_1 + \Delta_2) p_{z+1}^x + \Delta_1 p_{z+1}^x = \Delta_1 p_{z}^{x-1} + (1 - 2\Delta_1 - \Delta_2) p_{z}^x + \Delta_1 p_{z}^x \quad (A-11)
\]

where

\[
\begin{align*}
\Delta_1 &= \frac{n^2}{\Delta x^2}, \\
\Delta_2 &= -\frac{v_z \Delta z}{4v}. \quad (A-13)
\end{align*}
\]

Equation (A-11) is solved the same way as (A-6).

**APPENDIX B**

We provide here the algorithm of our mixed-domain implementation of both \( \Lambda^{-1} \) and \( \Gamma \) corrections. The notations refer to a single spatial axis, but they can be easily extended to two axes. The \( \Lambda^{-1} \) correction is only applied to the source wavefield, and is only applied once, at the surface, not for each depth step. The correction is \( P_{z=0} = -\frac{1}{2ik_z} Q \). Because it generates strong wraparound that translates as noise in the final image, it is recommended to pad the data with zeros, apply the correction, then window it to a smaller size for downward continuation.

For each frequency, to downward continue one step with corrected amplitudes (\( \Gamma \) operator) we Fourier transform to the wavenumber domain:

- \( P_z(k_x) \leftarrow P_z(x) \).

In the wavenumber domain, we perform the following operations:

- Phase shift (part of split-step): \( P_{z+\Delta z}^0 \leftarrow e^{ik_0\Delta z} P_{z+\Delta z} \)
  - where \( k_0 = \sqrt{\omega^2/v_0^2 - k^2_x} \), in which \( v_0 \) is the reference velocity at that depth.

- First order amplitude correction (part of \( \Gamma \)): \( P_{z+\Delta z}^1 \leftarrow \left( \frac{k_x}{\omega} \right)^2 P_{z+\Delta z}^0 \)

- Second order amplitude correction (part of \( \Gamma \)): \( P_{z+\Delta z}^2 \leftarrow \left( \frac{k_x}{\omega} \right)^4 P_{z+\Delta z}^0 \)

Three separate Fourier transforms are needed to transform back to the spatial domain:

- \( P_{z+\Delta z}^0(x) \leftarrow P_z(k_x) \),
- \( P_{z+\Delta z}^1(x) \leftarrow P_z^1(k_x) \),
- \( P_{z+\Delta z}^2(x) \leftarrow P_z^2(k_x) \).
In the spatial domain we perform:

- First order amplitude correction (part of $\Gamma_1$):\[ P_{z+\Delta z}^1 \leftarrow -\frac{v_z \Delta z}{2} P_{z+\Delta z}^1, \]

- Second order amplitude correction (part of $\Gamma_1$):\[ P_{z+\Delta z}^2 \leftarrow \frac{v_z \Delta z v^2}{2} \left( \frac{v_z \Delta z}{4} - v \right) P_{z+\Delta z}^1, \]

- Summation of all terms: \[ P_{z+\Delta z} \leftarrow e^{-\frac{v_z \Delta z}{2\nu}} \left( P_{z+\Delta z}^0 + P_{z+\Delta z}^1 + P_{z+\Delta z}^2 \right), \]

- Split step: \[ P_{z+\Delta z}(x) \leftarrow e^{i \omega \left( \frac{1}{2} - \frac{1}{v_0} \right) \Delta z} P_{z+\Delta z}(x). \]

The $\Gamma$ correction for the receiver wavefield is computed the same way. Once both wavefields have been computed, the imaging condition (9) is applied for each depth.

In laterally varying velocity the terms in the spatial wavenumber domain that contain $k_x$ do not commute with terms that contain functions of $x$, namely $v$ and $v_z$. We took care to apply the terms in the right order inside the $\Gamma$ operator. Outside $\Gamma$, things change. Strictly mathematically speaking, $\Gamma$ does not commute with the split-step, and the right order of applying the operators remains to be studied. As in the case of split-step (for which the two terms that compose it do not commute with each other either), the differences in output may not warrant the cost of the extra Fourier transforms.

**APPENDIX C**

At depth $z$ and at each point $x$ we compute:

\[ I_x = \frac{A_0}{A_x} I_0, \quad (C-1) \]

where $A_0$ and $A_x$ are the areas of the unitary wavefront and a particular wavefront respectively. For spherical wavefronts,

In 2-D: $A_0 = 2\pi R_0 = 2\pi$, $A_x = 2\pi R$ hence $I_x = \frac{1}{R} I_0$.

In 3-D: $A_0 = 4\pi R_0^2 = 4\pi$, $A_x = 4\pi R^2$ hence $I_x = \frac{1}{R^2} I_0$.

To be able to compute the perturbed amplitude $I_x$ from the unitary amplitude $I_0$, we need to compute $R$, the radius of the spherical wavefront. Two cases where the wavefronts are spherical are media of constant velocity and constant vertical velocity gradient.

**Constant velocity media** In this simple case, the length of the ray is:

\[ R = \sqrt{x^2 + z^2}. \quad (C-2) \]
**Constant vertical velocity gradient media**  Let us consider a velocity field of the form \( v = v_0 + az \), where \( a \) is a constant. Both the ray and the wavefront are circles as illustrated in Figure C-1. If the ray from a source located at the origin surfaces at \((x_{out}, 0)\), then according to Equation (15) in Slotnick (1936), the depth of penetration of the ray is:

\[
z_0 = \frac{v_0}{a} \left[ \sqrt{1 + \left( \frac{ax_{out}}{2v_0} \right)^2} - 1 \right]. \tag{C-3}
\]

The ray takes the time \( 2t \) to travel from origin to \((x_{out}, 0)\), and only half of that \(-t\) to get to the maximum depth point \( (\frac{x_{out}}{2}, z_0) \). According to Equation (14) in Slotnick (1936),

\[
t = \frac{1}{a} \sinh^{-1} \left( \frac{ax_{out}}{2v_0} \right). \tag{C-4}
\]

Because wavefronts are orthogonal to rays, and because the ray is horizontal at the location where it reaches penetration depth, the equation of the only wavefront that passes through \( (\frac{x_{out}}{2}, z_0) \) is:

\[
x^2 + (z - z_0)^2 = \left( \frac{x_{out}}{2} \right)^2. \tag{C-5}
\]

By plugging (C-3) and (C-4) into (C-5) we get the general equation of a wavefront:

\[
x^2 + \left[ z - \frac{v_0}{a} \left( \sqrt{1 + \sinh^2(at)} - 1 \right) \right]^2 = \left[ \frac{v_0}{a} \sinh(at) \right]^2 = R^2. \tag{C-6}
\]
There is a single ray that departs from the origin and arrives at the point \((x, z)\). Its ray parameter is \(p\) and the time it takes to get to \((x, z)\) is the solution of the integral in Equation (8) of Slotnick (1936) for the particular case of \(v = v_0 + az\):

\[
t = \frac{1}{a} \log F, \quad (C-7)
\]

where we use the notation

\[
F = \left(1 + \frac{az}{v_0}\right) \frac{1 + \sqrt{1 - p^2v_0^2}}{1 + \sqrt{1 - p^2(v_0 + az)^2}}. \quad (C-8)
\]

We can write \(p\) as a function of \(x\) and \(z\) by writing Equation (12) in Slotnick (1936) as:

\[
p^2 = \frac{1}{v_0^2 + \left[\frac{4(x^2 + z^2) + azv_0}{x}\right]^2}. \quad (C-9)
\]

By combining (C-6) and (C-7) we can write \(R\) as a function of \(p\) instead of \(t\):

\[
R = \frac{v_0}{2a} \left( F - \frac{1}{F} \right). \quad (C-10)
\]

To compute the wavefront radius \(R\) as a function only of \(x, z, v_0\) and \(a\), we have to plug (C-9) into (C-8), and then (C-8) into (C-10).
Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging

Biondo Biondi and William Symes

ABSTRACT

We analyze the kinematic properties of offset-domain Common Image Gathers (CIGs) and Angle-Domain CIGs (ADCIGs) computed by wavefield-continuation migration. Our results are valid regardless of whether the CIGs were obtained by using the correct migration velocity. They thus can be used as a theoretical basis for developing Migration Velocity Analysis (MVA) methods that exploit the velocity information contained in ADCIGs.

We demonstrate that in an ADCIG cube the image point lies on the normal to the apparent reflector dip, passing through the point where the source ray intersects the receiver ray. Starting from this geometric result, we derive an analytical expression for the expected movements of the image points in ADCIGs as functions of the traveltime perturbation caused by velocity errors. By applying this analytical result and assuming stationary raypaths, we then derive two expressions for the Residual Moveout (RMO) function in ADCIGs. We verify our theoretical results and test the accuracy of the proposed RMO functions by analyzing the migration results of a synthetic data set with a wide range of reflector dips.

Our kinematic analysis leads also to the development of a new method for computing ADCIGs when significant geological dips cause strong artifacts in the ADCIGs computed by conventional methods. The proposed method is based on the computation of offset-domain CIGs along the vertical-offset axis (VOCIGs) and on the “optimal” combination of these new CIGs with conventional CIGs. We demonstrate the need for and the advantages of the proposed method on a real data set acquired in the North Sea.

INTRODUCTION

With wavefield-continuation migration methods being used routinely for imaging project in complex areas, the ability to perform Migration Velocity Analysis (MVA) starting from the results of wavefield-continuation migration is becoming essential to advanced seismic imaging. As for Kirchhoff imaging, MVA for wavefield-continuation imaging is mostly based on the information provided by the analysis of Common Image Gather (CIGs). Most of the current MVA methods start from Angle-Domain CIGs (ADCIGs) (Biondi and Sava, 1999; Clapp

1email: biondo@sep.stanford.edu
and Biondi, 2000; Mosher et al., 2001; Liu et al., 2001), though the use of more conventional surface-offset-domain CIGs is also being evaluated (Stork et al., 2002).

Both kinematic and amplitude properties (de Bruin et al., 1990; Wapenaar et al., 1999; Sava et al., 2001; de Hoop et al., 2002) have been analyzed in the literature for ADCIGs obtained when the migration velocity is accurate. On the contrary, the properties of the ADCIGs obtained when the migration velocity is inaccurate have been only qualitatively discussed in the literature. This lack of quantitative understanding may lead to errors when performing MVA from ADCIGs. In this paper, we analyze the kinematic properties of ADCIGs under general conditions (accurate or inaccurate velocity). If the migration velocity is inaccurate, our analysis requires only a smooth migration velocity function in the neighborhood of the imaging point. We discuss this condition more extensively in the first section. The application of the insights provided by our analysis may substantially improve the results of the following three procedures: a) measurement of velocity errors from ADCIGs by residual moveout (RMO) analysis, b) inversion of RMO measurements into velocity updates, and c) computation of ADCIGs in the presence of complex geologic structure.

Our analysis demonstrates that in an ADCIG cube the image point lies on the normal to the apparent reflector dip passing through the point where the source ray intersects the receiver ray. We exploit this result to define an analytical expression for the expected movements of the image points in ADCIGs as a function of the traveltime perturbation caused by velocity errors. This leads us to the definition of two alternative residual moveout functions that can be applied when measuring velocity errors from migrated images. We test the accuracy of these alternatives and discuss their relative advantages and disadvantages. Furthermore, the availability of a quantitative expression for the expected movements of the image points is crucial when inverting those movements into velocity corrections by either simple vertical updating or sophisticated tomographic methods. Therefore, our results ought to be incorporated in velocity updating methods.

Our theoretical result also implies that ADCIGs are immune, at least at first order, from the distortions caused by image-point dispersal. Image-point dispersal occurs when migration velocity errors cause events from the same segment of a dipping reflector to be imaged at different locations (Etgen, 1990). This inconsistency creates substantial problems when using dipping reflections for velocity updating; its absence makes ADCIGs even more attractive for MVA.

The computation of ADCIGs is based on a decomposition (usually performed by slant-stacks) of the wavefield either before imaging (Mosher et al., 1997; Prucha et al., 1999; Xie and Wu, 2002), or after imaging (Sava and Fomel, 2002; Rickett and Sava, 2002; Biondi and Shan, 2002). In either case, the slant stack transformation is usually applied along the horizontal subsurface-offset axis. However, when the geologic dips are steep, this “conventional” way of computing CIGs does not produce useful gathers, even if it is kinematically valid for geologic dips milder than 90 degrees. As the geologic dips increase, the horizontal-offset CIGs (HOCIGs) degenerate, and their focusing around zero offset blurs. This limitation of HOCIGs can be sidestepped by computing offset-domain CIGs along the vertical subsurface-offset axis (VOCIGs) (Biondi and Shan, 2002). Although neither set of offset-domain gathers (HOCIG
or VOCIG) provides useful information for the whole range of geologic dips, an appropriate combination of the two sets does. Our analysis of the kinematic properties of ADCIGs suggests a simple and effective method for combining a HOCIG cube with a VOCIG cube to create an ADCIG cube that is immune to artifacts in the presence of arbitrary geologic dips.

The plan of attack for covering the broad, but interrelated, set of issues that are relevant to the use of ADCIGs for MVA is the following. We start by briefly reviewing the methodology for computing offset-domain and angle-domain CIGs by wavefield-continuation migration. The second section analyzes the kinematic properties of CIGs and ADCIGs, and contains the main theoretical development of the paper. The third section exploits the theoretical results to define a robust algorithm to compute ADCIGs in the presence of geological structure and illustrates its advantages with a real-data example. The fourth section verifies the theoretical analysis by using it to predict reflector movements in the migrated images of a synthetic data set. Finally, the fifth section derives two expressions for the RMO function to be applied for measuring velocity errors from migrated images.

**COMPUTATION OF COMMON IMAGE GATHERS BY WAVEFIELD CONTINUATION**

In this section we briefly revisit the method for computing Common Image Gathers (CIG) by wavefield-continuation migration. The following development assumes that both the source wavefield and the receiver wavefield have been numerically propagated into the subsurface. The analytical expressions represent wavefields in the time domain, and thus they appear to implicitly assume that the wavefields have been propagated in the time domain. However, all the considerations and results that follow are independent of the specific numerical method that was used for propagating the wavefields. They are obviously valid for reverse-time migration when the wavefields are propagated in the time domain (Whitmore, 1983; Baysal et al., 1983; Etgen, 1986; Biondi and Shan, 2002). They are also valid when the wavefields are propagated by downward continuation in the frequency domain, if there are no overturned events. The results presented in this paper are valid even when source-receiver migration is used instead of shot-profile migration, if the conditions are satisfied for these two apparently dissimilar methods to be equivalent (Biondi, 2003).

The conventional imaging condition for shot-profile migration is based on the crosscorrelation in time of the source wavefield \( S \) with the receiver wavefield \( R \). The equivalent of the stacked image is the average over sources \( s \) of the zero lag of this crosscorrelation; that is:

\[
I(z, x) = \sum_s \sum_t S_s(t, z, x) R_s(t, z, x),
\]

where \( z \) and \( x \) are respectively depth and the horizontal axes, and \( t \) is time. The result of this imaging condition is equivalent to stacking over offsets with Kirchhoff migration.

The imaging condition expressed in equation (1) has the substantial disadvantage of not providing prestack information that can be used for either velocity updates or amplitude analysis. Equation (1) can be generalized (de Bruin et al., 1990; Rickett and Sava, 2002; Biondi
Figure 1: Geometry of an ADCIG for a single event migrated with the wrong (low in this case) velocity. The propagation direction of the source ray forms the angle $\beta$ with the vertical, and the propagation direction of the receiver ray forms the angle $\delta$ with the vertical; $\gamma$ is the apparent aperture angle, and $\alpha$ is the apparent reflector dip. The source ray and the receiver ray cross at $\bar{I}$. Notice that in this figure $\beta, \delta$ and $\alpha$ are positive, but $\gamma$ is negative.

and Shan, 2002) by crosscorrelating the wavefields shifted horizontally with respect to each other. The prestack image becomes a function of the horizontal relative shift, which has the physical meaning of a subsurface half offset $(x_h)$. It can be computed as

$$I(z, x, x_h) = \sum_s \sum_t S_s(t, z, x - x_h) R_s(t, z, x + x_h).$$  \hspace{1cm} (2)

A section of the image cube taken at constant horizontal location $x$ is a Horizontal Offset Common Image Gather, or HOCIG. The whole image cube can be seen as a collection of HOCIGs.

Sava and Fomel (2002) presented a simple method for transforming HOCIGs into ADCIGs by a slant stack transformation applied independently to each HOCIG (Schultz and Claerbout, 1978):

$$I_{\gamma}(z, x, \gamma) = \text{SlantStack}[I(z, x, x_h)];$$  \hspace{1cm} (3)

where $\gamma$ is the aperture angle of the reflection, as shown in Figure 1. This transformation from HOCIG to ADCIG is based on the following relationship between the aperture angle and the slope, $\partial z / \partial x_h$, measured in image space:

$$\left. \frac{\partial z}{\partial x_h} \right|_{t,x} = \tan \gamma = -\frac{k_{x_h}}{k_z};$$  \hspace{1cm} (4)

where $k_{x_h}$ and $k_z$ are respectively the half-offset wavenumber and the vertical wavenumber. The relationship between $\tan \gamma$ and the wavenumbers also suggests that the transformation to ADCIGs can be accomplished in the Fourier domain by a simple radial-trace transform (Sava and Fomel, 2002).

Sava and Fomel (2002) demonstrated the validity of equation (4) based only on Snell’s law and on the geometric relationships between the propagation directions of the source ray (determined by $\beta$ in Figure 1) and receiver ray (determined by $\delta$ in Figure 1). Its validity
is thus independent of the focusing of the reflected energy at zero offset; that is, it is valid regardless of whether the image point coincides with the intersection of the two rays (marked as $\bar{I}$ in Figure 1). In other words, it is independent of whether the correct migration velocity is used. The only assumption about the migration velocity is that the velocity at the imaging depth is locally the same along the source ray and the receiver ray. This condition is obviously fulfilled when the reflected energy focuses at zero offset, but it is, at least approximately, fulfilled in most practical situations of interest. In most practical cases we can assume that the migration velocity function is smooth in a neighborhood of the imaging point, and thus that the velocity at the end point of the source ray is approximately the same as the velocity at the end point of the receiver ray. The only exception of practical importance is when the reflection is caused by a high-contrast interface, such as a salt-sediment interface. In these cases, our results must be applied with particular care. When the migration velocity is correct, $\alpha$ and $\gamma$ are respectively the true reflector dip and the true aperture angle; otherwise they are the apparent dip and the apparent aperture angle. In Figure 1, the box around the imaging point signifies the local nature of the geometric relationships relevant to our discussion; it emphasizes that these relationships depend only on the local velocity function.

When the velocity is correct, the image point obviously coincides with the crossing point of the two rays $\bar{I}$. However, the position of the image point when the velocity is not correct has been left undefined by previous analyses (Prucha et al., 1999; Sava and Fomel, 2002). In this paper, we demonstrate the important result that in an ADCIGs, when the migration velocity is incorrect, the image point lies along the direction normal to the apparent geological dip. We identify this normal direction with the unit vector $\mathbf{n}$ that we define as oriented in the direction of increasing traveltimes for the rays (see Figure 1).

Notice that the geometric arguments presented in this paper are based on the assumption that the source and receiver rays cross, even when the data were migrated with the wrong velocity. This assumption is valid in 2-D except in degenerate cases of marginal practical interest (e.g. diverging rays). In 3-D, this assumption is more easily violated, because the two rays are not always coplanar. This discrepancy between 2-D and 3-D geometries makes the generalization to 3-D of the results presented in this paper less than trivial. Therefore, we consider the 3-D generalization beyond the scope of this paper.

As will be discussed in the following and exemplified by the real-data example in Figure 6a, the HOCIGs, and consequently the ADCIGs computed from the HOCIGs (Figure 7a), have problems when the reflectors are steeply dipping. At the limit, the HOCIGs become useless when imaging almost vertical reflectors using either overturned events or prismatic reflections. To create useful ADCIGs in these situations we introduce a new kind of CIGs (Biondi and Shan, 2002). This new kind of CIG is computed by introducing a vertical half offset ($z_h$) into equation (1) to obtain:

$$I(z, x, z_h) = \sum_s \sum_t S_s(t, z - z_h, x) R_s(t, z + z_h, x).$$

A section of the image cube computed by equation (5) taken at constant depth $z$ is a Vertical Offset Common Image Gathers, or VOCIG.

As for the HOCIGs, the VOCIGs can be transformed into an ADCIG by applying a slant
stack transformation to each individual VOCIG; that is:

\[ I_{\gamma z}(z, x, \gamma) = \text{SlantStack}[I(z, x, z_h)]. \]

This transformation is based on the following relationship between the aperture angle and the slope \( \frac{\partial x}{\partial z_h} \) measured in image space:

\[ -\left| \frac{\partial x}{\partial z_h} \right|_{t, z} = \tan \gamma = \frac{k_{z_h}}{k_x}. \]

Equation (7) is analogous to equation (4), and its validity can be trivially demonstrated from equation (4) by a simple axes rotation. However, notice the sign differences between equation (7) and equation (4) caused by the conventions defined in Figure 1.

Notice that our notation distinguishes the result of the two transformations to ADCIG \( (I_{\gamma x} \text{ and } I_{\gamma z}) \), because they are different objects even though they are images defined in the same domain \((z, x, \gamma)\). One of the main results of this paper is the definition of the relationship between \( I_{\gamma x} \text{ and } I_{\gamma z} \), and the derivation of a robust algorithm to “optimally” merge the two sets of ADCIGs. To achieve this goal we will first analyze the kinematic properties of HOCIGs and VOCIGs.

**KINEMATIC PROPERTIES OF COMMON IMAGE GATHERS**

In this section we analyze the kinematic properties of CIGs, with particular emphasis on the case when velocity errors prevent the image from focusing at zero offset, causing the reflected energy to be imaged over a range of offsets. We will start by analyzing the kinematics of offset-domain CIGs.

To analyze the kinematic properties of HOCIGs and VOCIGs, it is useful to observe that they are just particular cases of offset-domain gathers. In general, the offset can be oriented along any arbitrary direction. In particular, the offset direction aligned with the apparent geological dip of the imaged event has unique properties. We will refer to this offset as the *geological-dip offset*, and the corresponding CIGs as Geological Offset CIGs, or GOCIGs.

Figure 2 illustrates the geometry of the different kinds of offset-domain CIGs for a single event. In this sketch, the migration velocity is assumed to be lower than the true velocity, and thus the reflections are imaged too shallow and above the point where the source ray crosses the receiver ray \( (\bar{I}) \). The line passing through \( \bar{I} \), and bisecting the angle formed by the source and receiver ray, is oriented at an angle \( \alpha \) with respect to the vertical direction. The angle \( \alpha \) is the apparent geological dip of the event after imaging. Half of the angle formed between the source and receiver ray is the apparent aperture angle \( \gamma \).

When HOCIGs are computed, the end point of the source ray \( (S_{x_h}) \) and the end point of the receiver ray \( (R_{x_h}) \) are at the same depth. The imaging point \( I_{x_h} \) is midway between \( S_{x_h} \) and \( R_{x_h} \), and the imaging half offset is \( x_h = R_{x_h} - I_{x_h} \). Similarly, when VOCIGs are computed, the end point of the source ray \( (S_{z_h}) \) and the end point of the receiver ray \( (R_{z_h}) \) are at the same horizontal location. The imaging point \( I_{z_h} \) is midway between \( S_{z_h} \) and \( R_{z_h} \), and the imaging
half offset is \( z_h = R_{zh} - I_{zh} \). When the offset direction is oriented along the apparent geological dip \( \alpha \) (what we called the geological-dip offset direction), the end point of the source ray is \( S_0 \) and the end point of the receiver ray is \( R_0 \). The imaging point \( I_0 \) is midway between \( S_0 \) and \( R_0 \), and the imaging half offset is \( h_0 = R_0 - I_0 \). Notice that the geological-dip half offset \( h_0 \) is a vector, because it can be oriented arbitrarily with respect to the coordinate axes.

Figure 2 shows that both \( I_{xh} \) and \( I_{zh} \) lie on the line passing through \( S_0, I_0 \) and \( R_0 \). This is an important property of the offset-domain CIGs and is based on a crucial constraint imposed on our geometric construction; that is, the traveltime along the source ray summed with the traveltime along the receiver ray is the same for all the offset directions, and is equal to the recording time of the event. The independence of the total traveltimes from the offset directions is a direct consequence of taking the zero lag of the crosscorrelation in the imaging conditions of equation (2) and (5). This constraint, together with the assumption of locally constant velocity that we discussed above, directly leads to the following equalities:

\[
|S_{xh} - S_0| = |R_{xh} - R_0|, \quad \text{and} \quad |S_{zh} - S_0| = |R_{zh} - R_0|,
\]

which in turn are at the basis of the collinearity of \( I_0, I_{xh} \) and \( I_{zh} \).

The offsets along the different directions are linked by the following simple relationship, which can be readily derived by trigonometry applied to Figure 2:

\[
x_{h} = \frac{\tilde{h}_0}{\cos \alpha},
\]

\[
z_{h} = -\frac{\tilde{h}_0}{\sin \alpha},
\]

where \( \tilde{h}_0 = n \times h_0 \). Notice that the definition of \( \tilde{h}_0 \) is such that its sign depends on whether \( I_0 \) is before or beyond \( \bar{I} \), and that for flat events \( (\alpha = 0) \) we have \( \tilde{h}_0 = x_h \).

Although \( I_{xh} \) and \( I_{zh} \) are both collinear with \( I_0 \), they are shifted with respect to each other and with respect to \( I_0 \). The shifts of the imaging points \( I_{xh} \) and \( I_{zh} \) with respect to \( I_0 \) can be easily expressed in terms of the offset \( h_0 \) and the angles \( \alpha \) and \( \gamma \) as follows:

\[
\Delta I_{xh} = (I_{xh} - I_0) = h_0 \tan \gamma \tan \alpha,
\]

\[
\Delta I_{zh} = (I_{zh} - I_0) = -h_0 \frac{\tan \gamma}{\tan \alpha}.
\]

The shift between \( I_{xh} \) and \( I_{zh} \) prevents us from constructively averaging HOCIGs with VO-CIGs to create a single set of offset-domain CIGs.

Notice the dependence of \( \Delta I_{xh} \) and \( \Delta I_{zh} \) on the aperture angle \( \gamma \). This dependence causes events with different aperture angles to be imaged at different locations, even if they originated at the same reflecting point in the subsurface. This phenomenon is related to the well known reflector-point dispersal in common midpoint gathers. In this context, this dispersal is a consequence of using a wrong imaging velocity, and we will refer to it as image-point dispersal. We will now discuss how the transformation to ADCIGs overcomes the problems related to the image-point shift and thus removes, at least at first order, the image-point dispersal.
Figure 2: Geometry of the three different kinds of offset-domain (horizontal, vertical and geological-dip) CIG for a single event migrated with the wrong velocity. $I_{xh}$ is the horizontal-offset image point, $I_{zh}$ is the vertical-offset image point, and $I_0$ is the geological-dip offset image point.

Figure 3: Geometry of an angle-domain CIG for a single event migrated with the wrong velocity. The transformation to the angle domain shifts all the offset-domain image points ($I_{xh}, I_{zh}, I_0$) to the same angle-domain image point $I_\gamma$. 
Kinematic properties of ADCIGs

The transformation to the angle domain, as defined by equations (3–4) for HOCIGs and equations (6–7) for VOCIGs, acts on each offset-domain CIG independently. Therefore, when the reflected energy does not focus at zero offset, the transformation to the angle domain shifts the image point along the direction orthogonal to the offset. The horizontal-offset image point \((I_{xh})\) shifts vertically, and the vertical-offset image point \((I_{zh})\) shifts horizontally. We will demonstrate the two following important properties of this normal shift:

I) The normal shift corrects for the effects of the offset direction on the location of the image point; that is, the transformation to the angle domain shifts the image points from different locations in the offset domain \((I_{xh}, I_{zh} \text{ and } I_0)\) to the same location in the angle domain \((I_\gamma)\).

II) The image location in the angle domain \((I_\gamma)\) lies on the normal to the apparent geological dip passing through the crossing point of the source and receiver rays \((\bar{I})\). \(I_\gamma\) is located at the crossing point of the lines passing through \(S_0\) and \(R_0\) and orthogonal to the source ray and receiver ray, respectively. The shift along the normal to the reflector, caused by the transformation to angle domain, is thus equal to:

\[
\Delta n_\gamma = (I_\gamma - I_0) = \tilde{h}_0 \tan \gamma n = \tan^2 \gamma \Delta n_{h0},
\]

where \(\Delta n_{h0} = (\tilde{h}_0 / \tan \gamma) n\) is the normal shift in the geological-dip domain. The total normal shift caused by incomplete focusing at zero offset is thus equal to:

\[
\Delta n_{tot} = (I_\gamma - \bar{I}) = \Delta n_{h0} + \Delta n_\gamma = \Delta n_{h0} (1 + \tan^2 \gamma) = \frac{\Delta n_{h0}}{\cos^2 \gamma}.
\]

Figure 3 illustrates Properties I and II. These properties are far from obvious and their demonstration constitutes one of the main results of this paper. They also have several important consequences; the three results most relevant to migration velocity analysis are:

1. ADCIGs obtained from HOCIGs and VOCIGs can be constructively averaged, in contrast to the original HOCIGs and VOCIGs. We will exploit this property to introduce a robust algorithm for creating a single set of ADCIGs that is insensitive to geological dips, and thus is ready to be analyzed for velocity information.

2. The reflector-point dispersal that negatively affects offset-domain CIGs is corrected in the ADCIGs, at least at first order. If we assume the raypaths to be stationary, for a given reflecting segment the image points for all aperture angles \(\gamma\) share the same apparent dip, and thus they are all aligned along the normal to the apparent reflector dip.

3. From equation (14), invoking Fermat’s principle and applying simple trigonometry, we can also easily derive a relationship between the total normal shift \(\Delta n_{tot}\) and the total traveltime perturbation caused by velocity errors as follows:

\[
\Delta n_{tot} = -\frac{\Delta t}{2S \cos \gamma} n.
\]
where $S$ is the background slowness around the image point and $\Delta t$ is defined as the difference between the perturbed traveltime and the background traveltime. We will exploit this relationship to introduce a simple and accurate expression for measuring residual moveouts from ADCIGs.

**Demonstration of kinematic properties of ADCIGs**

Properties I and II can be demonstrated in several ways. In this paper, we will follow an indirect path that might seem circuitous but will allow us to gather further insights on the properties of ADCIGs.

We first demonstrate Property I by showing that the radial-trace transformations represented by equation (4), and analogously equation (7), are equivalent to a chain of two transformations. The first one is the transformation of the HOCIGs (or VOCIGs) to GOCIGs by a dip-dependent stretching of the offset axis; that is:

$$\tilde{h}_0 = x_h \cos \alpha, \quad \text{or} \quad \tilde{h}_0 = -z_h \sin \alpha; \quad (16)$$

or in the wavenumber domain,

$$k_{h_0} = \frac{k_{x_h}}{\cos \alpha}, \quad \text{or} \quad k_{h_0} = -\frac{k_{z_h}}{\sin \alpha}; \quad (17)$$

where $k_{h_0}$ is the wavenumber associated with $\tilde{h}_0$, and $k_{x_h}$ and $k_{z_h}$ are the wavenumbers associated with $x_h$ and $z_h$.

The second is the transformation of HOCIGs to the angle domain according to the relation

$$\tan \gamma = -\frac{k_{h_0}}{k_n}; \quad (18)$$

where $k_n$ is the wavenumber associated with the direction normal to the reflector. This direction is identified by the line passing through $\bar{I}$ and $I_\gamma$ in Figures 2 and 3.

The transformation of HOCIGs to GOCIGs by equations (16) and (17) follows directly from equations (9) and (10). Because the transformation is a dip-dependent stretching of the offset axis, it shifts energy in the $(z, x)$ plane. Appendix A demonstrates that the amount of shift in the $(z, x)$ plane exactly corrects for the image-point shift characterized by equations (11) and (12).

Appendix B demonstrates the geometrical property that for energy dipping at an angle $\alpha$ in the $(z, x)$ plane, the wavenumber $k_n$ along the normal to the dip is linked to the wavenumbers along $(z, x)$ by the following relationships:

$$k_n = \frac{k_z}{\cos \alpha} = \frac{k_x}{\sin \alpha}. \quad (19)$$

Substituting equations (17) and (19) into equation (18), we obtain equations (4) and (7). The graphical interpretation of this analytical result is immediate. In Figure 3, the transformation...
to GOCIG [equations (17)] moves the imaging point $I_{xh}$ (or $I_{zh}$) to $I_0$, and the transformation to the angle domain [equation (18)] moves $I_0$ to $I_\gamma$. This sequence of two shifts is equivalent to the direct shift from $I_{xh}$ (or $I_{zh}$) to $I_\gamma$ caused by the transformation to the angle domain applied to a HOCIG (or VOCIG).

We just demonstrated that the transformation to ADCIG is independent from which type of offset-domain CIGs we started from (HOCIG, VOCIG, or GOCIG). Consequently, the imaging point $I_\gamma$ must be common to all kinds of ADCIGs. Furthermore, the image point must lie along each of the normals to the offset directions passing through the respective image points. In particular, it must lie along the normal to the apparent geological dip, and at the crossing point of the the vertical line passing through $I_{xh}$ and the horizontal line passing through $I_{zh}$.

Given these constraints, the validity of Property II [equations (13) and (14)] can be easily verified by trigonometry, assuming that the image-point shifts are given by the expressions in equations (9) and (10). However, we will now demonstrate Property II in an alternative way; that is, by analyzing a GOCIG computed from an event with no apparent geological dip ($\alpha = 0$). This analysis provides intuitive understanding of the relation between offset-domain and angle domain CIGs when the migration velocity is incorrect. Furthermore, the analysis of a GOCIG with flat dip is representative of all the GOCIGs, as a rotation of Figure 3 suggests.

Figure 4 shows the geometry of a GOCIG with flat apparent dip. In this particular case, the imaging condition for ADCIGs has a direct "physical" explanation. The source and receiver rays can be associated with the corresponding planar wavefronts propagating in the same direction (and thus tilted by an angle $\gamma$ with respect to the horizontal). The crosscorrelation of the plane waves creates the angle-domain image point $I_\gamma$ where the plane waves intersect. $I_\gamma$ is shifted vertically by $\tilde{h}_0 \tan \gamma$ with respect to the offset-domain imaging point $I_0$. In this case, there is also a direct connection between the computation of ADCIGs in the image space and the computation of ADCIGs in the data space by plane-wave decomposition of the full prestack wavefield obtained by recursive survey sinking (Prucha et al., 1999).

The interpretation of ADCIGs in the "physical" space (Figure 4) can also be easily connected to the effects of applying slant stacks in the image space (Figure 5). Migration of a prestack flat event with too low a migration velocity generates an incompletely focused hyperbola in the image space, as sketched in Figure 5. According to equation (4), the tangent to the hyperbola at offset $x_h = x h$ has the slope $\partial z / \partial x_h = -\tan \gamma$. This tangent intersects the vertical axis at a point shifted by $\Delta n_\gamma = \tilde{h}_0 \tan \gamma n$ from $I_0$.

In the more general case of dipping reflectors (i.e. with $\alpha \neq 0$), when $x_h = \tilde{h}_0 / \cos \alpha$, the shift along the vertical is $x_h \tan \gamma n = (\tilde{h}_0 \tan \gamma / \cos \alpha) n$. This result is consistent with the geometric construction represented in Figure 3.

**ROBUST COMPUTATION OF ADCIGS IN PRESENCE OF GEOLOGICAL STRUCTURE**

Our first application of the CIG kinematic properties analyzed in the previous section is the definition of a robust method to compute high-quality ADCIGs for all events, including steeply
Figure 4: Geometry of a GOCIG with flat apparent dip. In this case, the source and receiver rays can be associated with the corresponding planar wavefronts propagating in the same direction. The crosscorrelation of the plane waves creates the angle-domain image point \( I_\gamma \) where the plane waves intersect.

Figure 5: Graphical analysis of the application of slant stacks to a GOCIG when an event with flat apparent dip is migrated with a low velocity. The event is an incompletely focused hyperbola in the image space. The tangent of this hyperbola at \( h_0 \) crosses the vertical axis at \( I_\gamma \).
dipping and overturned reflections. In presence of complex geological structure, the computation of neither the conventional HOCIGS nor the new VOCIGs is sufficient to provide complete velocity information, because the image is stretched along both the subsurface-offset axes.

According to equation (9), as the geological dip increases the horizontal-offset axis is stretched. At the limit, when \( \alpha \) is equal to 90 degrees, the relation between the horizontal-offset and the geological-dip offset becomes singular. Similarly, VOCIGs have problems when the geological dip is close to flat (\( \alpha = 0 \) degrees) and equation (10) becomes singular. This dip-dependent offset-stretching of the offset-domain CIGs causes artifacts in the corresponding ADCIGs.

The fact that relationships (9) and (10) diverge only for isolated dips (0, 90, 180, and 270 degrees) may falsely suggest that problems are limited to rare cases. However, in practice there are two factors that contribute to make the computation of ADCIGs in presence of geological dips prone to artifacts:

- To limit the computational cost, we would like to compute the offset-domain gathers over a range of offsets as narrow as possible. This is particularly true for shot-profile migrations, where the computation of the imaging conditions by equation (2) can add substantially to the computational cost when it is carried over a wide range of subsurface offsets.
- The attractive properties of the ADCIGs that we demonstrated above, including the elimination of the image-point dispersal, depend on the assumption of locally constant velocity. In particular, velocity is assumed to be constant along the ray segments \( \overline{S_x S_0} \), \( \overline{R_x R_0} \), \( \overline{S_z S_0} \), and \( \overline{R_z R_0} \) drawn in Figure 2. The longer those segments are, the more likely it is that the constant velocity assumption will be violated sufficiently to cause substantial errors.

These considerations suggest that, in presence of complex structures, high-quality ADCIGs ought to be computed using the information present in both HOCIGs and VOCIGs. There are two alternative strategies for obtaining a single set of ADCIGs from the information present in HOCIGs and VOCIGs. The first method merges HOCIGs with VOCIGs after they have been transformed to GOCIGs by the application of the offset stretching expressed in equation (16). The merged GOCIGs are then transformed to ADCIGs by applying the radial-trace transformation expressed in equation (18). The second method merges HOCIGs with VOCIGs directly in the angle domain, after both have been transformed to ADCIGs by the radial-trace transforms expressed in equations (4) and (7).

The two methods are equivalent if the offset range is infinitely wide, but they may have different artifacts when the offset range is limited. Since the first method merges the images in the offset domain, it can take into account the offset-range limitation more directly, and thus it has the potential to produce more accurate ADCIGs. However, the second method is more direct and simpler to implement. In both methods, an effective, though approximate, way for taking into account the limited offset ranges is to weight the CIGs as a function of the apparent
dips $\alpha$ in the image. A simple weighting scheme is:

$$w_{xh} = \cos^2 \alpha,$$
$$w_{zh} = \sin^2 \alpha,$$

(20)

where the weights $w_{xh}$ and $w_{zh}$ are respectively for the CIGs computed from the HOClGIs and the VOClGIs. These weights have the attractive property that their sum is equal to one for any $\alpha$. We used this weighting scheme for all the results shown in this paper.

**ADCIGs in the presence of geological structure: a North Sea example**

The following marine-data example demonstrates that the application of the robust method for computing ADCIGs presented in this section substantially improves the quality of ADCIGs in the presence of geological structure. Our examples show migration results of a 2-D line extracted from a 3-D data set acquired in the North Sea over a salt body with a vertical edge. The data were imaged using a shot-profile reverse time migration, because the reflections from the salt edge had overturned paths.

As predicted by our theory, in the presence of a wide range of reflector dips (e.g. flat sediments and salt edges), both the HOClGIs and the VOClGIs are affected by artifacts. Figure 6 illustrates this problem. It displays orthogonal sections cut through the HOClGI cube (Figure 6a), and through the VOClGI cube (Figure 6b). The front faces show the images at zero offset and are the same in the two cubes. The side face of Figure 6a shows the HOClGIs taken at the horizontal location corresponding to the vertical salt edge. We immediately notice that, at the depth interval corresponding to the salt edge, the image is smeared along the offset axis, which is consistent with the horizontal-offset stretch described by equation (9). On the contrary, the image of the salt edge is well focused in the VOClGI displayed in the top face of Figure 6b, which is consistent with the vertical-offset stretch described by equation (10). However, the flattish reflectors are unfocused in the VOClGI cube, whereas they are well focused in the HOClGI cube. The stretching of the offset axes causes useful information to be lost when significant energy is pushed outside the range of offsets actually computed. In this example, the salt edge reflection is clearly truncated in the HOClGI cube displayed in Figure 6a, notwithstanding that the image was computed for a fairly wide offset-range (800 meters, starting at -375 meters and ending at 425 meters).

The ADCIGs computed from either the HOClGIs or the VOClGIs have similar problems with artifacts caused by the wide range of reflector dips. Figure 7 shows the ADCIG computed from the offset-domain CIGs shown in Figure 6. The salt edge is smeared in the ADCIG computed from HOClGI (side face of Figure 7a), whereas it is fairly well focused in the ADCIG computed from VOClGI (top face of Figure 7b). Conversely, the flattish reflectors are well focused in the ADCIG computed from HOClGI, whereas they are smeared in the ADCIG computed from VOClGI.

The artifacts mostly disappear when the ADCIG cubes shown in Figure 7 are merged according to the simple scheme discussed above, which uses the weights defined in equations (20). Figure 8 shows the ADCIG cube resulting from the merge. The moveouts for the
salt edge and the sediment reflections are now clearly visible in the merged ADCIG cube and could be analyzed for extracting velocity information. To confirm these conclusions we migrated the same data after scaling the slowness function with a constant factor equal to 1.04. Figure 9 shows the ADCIG cubes computed from the HOClG cube (Figure 9a), and from the VOClG cube (Figure 9b). When comparing Figure 7 with Figure 9, we notice the 175-meter horizontal shift of the salt edge reflection toward the left, caused by the decrease in migration velocity. However, the artifacts related to the salt edge reflection are similar in the two figures, and they similarly obscure the moveout information. On the contrary, the moveout information is ready to be analyzed in the cube displayed in Figure 10, which shows the ADCIG cube resulting from the merge of the ADCIG cubes shown in Figure 9. In particular, both the flattish event above the salt edge (at about 1,000 meters depth) and the salt edge itself show a typical upward smile in the angle-domain gathers, indicating that the migration velocity was too slow.

ILLUSTRATION OF CIGS KINEMATIC PROPERTIES WITH A SYNTHETIC DATA SET

To verify the results of our geometric analysis of the kinematic properties of CIGs, we modeled and migrated a synthetic data set with a wide range of dips. The reflector has spherical shape with radius of 500 m. The center is at 1,000 meters depth and 3,560 meters horizontal coordinate. The velocity is constant and equal to 2,000 m/s. The data were recorded in 630 shot records. The first shot was located at a surface coordinate of -2,000 meters, and the shots were spaced 10 meters apart. The receiver array was configured with an asymmetric split-spread geometry. The minimum negative offset was constant and equal to -620 meters. The maximum offset was 4,400 meters for all the shots, with the exception of the first 100 shots (from -2,000 meters to -1,000 meters), where the maximum offset was 5,680 meters to record all the useful reflections. To avoid boundary artifacts at the top of the model, both sources and receivers were buried 250 meters deep. Some of the reflections from the top of the sphere were muted out before migration to avoid migration artifacts caused by spurious correlations with the first arrival of the source wavefield. The whole data set was migrated twice: first using the correct velocity (2,000 m/s), and second after scaling the slowness function by a constant factor $\rho = 1.04$ (corresponding to a velocity of 1,923 m/s). The ADCIGs shown in this section and the following section were computed by merging the ADCIGs computed from both the HOClGs and VOClGs according to the robust algorithm presented in the previous section.

Figure 11a shows the zero-offset section (stack) of the migrated cubes with the correct velocity and Figure 11b shows the zero-offset section obtained with the low velocity. Notice that, despite the large distance between the first shot and the left edge of the sphere (about 5,000 meters), normal incidence reflections illuminate the target only up to about 70 degrees. As we will see in the angle-domain CIGs, the aperture angle coverage shrinks dramatically with increasing reflector dip. On the other hand, real data cases are likely to have a vertical velocity gradient that improves the angle coverage of steeply dipping reflectors.
Figure 6: Migrated images of North Sea data set. Orthogonal sections cut through offset-domain CIG cubes: a) HOCIG cube, b) VOCIG cube. Notice the artifacts in both cubes.
Figure 7: Orthogonal sections cut through ADCIG cubes: a) ADCIG computed from HO-CIG cube, b) ADCIG computed from VOCIG cube. Notice the artifacts in both cubes that are related to the artifacts visible in the corresponding offset-domain CIG cubes (Figure 6).

[biondo1-Ang-Cube-both-v7newsc-overn] [CR]
Figure 8: Orthogonal sections cut through the ADCIG cube that was obtained by merging the cubes displayed in Figure 7 using the proposed method. Notice the lack of artifacts compared with Figure 7.

Transformation of HOCIGs and VOCIGs to GOCIGs

Figure 12 illustrates the differences between HOCIGs and VOCIGs caused by the image-point shift, and it demonstrates that the image-point shift is corrected by the transformation to GOCIGs described in equations (9) and (10).

Figures 12a and 12b show orthogonal sections cut through the offset-domain image cubes in the case of the low velocity migration. Figure 12a displays the horizontal-offset image cube, while Figure 12b displays the vertical-offset image cube. Notice that the offset axis in Figure 12b has been reversed to facilitate its visual correlation with the image cube displayed in Figure 12a. The side faces of the cubes display the CIGs taken at the surface location corresponding to the apparent geological dip of 45 degrees. The events in the two types of CIGs have similar shapes, as expected from the geometric analysis presented in a previous section \( \cos \alpha = \sin \alpha \) when \( \alpha = 45 \) degrees), but their extents are different. The differences between the two image cubes are more apparent when comparing the front faces, which show the image at a constant offset of 110 meters (-110 meters in Figure 12b). These differences are due to the differences in image-point shift for the two offset directions [equation (11) and equation (12)].

Figure 12c and 12d show the image cubes of Figures 12a and 12b after the application of the transformations to GOCIG, described in equations (9) and (10), respectively. The two transformed cubes are almost identical, because both the offset stretching and the image-point shift have been removed. The only significant differences are visible in the front face for the reflections corresponding to the top of the sphere. These reflections cannot be fully cap-
Figure 9: Migrated images of North Sea data set. The migration slowness had been scaled by 1.04 with respect to the migration slowness used for the images shown in Figures 6–8. Orthogonal sections cut through ADCIG cubes: a) ADCIG computed from HOCIG cube, b) ADCIG computed from VOCIG cube. Notice that the artifacts obscure the moveout information in both cubes.
Figure 10: Orthogonal sections cut through the ADCIG cube that was obtained by merging the cubes displayed in Figure 9 using the proposed method. Notice the typical upward smile in the moveouts from both the salt edge and the flattish event above it.

...tured within the vertical-offset image cube because the expression in equation (10) diverges as \( \alpha \) goes to zero. Similarly, reflections from steeply dipping events are missing from the horizontal-offset image cube because the expression in equation (9) diverges as \( \alpha \) goes to 90 degrees.

**Image mispositioning in ADCIGs migrated with wrong velocity**

In a previous section, we demonstrated that in an ADCIG cube the imaging point \( I_y \) lies on the line normal to the apparent geological dip and passing through the point where the source and receiver rays cross (Figure 3). This geometric property enabled us to define the analytical relationship between reflector movement and traveltime perturbation expressed in equation (15). This important result is verified by the numerical experiment shown in Figure 13. This figure compares the images of the spherical reflector obtained using the low velocity (slowness scaled by \( \rho = 1.04 \)) with the reflector position computed analytically under the assumption that \( I_y \) is indeed the image point in an ADCIG. Because both the true and the migration velocity functions are constant, the migrated reflector location can be computed exactly by a simple “kinematic migration” of the recorded events. This process takes into account the difference in propagation directions between the “true” events and the “migrated” events caused by the scaling of the velocity function. Appendix C derives the equations used to compute the migrated reflector location as a function of \( \rho, \alpha\rho, \) and \( \gamma\rho \).
The images shown in the six panels in Figure 13 correspond to six different apparent aperture angles: a) $\gamma_{\rho} = 0$, b) $\gamma_{\rho} = 10$, c) $\gamma_{\rho} = 20$, d) $\gamma_{\rho} = 30$, e) $\gamma_{\rho} = 40$, f) $\gamma_{\rho} = 50$. The black lines superimposed onto the images are the corresponding reflector locations predicted by the relationships derived in Appendix C. The analytical lines perfectly track the migrated images for all values of $\gamma_{\rho}$. The lines terminate when the corresponding event was not recorded by the data acquisition geometry (described above). The images extend beyond the termination of the analytical lines because the truncation artifacts are affected by the finite-frequency nature of the seismic signal, and thus they are not predicted by the simple kinematic modeling described in Appendix C.

**RESIDUAL MOVEOUT IN ADCIGS**

The inconsistencies between the migrated images at different aperture angles are the primary source of information for velocity updating during Migration Velocity Analysis (MVA). Figure 13 demonstrated how the reflector mispositioning caused by velocity errors can be exactly predicted by a kinematic migration that assumes the image point to lie on the normal to the apparent geological dip. However, this exact prediction is based on the knowledge of the true velocity model. Of course, this condition is not realistic when we are actually trying to estimate the true velocity model by MVA. In these cases, we first measure the inconsistencies between the migrated images at different aperture angles, and then we “invert” these measures into perturbations of the velocity model.

An effective and robust method for measuring inconsistencies between images is to compute semblance scans as a function of one “residual moveout” (RMO) parameter, and then pick the maxima of the semblance scan. This procedure is most effective when the residual move-
Figure 12: Orthogonal sections cut through offset-domain CIG cubes obtained with too low velocity ($\rho = 1.04$): a) HO-CIG cube, b) VO-CIG cube, c) GO-CIG cube computed from HO-CIG cube, d) GO-CIG cube computed from VO-CIG cube. Notice the differences between the HO-CIG (panel a) and the VO-CIG (panel b) cubes, and the similarities between the GO-CIG cubes (panel c and panel d).

...out function used for computing the semblance scans closely approximates the true moveouts in the images. In this section, we use the kinematic properties that we derived and illustrated in the previous sections to derive two alternative RMO functions for scanning ADCIGs computed from wavefield-continuation migration.

As discussed above, the exact relationships derived in Appendix C cannot be used, because the true velocity function is not known. Thus we cannot realistically estimate the changes in ray-propagation directions caused by velocity perturbations. However, we can linearize the relations and estimate the reflector movement by assuming that the raypaths are stationary. This assumption is consistent with the typical use of measured RMO functions by MVA procedures. For example, in a tomographic MVA procedure the velocity is updated by applying a tomographic scheme that “backprojects” the image inconsistencies along unperturbed ray-paths. Furthermore, the consequences of the errors introduced by neglecting ray bending are...
Figure 13: Comparison of the actual images obtained using the low velocity, with the reflector position computed analytically under the assumption that the image point lies on the normal to the apparent geological dip \( I_\rho \) in Figure 3. The black lines superimposed onto the images are the reflector locations predicted by the relationships presented in Appendix C. The six panels correspond to six different apparent aperture angles: 

- a) \( \gamma_\rho = 0 \)
- b) \( \gamma_\rho = 10 \)
- c) \( \gamma_\rho = 20 \)
- d) \( \gamma_\rho = 30 \)
- e) \( \gamma_\rho = 40 \)
- f) \( \gamma_\rho = 50 \).

significantly reduced by the fact that RMO functions describe the movements of the reflectors relative to the reflector position imaged at normal incidence \( (\gamma = 0) \), not the absolute movements of the reflectors with respect to the true (unknown) reflector position.

Appendix D derives two expressions for the RMO shift along the normal to the reflector \( (\Delta n_{\text{RMO}}) \), under the assumptions of stationary raypaths and constant scaling of the slowness function by a factor \( \rho \). The first expression is \[ \Delta n_{\text{RMO}} = \frac{1 - \rho}{1 - \rho (1 - \cos \alpha)} \frac{\sin^2 \gamma}{\cos^2 \alpha - \sin^2 \gamma} z_0 n, \] (21)

where \( z_0 \) is the depth at normal incidence.

The second RMO function is directly derived from the first by assuming flat reflectors \( (\alpha = 0) \) \[ \Delta n_{\text{RMO}} = (1 - \rho) \tan^2 \gamma z_0 n. \] (22)
As expected, in both expressions the RMO shift is null at normal incidence ($\gamma = 0$), and when the migration slowness is equal to the true slowness ($\rho = 1$).

According to the first expression [equation (21)], the RMO shift increases as a function of the apparent geological dip $|\alpha|$. The intuitive explanation for this behavior is that the rays become longer as the apparent geological dip increases, and consequently the effects of the slowness scaling increase. The first expression is more accurate than the second one when the spatial extent of the velocity perturbations is large compared to the raypath length, and consequently the velocity perturbations are uniformly felt along the entire raypaths. Its use might be advantageous at the beginning of the MVA process when slowness errors are typically large scale. However, it has the disadvantage of depending on the reflector dip $\alpha$, and thus its application is somewhat more complex.

The second expression is simpler and is not as dependent on the assumption of large-scale velocity perturbations as the first one. Its use might be advantageous for estimating small-scale velocity anomalies at a later stage of the MVA process, when the gross features of the slowness function have been already determined.

To test the accuracy of the two RMO functions we will use the migration results of a synthetic data set acquired over a spherical reflector. This data set was described in the previous section. Figure 14 illustrates the accuracy of the two RMO functions when predicting the actual RMO in the migrated images obtained with a constant slowness function with $\rho = 1.04$. The four panels show the ADCIGs corresponding to different apparent reflector dip: a) $\alpha = 0$; b) $\alpha = 30$; c) $\alpha = 45$; d) $\alpha = 60$. Notice that the vertical axes change across the panels; in each panel the vertical axis is oriented along the direction normal to the respective apparent geological dip. The solid lines superimposed onto the images are computed using equation (21), whereas the dashed lines are computed using equation (22). As in Figure 13, the images extend beyond the termination of the analytical lines because of the finite-frequency nature of the truncation artifacts.

The migrated images displayed in Figure 14 were computed by setting both the true and the migration slowness function to be constant. Therefore, this case favors the first RMO function [equation (21)] because it nearly meets the conditions under which equation (21) was derived in Appendix D. Consequently, the solid lines overlap the migration results for all dip angles. This figure demonstrates that, when the slowness perturbation is sufficiently small (4 % in this case), the assumption of stationary raypaths causes only small errors in the predicted RMO.

On the contrary, the dashed lines predicted by the second RMO function [equation (22)] are an acceptable approximation of the actual RMO function only for small dip angles (up to 30 degrees). For large dip angles, a value of $\rho$ substantially higher than the correct one would be necessary to fit the actual RMO function with equation (22). If this effect of the reflector dip is not properly taken into account, the false indications provided by the inappropriate use of equation (22) can prevent the MVA process from converging.
Figure 14: ADCIGs for four different apparent reflector dips: a) $\alpha = 0$; b) $\alpha = 30$; c) $\alpha = 45$; d) $\alpha = 60$ with $\rho = 1.04$. Superimposed onto the images are the RMO functions computed using equation (21) (solid lines), and using equation (22) (dashed lines). Notice that the vertical axes change across the panels; in each panel the vertical axis is oriented along the direction normal to the respective apparent geological dip.
CONCLUSIONS

We analyzed the kinematic properties of ADCIGs in presence of velocity errors. We proved that in the angle domain the image point lies along the normal to the apparent reflector dip. This geometric property of ADCIGs makes them immune to the image-point dispersal and thus attractive for MVA.

We derived a quantitative relationship between image-point movements and traveltime perturbations caused by velocity errors, and verified its validity with a synthetic-data example. This relationship should be at the basis of velocity-updating methods that exploit the velocity information contained in ADCIGs.

Our analysis leads to the definition of two RMO functions that can be used to measure inconsistencies between migrated images at different aperture angles. The RMO functions describe the relative movements of the imaged reflectors only approximately, because they are derived assuming stationary raypaths. However, a synthetic example shows that, when the velocity perturbation is sufficiently small, one of the proposed RMO functions is accurate for a wide range of reflector dips and aperture angles.

The insights gained from our kinematic analysis explain the strong artifacts that affect conventional ADCIG in presence of steeply dipping reflectors. They also suggest a procedure for overcoming the problem: the computation of vertical-offset CIGs (VOCIGs) followed by the combination of VOCIGs with conventional HOCIGs. We propose a simple and robust scheme for combining HOCIGs and VOCIGs. A North Sea data example clearly illustrates both the need for and the advantages of our method for computing ADCIGs in presence of a vertical salt edge.

ACKNOWLEDGMENTS

We thank Guojian Shan for helping in the development of the program that we used to migrate both the synthetic and the real data sets. We also thank Henri Calandra and TotalFinaElf for making the North Sea data set available to the Stanford Exploration Project (SEP).

REFERENCES


APPENDIX A

PROOF THAT THE TRANSFORMATION TO GOCIG CORRECTS FOR THE IMAGE-POINT SHIFT

This appendix proves that by applying the offset transformations described in equations (9) and (10) we automatically remove the image-point shift characterized by equations (11) and (12). The demonstration for the VOCIG transformation is similar to the one for the HOCIG transformation, and thus we present only the demonstration for the HOCIGs. HOCIGs are transformed into GOCIGs by applying the following change of variables of the offset axis $x_h$, in the vertical wavenumber $k_z$ and horizontal wavenumber $k_x$ domain:

$$
x_h = \frac{\tilde{h}_0}{\cos \alpha} = \text{sign}(\tan \alpha)\tilde{h}_0\sqrt{1 + \tan^2 \alpha} = \text{sign}\left(\frac{k_x}{k_z}\right)\tilde{h}_0\left(1 + \frac{k_x^2}{k_z^2}\right)^{-\frac{1}{2}}.
$$

(A-1)

For the sake of simplicity, in the rest of the appendix we will drop the sign in front of expression (A-1) and consider only the positive values of $k_x/k_z$.

We want to prove that by applying (A-1) we also automatically shift the image by

$$
\Delta_z I_{x_h} = -\tilde{h}_0 \tan \gamma \tan \alpha \sin \alpha
$$

(A-2)

in the vertical direction, and

$$
\Delta_x I_{x_h} = \tilde{h}_0 \tan \gamma \tan \alpha \cos \alpha
$$

(A-3)

in the horizontal direction.

The demonstration is carried out in two steps: 1) we compute the kinematics of the impulse response of transformation (A-1) by a stationary-phase approximation of the inverse Fourier transform along $k_z$ and $k_x$, and 2) we evaluate the dips of the impulse response, relate them to the angles $\alpha$ and $\gamma$, and then demonstrate that relations (A-3) and (A-2) are satisfied.
Evaluation of the impulse response of the transformation to GOCIGs

The transformation to GOCIG of an image \( I_{x_h}(k_z, k_x, x_{h}) \) is defined as

\[
I_{0}(k_z, k_x, k_{h}) = \int d\tilde{h}_{0} I_{0}(k_z, k_x, \tilde{h}_{0}) e^{i k_{h} \tilde{h}_{0}} = \int d x_{h} \left( \frac{d\tilde{h}_{0}}{d x_{h}} \right) I_{x_h}(k_z, k_x, x_{h}) e^{i k_{h} x_{h} \left( \frac{1}{k_z^2 x_{h}^2} \right)^{\frac{1}{2}}}. \tag{A-4}
\]

The transformation to GOCIG of an impulse located at \((\bar{z}, \bar{x}, \bar{x}_{h})\) is thus (after inverse Fourier transforms):

\[
\tilde{\text{Imp}}(z, x, \tilde{h}_{0}) = \int d k_{h} \int d x_{h} \int d k_{x} \int d k_{z} \left( \frac{d\tilde{h}_{0}}{d x_{h}} \right) e^{i \left\{ k_{h} \left[ \bar{x}_{h} \left( 1 + \frac{k_{x}^2}{k_{z}^2} \right)^{-\frac{1}{2}} - \tilde{h}_{0} \right] + k_{z}(\bar{z} - z) + k_{x}(\bar{x} - x) \right\}}. \tag{A-5}
\]

We now approximate by stationary phase the inner double integral. The phase of this integral is:

\[
\Phi \equiv k_{h} \left[ \bar{x}_{h} \left( 1 + \frac{k_{x}^2}{k_{z}^2} \right)^{-\frac{1}{2}} - \tilde{h}_{0} \right] + k_{z}(\bar{z} - z) + k_{x}(\bar{x} - x). \tag{A-6}
\]

The stationary path is defined by the solutions of the following system of equations:

\[
\frac{\partial \Phi}{\partial k_{z}} = k_{h} \bar{x}_{h} \frac{k_{x}^2}{k_{z}^2} \left( 1 + \frac{k_{x}^2}{k_{z}^2} \right)^{-\frac{3}{2}} + (\bar{z} - z) = 0, \tag{A-7}
\]

\[
\frac{\partial \Phi}{\partial k_{x}} = -k_{h} \bar{x}_{h} \frac{k_{x}}{k_{z}} \left( 1 + \frac{k_{x}^2}{k_{z}^2} \right)^{-\frac{3}{2}} + (\bar{x} - x) = 0. \tag{A-8}
\]

By moving both \((\bar{z} - z)\) and \((\bar{x} - x)\) to the right of equations (A-7) and (A-8), and then dividing equation (A-7) by equation (A-8), we obtain the following relationship between \((\bar{z} - z)\) and \((\bar{x} - x)\):

\[
\frac{\bar{z} - z}{\bar{x} - x} = \frac{k_{x}}{k_{z}}. \tag{A-9}
\]

Furthermore, by multiplying equation (A-7) by \(k_{z}\) and equation (A-8) by \(k_{x}\), and then substituting them appropriately in the phase function (A-6), we can evaluate the phase function along the stationary path as follows:

\[
\Phi_{\text{stat}} = k_{h} \left[ \bar{x}_{h} \left( 1 + \frac{k_{x}^2}{k_{z}^2} \right)^{-\frac{1}{2}} - \tilde{h}_{0} \right], \tag{A-10}
\]

which becomes, by substituting equation (A-9),

\[
\Phi_{\text{stat}} = k_{h} \left\{ -\bar{x}_{h} \left[ 1 + \left( \frac{(\bar{z} - z)^2}{(\bar{x} - x)^2} \right) \right]^{-\frac{1}{2}} - \tilde{h}_{0} \right\}. \tag{A-11}
\]
Notice that the minus sign comes from the sign function in expression (A-1). By substituting expression (A-11) in equation (A-5) it is immediate to evaluate the kinematics of the impulse response as follows:

\[ \tilde{h}_0 = -x_h \left[ 1 + \frac{(\bar{z} - z)^2}{(\bar{x} - x)^2} \right]^{-\frac{1}{2}} . \]  (A-12)

**Evaluation of the image shift as a function of \( \alpha \) and \( \gamma \)**

The final step is to take the derivative of the impulse response of equation (A-12) and use the relationships of these derivatives with \( \tan \alpha \) and \( \tan \gamma \):

\[ \frac{\partial z}{\partial x} = \tan \alpha = \sqrt{\frac{x_h^2}{\tilde{h}_0^2} - 1} , \]  (A-13)

\[ -\frac{\partial z}{\partial x_h} = \tan \gamma = -(\bar{x} - x) \frac{x_h}{\tilde{h}_0} \frac{\bar{z} - z}{\sqrt{\frac{x_h^2}{\tilde{h}_0^2} - 1}} = -(\bar{z} - z) \frac{x_h}{\tilde{h}_0} \]  (A-14)

Substituting equations (A-13) and (A-14) into the following relationships:

\[ \Delta z \ I_{x_h} = \bar{z} - z = -\tilde{h}_0 \tan \gamma \tan \alpha \sin \alpha , \]  (A-15)

\[ \Delta x \ I_{x_h} = \bar{x} - x = \tilde{h}_0 \tan \gamma \tan \alpha \cos \alpha , \]  (A-16)

and after some algebraic manipulation, we prove the thesis.

**APPENDIX B**

This appendix demonstrates equations (19) in the main text: that for energy dipping at an angle \( \alpha \) in the \( (z, x) \) plane, the wavenumber \( k_n \) along the normal to the dip is linked to the wavenumbers \( k_z \) and \( k_x \) by the following relationships:

\[ k_n = \frac{k_z \cos \alpha}{\sin \alpha} = \frac{k_x}{\sin \alpha} . \]  (B-1)

For energy dipping at an angle \( \alpha \) the wavenumbers satisfy the well-known relationship

\[ \tan \alpha = \frac{k_x}{k_z} , \]  (B-2)

where the positive sign is determined by by the conventions defined in Figure 1. The wavenumber \( k_n \) is related to \( k_x \) and \( k_z \) by the axes rotation

\[ k_n = k_z \cos \alpha + k_x \sin \alpha . \]  (B-3)
Substituting equation (B-2) into equation (B-3) we obtain
\[ k_n = \frac{k_z}{\cos \alpha} \left( \cos^2 \alpha + \tan \alpha \cos \alpha \sin \alpha \right) = \frac{k_z}{\cos \alpha} \left( \cos^2 \alpha + \sin^2 \alpha \right) = \frac{k_z}{\cos \alpha}, \quad (B-4) \]
or,
\[ k_n = \frac{k_x}{\sin \alpha} \left( \cot \alpha \sin \alpha \cos \alpha + \sin^2 \alpha \right) = \frac{k_x}{\sin \alpha} \left( \cos^2 \alpha + \sin^2 \alpha \right) = \frac{k_x}{\sin \alpha}. \quad (B-5) \]

**APPENDIX C**

In this appendix we derive the equations for the “kinematic migration” of the reflections from a sphere, as a function of the ratio \( \rho \) between the true constant slowness \( S \) and the migration slowness \( S_\rho = \rho S \). For a given \( \rho \) we want to find the coordinates of the imaging point \( I_\gamma(z_\gamma, x_\gamma) \) as a function of the apparent geological dip \( \alpha_\rho \) and the apparent aperture angle \( \gamma_\rho \). Central to our derivation is the assumption that the imaging point \( I_\gamma \) lies on the normal to the apparent reflector dip passing through \( \bar{I} \), as represented in Figure 3.

The first step is to establish the relationships between the true \( \alpha \) and \( \gamma \) and the apparent \( \alpha_\rho \) and \( \gamma_\rho \). This can be done through the relationships between the propagation directions of the source/receiver rays (respectively marked as the angles \( \beta \) and \( \delta \) in Figure 1), and the event time dips, which are independent on the migration slowness. The true \( \beta \) and \( \delta \) can be thus estimated as follows:

\[ \beta = \arcsin \left( \rho \sin \beta_\rho \right) = \arcsin \left[ \rho \sin \left( \alpha_\rho - \gamma_\rho \right) \right], \quad (C-1) \]
\[ \delta = \arcsin \left( \rho \sin \delta_\rho \right) = \arcsin \left[ \rho \sin \left( \alpha_\rho + \gamma_\rho \right) \right]; \quad (C-2) \]

and then the true \( \alpha \) and \( \gamma \) are:

\[ \alpha = \frac{\beta + \delta}{2}, \quad \text{and} \quad \gamma = \frac{\delta - \beta}{2}. \quad (C-3) \]

Next step is to take advantage of the fact that the reflector is a sphere, an thus that the coordinates \((\hat{z}, \hat{x})\) of the true reflection point are uniquely identified by the dip angle \( \alpha \) as follows:

\[ \hat{z} = (z_c - R \cos \alpha), \quad \text{and} \quad \hat{x} = (x_c + R \sin \alpha), \quad (C-4) \]

where \((z_c, x_c)\) are the coordinates of the center of the sphere and \( R \) is its radius.

The midpoint, offset, and traveltime of the event can be found by applying simple trigonometry (see Sava and Fomel (2002)) as follows:

\[ x_{h, \text{surf}} = \sin \gamma \cos \gamma \frac{\cos \alpha}{\cos^2 \alpha - \sin^2 \gamma} \hat{z}, \quad (C-5) \]
\[ x_{m, \text{surf}} = \hat{x} + \sin \alpha \cos \alpha \frac{\cos \gamma}{\cos^2 \alpha - \sin^2 \gamma} \hat{z}, \quad (C-6) \]
\[ t_D = 2S \frac{\cos \alpha \cos \gamma}{\cos^2 \alpha - \sin^2 \gamma} \hat{z}. \quad (C-7) \]
The coordinates of the point $\bar{I}(\bar{z}, \bar{x})$, where the source and the receiver rays cross, are:

\[
\bar{z} = x_{h\text{surf}} \frac{\cos^2 \alpha - \sin^2 \gamma}{\sin \gamma \cos \gamma}, \quad (C-8)
\]

\[
\bar{x} = x_{m\text{surf}} - \frac{\sin \alpha \cos \alpha}{\cos^2 \alpha - \sin^2 \gamma} \bar{z} = \frac{\sin \alpha \cos \alpha}{\cos^2 \alpha - \sin^2 \gamma} x_{m\text{surf}} = \frac{\sin \alpha \cos \alpha}{\sin \gamma \cos \gamma} x_{h\text{surf}}, \quad (C-9)
\]

and the corresponding traveltime $t_{D\rho}$ is:

\[
t_{D\rho} = 2 \rho S \frac{\cos \alpha \cos \gamma}{\cos^2 \alpha - \sin^2 \gamma} \bar{z}. \quad (C-10)
\]

Once we have the traveltimes $t_D$ and $t_{D\rho}$, the normal shift $\Delta n_{\text{tot}}$ can be easily evaluated by applying equation (15) (where the background velocity is $S_\rho$ and the aperture angle is $\gamma_\rho$), which yields:

\[
\Delta n_{\text{tot}} = -\frac{(t_{D\rho} - t_D)}{2 \rho S \cos \gamma} n. \quad (C-11)
\]

We use equation (C-11), together with equations (C-8) and (C-9), to compute the lines superimposed onto the images in Figure 13.

**APPENDIX D**

In this Appendix we derive the expression for the residual moveout (RMO) function to be applied to ADCIGs computed by wavefield continuation. The derivation follows the derivation presented in Appendix C. The main difference is that in this appendix we assume the rays to be stationary. In other words, we assume that the apparent dip angle $\alpha_\rho$ and aperture angle $\gamma_\rho$ are the same as the true angles $\alpha$ and $\gamma$. This assumption also implies that the (unknown) true reflector position $(\hat{z}, \hat{x})$ coincides with the point $I(\bar{z}, \bar{x})$ where the source and the receiver ray cross.

Given these assumptions, the total traveltime through the perturbed slowness function $S_\rho$ is given by the following expression:

\[
t_{D\rho} = 2 \rho S \frac{\cos \alpha \cos \gamma}{\cos^2 \alpha - \sin^2 \gamma} \bar{z}, \quad (D-1)
\]

which is different from the corresponding equation in Appendix C [equation (C-10)]. The difference in traveltimes $(t_{D\rho} - t_D)$, where $t_D$ is given by equation equation (C-7), is thus a linear function of the difference in slownesses $[(\rho - 1)S]$; that is,

\[
t_{D\rho} - t_D = 2 (\rho - 1) S \frac{\cos \alpha \cos \gamma}{\cos^2 \alpha - \sin^2 \gamma} \bar{z}. \quad (D-2)
\]
As in Appendix C, the normal shift $\Delta n_{tot}$ can be evaluated by applying equation (15) (where the background velocity is $S_\rho$ and the aperture angle is $\gamma$), which yields:

$$
\Delta n_{tot} = \frac{1 - \rho}{\rho} \frac{\cos \alpha}{\cos^2 \alpha - \sin^2 \gamma} \bar{z} \mathbf{n}.
$$

(D-3)

The RMO function ($\Delta n_{RMO}$) describes the relative movement of the image point at any $\gamma$ with respect to the image point for the normal-incidence event ($\gamma = 0$). From equation (D-3), it follows that the RMO function is:

$$
\Delta n_{RMO} = \Delta n_{tot}(\gamma) - \Delta n_{tot}(\gamma = 0) = \frac{1 - \rho}{\rho} \left[ \frac{\cos \alpha}{\cos^2 \alpha - \sin^2 \gamma} - \frac{1}{\cos \alpha} \right] \bar{z} \mathbf{n} = \frac{1 - \rho}{\rho} \frac{\sin^2 \gamma}{\cos^2 \alpha - \sin^2 \gamma \cos \alpha} \bar{z} \mathbf{n}.
$$

(D-4)

The true depth $\bar{z}$ is not known, but at normal incidence it can be estimated as a function of the migrated depth $z_0$ by inverting the following relationship:

$$
z_0 = \left( \frac{1 - \rho}{\rho \cos \alpha} + 1 \right) \bar{z},
$$

(D-5)
as:

$$
\bar{z} = \left[ \frac{\rho \cos \alpha}{1 - \rho (1 - \cos \alpha)} \right] z_0.
$$

(D-6)

Substituting relation (D-6) in equation (D-4) we obtain the result:

$$
\Delta n_{RMO} = \frac{1 - \rho}{\rho} \frac{\sin^2 \gamma}{1 - \rho (1 - \cos \alpha) (\cos^2 \alpha - \sin^2 \gamma)} z_0 \mathbf{n},
$$

(D-7)

which for flat reflectors ($\alpha = 0$) simplifies into:

$$
\Delta n_{RMO} = (1 - \rho) \tan^2 \gamma z_0 \mathbf{n}.
$$

(D-8)

In Figure 14, the solid lines superimposed into the images are computed using equation (D-7), whereas the dashed lines are computed using equation (D-8).
INTRODUCTION

Angle-Domain Common-Image Gathers (ADCIGs) are often used for velocity analysis or Amplitude Versus Angle studies. Wavefield-continuation methods can easily generate angle gathers before (Prucha et al., 1999) or after imaging (Sava and Fomel, 2000). The method proposed by Sava and Fomel (2000) assumes that the source and receiver rays propagate in a vertical plane, or that the source-receiver orientation is parallel to the dip of the reflector. This assumption is valid in 2-D and in some specific 3-D cases. In more general cases, however, this assumption and the method need to be revised. We propose an extension of the method such as it can be applied after the simpler case of common-azimuth migration, or after a more general full prestack migration. We present an application of the implementation of both extensions.

DEFINITION OF THE 3-D PROBLEM

The 3-D construction in Figure 1 generalizes the 2-D construction presented in Biondi and Symes (2003). If the migration velocity is not correct, the source and receiver rays may not be coplanar at the image point. We limit ourselves, though, to the case where the two rays are coplanar. An incorrect migration velocity introduces also a premature focusing of the energy if the velocity is too slow, or a late focusing if the velocity is too fast. Given this geometry, and considering only the constant \( z \) Horizontal Offset Common Image Gather, we examine how to transform an offset gather into an angle gather.

OFFSET TO ANGLE TRANSFORMATION IN COMMON-AZIMUTH MIGRATION

We first present the simplest extension to 3-D: the common-azimuth case.

In 2-D, the transformation from offset-gather to angle-gather is accomplished by applying
Figure 1: 3-D geometry of coplanar source and receiver rays when using too slow a velocity. Due to the incorrect velocity, source and receiver rays do not end up meeting at the true image point $I$. They stop instead in $S_0$ and $R_0$. $I_z$ is the image point of a Horizontal Offset Common Image Gather (at constant $z$). $I_g$ is the image point in Angle Domain Common Image Gather.

the relation (Sava and Fomel, 2000)

\[
\tan \gamma = \frac{k_{h_x}}{k_z}, \tag{1}
\]

where $\gamma$ is the aperture angle, $k_{h_x}$ the in-line offset wavenumber and $k_z$ the vertical wavenumber.

This relation can be applied in 3-D, but it is then assumed that the two rays propagate vertically. In 3-D however, rays can propagate out of the vertical plane. In the common-azimuth assumption, the two rays propagate within a slanted plane. We call $\delta$ the dip angle of the slanted propagation plane. The transformation from offset to angle needs to take into account $\delta$. The new form of equation (1) is

\[
\tan \gamma = \frac{k_{h_x}}{k_z} \cos \delta = \frac{k_{h_x}}{k_z} \frac{1}{\sqrt{1 + \frac{k_{m_y}^2}{k_z^2}}}, \tag{2}
\]

where $k_{m_y}$ is the cross-line midpoint wavenumber. We give in Appendix A a geometrical and an analytical derivation of equation (2). This equation is valid for common-azimuth migration, that is to say, for an in-line orientation of the source-receiver axis and for the coplanarity of the source and receiver rays.

Application

We computed angle gathers using equation (2) from a common-azimuth migrated SEG-EAGE salt dataset. Figure 2 shows a depth slice taken at a depth of 580 meters. Figure 3 shows two ADCIGs taken at an in-line location of 8,350 meters and a cross-line location of 5,250 meters. The ADCIG on the left was computed using the 2-D relationship (equation (1)) for the transformation to angle domain. The ADCIG on the right was computed using the correct 3-D relationship (equation (2)). The $\cos \delta$ term corrects the ADCIG for the top of the salt reflection...
(z ≃ 580 meters) that dips at approximately 50 degrees in the cross-line direction. Notice that the bottom of the salt reflection (z ≃ 2,100 meters) is unaffected by the cos δ term because it is flat.

Figure 2: Depth slice cut through the common-azimuth image of the SEG-EAGE salt dataset at a depth of 580 meters. The crossing point of the grid lines indicates the location of the ADCIGs shown in Figure 3.

**FULL PRESTACK MIGRATION**

In the previous section we have presented the simple common-azimuth case. We examine here how to extend the common-azimuth case to the full prestack case. There are two main differences. First, the rays may not be coplanar anymore along the travel path from the surface to the reflection point because of a cross-line offset due to ray bending (Biondi, 2001). If the migration velocity is correct, though, the two rays meet at the reflection point where they must be coplanar. Second, the non-coplanarity of the rays during the propagation induces the azimuth of the reflection to be different from the in-line direction.

The coplanarity is imposed at the reflection point by the coplanarity condition introduced by Biondi and Palacharla (1996). Some algebraic manipulations presented in annexe B allow us to write the coplanarity condition as

\[ k_{h_y} = \frac{k_{m_y} k_{m_x} k_{h_x}}{k_{m_x}^2 + k_{m_y}^2}. \]  

(3)

where \( k_{m_x} \) and \( k_{m_y} \) are the two components of the midpoint wavenumber, and \( k_{h_x} \) and \( k_{h_y} \) are the two components of the offset wavenumber.

The non-zero reflection azimuth issue is addressed by doing a local change of coordinates such as the azimuth is zero in the new coordinates system. If \( \beta \) is the azimuth of the reflection in the original coordinates system \((x, z)\), a rotation by an angle \( \beta \) makes the reflection azimuth to be parallel to the new in-line direction in the new coordinates system \((x', y')\) (cf. Figure 4). In the new coordinates system, the midpoint wavenumber \( \mathbf{k}_m' \) and the offset wavenumber \( \mathbf{k}_h' \) are defined by:

\[
\mathbf{k}_m' = \begin{bmatrix} k_{m_x}' \\ k_{m_y}' \end{bmatrix} = \begin{bmatrix} \cos \beta & -\sin \beta \\ \sin \beta & \cos \beta \end{bmatrix} \begin{bmatrix} k_{m_x} \\ k_{m_y} \end{bmatrix} = \begin{bmatrix} \cos \beta k_{m_x} - \sin \beta k_{m_y} \\ \sin \beta k_{m_x} + \cos \beta k_{m_y} \end{bmatrix},
\]

(4)
Figure 3: ADCIGs taken at in-line location of 8,350 meters and cross-line location of 5,250 meters: a) computed using the 2-D relationship for the transformation to angle domain (equation (1)). b) computed using the correct 3-D relationship (equation (2)).

\[
\begin{align*}
\mathbf{k}_h' &= \begin{bmatrix} k_{hx}' \\ k_{hy}' \end{bmatrix} = \begin{bmatrix} \cos \beta & -\sin \beta \\ \sin \beta & \cos \beta \end{bmatrix} \begin{bmatrix} k_{hx} \\ k_{hy} \end{bmatrix} = \begin{bmatrix} \cos \beta k_{hx} - \sin \beta k_{hy} \\ \sin \beta k_{hx} + \cos \beta k_{hy} \end{bmatrix} \\
\end{align*}
\]  

With these transformations, equations (2) and (3) can be rewritten for a non-zero azimuth of the reflection:

\[
\tan \gamma = \frac{k_{hx}'}{k_z} \frac{1}{\sqrt{1 + \frac{k_{my}^2}{k_z^2}}} \tag{6}
\]

\[
\frac{k_{hy}'}{k_z^2 + k_{my}^2} \tag{7}
\]

**Application**

We apply equation (2) on a five slanted planes model (Vaillant and Biondi, 2000). The algorithm is given in Appendix C. The dips are 0°, 15°, 30°, 45° and 60°, and their azimuth is 45°. The velocity function is \( v(z) = 1.5 + 0.5z \) km/s. Because of the velocity gradient, there is ray-bending. The ray bending induces a rotation of the source-receiver axis which breaks the common-azimuth assumption. The steeper the plane, the stronger the bending of the rays and
Figure 4: The source-receiver axis is in the $x'$ direction. Axis $x'$ is the image of axis $x$ after a rotation by $\beta$. In the new basis, the reflection has zero azimuth.  

the wider the rotation of the azimuth of the reflection ($\beta$). Figures 5, 6 and 7 show the reflections for three different azimuths of the reflection. 2) for 3 different values of the reflection azimuth angle. Figure 5 has been computed for a reflection azimuth set to zero (\(\beta = 0\)). The

Figure 5: The reflection azimuth angle ($\beta$) is 0°. The angle gather associated to the horizontal reflector is well defined. For steeper reflectors, the quality of the gathers is deteriorated.

Figure 6: The reflection azimuth angle ($\beta$) is 12°. It is a good estimation for the azimuth of the reflection dipping at 45° at 1,150 meters. But it is not for shallower or deeper reflectors. The best imaged reflectors are the least dipping ones. Indeed, when the reflector is close from the horizontal, the bending is not important and the rotation of the source ray with respect to the receiver ray is negligible. The azimuth displayed in Figure 6 is 12°. It means that when the rotation of the rays is 12° at the reflection point then the reflector is well imaged. For such amount of rotation, the reflector must have a reasonable dip. That is what we observe in Figure 6: the plane with a 45° dip displays a wide and sharp angle gather. In Figure 7, the azimuth selected is even bigger. It is for this amount of rotation that the angle gather for the 60° dipping reflector is the best defined.
CONCLUSION

We have presented an improved method to compute ADCIGs from ODCIGs. Our implementation demonstrates the importance of using a 3-D transformation. We have first considered a common-azimuth case in which the rays are coplanar and the source receiver orientation is parallel to the in-line direction. We have then presented the full prestack migration case in which we have imposed the rays to be coplanar at the imaging point. Non-zero azimuth reflections have also been taken into account. Our next step is to consider the more general case of non-coplanar rays at the imaging point. This situation occurs when the migration velocity is not correct.

ACKNOWLEDGMENTS

We would like to thank Paul Sava for his relevant comments and advice.

REFERENCES


Fomel, S., 1996, Migration and velocity analysis by velocity continuation: SEP–92, 159–188.


APPENDIX A

Geometrical derivation

To derive equation (2) we will change the coordinates system in order to find the 2-D case illustrated in Figure A-1-a) and follow the derivation given in (Fomel, 1996). Consider the source and the receiver rays constrained to a slanted propagation plane as illustrated in Figure A-1-b). The plane has dip angle $\delta$. Within the propagation plane, the source ray has a dip angle $\alpha_S$ and the receiver ray a dip angle $\alpha_R$. We define $z'$ as being a new vertical axis within the propagation plane: the 3-D problem becomes a 2-D problem in this new basis. The aperture angle $\gamma$ and the dipping angle $\alpha_n$ of the normal of the interface at the reflection point can be expressed as a function of $\alpha_S$ and $\alpha_R$:

$$\gamma = \frac{\alpha_R + \alpha_S}{2} \quad (A-1)$$

$$\alpha_n = \frac{\alpha_R - \alpha_S}{2} \quad (A-2)$$

The offset ray parameter in the propagation plane is:

$$\frac{\partial t}{\partial h_x} = \frac{\sin \alpha_S}{v} + \frac{\sin \alpha_R}{v} = \frac{2}{v} \cos \alpha_n \sin \gamma \quad (A-3)$$

We can write a 2-D version of the Double Square Root (DSR) in the propagation plane:

$$\frac{\partial t}{\partial z'} = \frac{\cos \alpha_S}{v} + \frac{\cos \alpha_R}{v} = \frac{2}{v} \cos \alpha_n \cos \gamma \quad (A-4)$$
The change of variable between the pseudo vertical axis $z'$ and the real vertical axis $z$ is made with the relation:

$$\frac{\partial z'}{\partial z} = \cos \delta$$  \hspace{1cm} (A-5)$$

With equations (A-4) and (A-5) the DSR expression in the original coordinates system becomes:

$$\frac{\partial t}{\partial z} = \frac{2}{v} \cos \alpha_n \cos \gamma \cos \delta$$  \hspace{1cm} (A-6)$$

The quotient of equations (A-3) and (A-6) leads to the expected relation 2:

$$\frac{\partial z}{\partial h} = \frac{k_{h_y}}{k_z} = \tan \gamma \cos \delta$$  \hspace{1cm} (A-7)$$

**Analytical derivation**

Another way to find the relation between $z$ and $z'$ starts from the cascade of two dispersion relations (Biondi and Palacharla, 1996). The first one is for 2-D prestack downward-continuation along the in-line direction,

$$k'_z = \sqrt{\frac{\omega^2}{v^2} - \frac{1}{4} (k_{m_x} + k_{h_x})^2} + \sqrt{\frac{\omega^2}{v^2} - \frac{1}{4} (k_{m_x} - k_{h_x})^2},$$  \hspace{1cm} (A-8)$$

which is equivalent to equation (A-4). The second one is for 2-D zero-offset downward continuation along the cross-line axis,

$$k_z = \sqrt{k'_z^2 - k_{m_y}^2},$$  \hspace{1cm} (A-9)$$

which is equivalent to equation (A-5). Indeed, by taking the square and dividing by $k_z$,

$$1 = \frac{k_z^2}{k'_z^2 - k_{m_y}^2}$$  \hspace{1cm} (A-10)$$

and by introducing $\tan \delta = \frac{k_{m_y}}{k_z}$,

$$\frac{\partial z}{\partial z'} = \frac{k'_z}{k_z} = \sqrt{1 + \tan^2 \delta} = \frac{1}{\cos \delta}.$$  \hspace{1cm} (A-11)$$

**APPENDIX B**

The coplanarity condition presented in (Biondi and Palacharla, 1996) is:

$$\hat{k}_{h_y} = k_{m_y} \left[ \sqrt{\frac{\omega^2}{v^2} - \frac{1}{4} (k_{m_x} + k_{h_x})^2} - \sqrt{\frac{\omega^2}{v^2} - \frac{1}{4} (k_{m_x} - k_{h_x})^2} \right]$$  \hspace{1cm} (B-1)$$

$$\sqrt{\frac{\omega^2}{v^2} - \frac{1}{4} (k_{m_x} + k_{h_x})^2} + \sqrt{\frac{\omega^2}{v^2} - \frac{1}{4} (k_{m_x} - k_{h_x})^2}.$$
Each square root in the previous equation can be substituted with the expression including the dip angle of the source or the receiver ray

\[ \sqrt{\frac{\omega^2}{v^2} - \frac{1}{4} (k_{mx} + k_{hx})^2} = \frac{\omega}{v} \cos\alpha_R \]  

(B-2)

\[ \sqrt{\frac{\omega^2}{v^2} - \frac{1}{4} (k_{mx} + k_{hx})^2} = \frac{\omega}{v} \cos\alpha_S \]  

(B-3)

where \( \alpha_S \) and \( \alpha_R \) can be expressed in terms of the aperture angle \( \gamma \) and the normal dip angle \( \alpha_n \):

\[ k_{h_y} = k_{m_y} \frac{\cos(\alpha_n + \gamma) - \cos(\alpha_n - \gamma)}{\cos(\alpha_n + \gamma) + \cos(\alpha_n - \gamma)} \]  

(B-4)

Or, after applying some trigonometric relations

\[ k_{h_y} = k_{m_y} \tan\gamma \tan\alpha_n. \]  

(B-5)

We now introduce equation (2) as well as \( \tan\alpha = \frac{k_{mx}}{k_{m_z}} \) where \( \alpha \) is the projection of the angle \( \alpha_n \) on the vertical plane passing through the source-receiver axis. The two angles are linked by the relation \( \tan\alpha_n = \cos\delta \tan\alpha \). Equation B-5 becomes

\[ k_{h_y} = \frac{k_{m_y} k_{m_z} k_{h_x}}{k_{m_z}^2 + k_{m_y}^2} \]  

(B-6)

APPENDIX C

Consider one image point in the Fourier domain, i.e. at given \( k_{mx}, k_{my} \) and \( k_z \). For one image point, we have all the values of the offset gather. Each value is referenced in the Fourier domain by the offset wavenumbers \( k_{hx} \) and \( k_{hy} \). Given a reflection azimuth \( \beta \) and an aperture angle \( \gamma \), we want to get the value of the sample associated with \( \gamma \), hence what are the offset wavenumbers associated with \( \gamma \). Given \( k_{mx}, k_{my}, k_z, \beta \) and \( \gamma \):

- Because of the reflection azimuth, we need to get into the new coordinates system. \( k'_{mx}, k'_{my} \) are computed using \( k_{mx}, k_{my} \) and \( \beta \) in equation (4).

- For a given \( \gamma \) and \( k_z \) we can determine \( k'_{h_y} \) from equation (6).

- The second component of the offset wavenumber, \( k'_{hy} \), must satisfy the coplanarity condition (7).

- The two components of the offset wavenumber were obtained in the new system of coordinates. To return to the original system of coordinates, we use the inverse of transformations (4).

- We have determined which sample of the offset gather should be associated with the aperture angle \( \gamma \).

The entire angle gather at the image point \( (k_{mx}, k_{my}, k_z) \) and at the reflection azimuth \( \beta \) is obtained by looping over \( \gamma \).
Angle decompositions of images migrated by wavefield extrapolation

Paul Sava

ABSTRACT
I present an extension to the angle-domain decomposition of images migrated using wavefield extrapolation. Traditionally, reflectivity is described by a 1-D function of scattering angle. I show that we can further decompose the image function of other angles related to the structure and acquisition. In the 2-D case, the reflectivity is described function of two angles, while in the 3-D case the reflectivity is described function of four angles. Applications for such a multi-angle decomposition include amplitude and illumination compensation due to limited acquisition.

INTRODUCTION
Angle-domain common image gathers (ADCIGs) are decompositions of seismic images in components proportional to the reflection magnitude at various incidence angles at the reflector. ADCIGs can be constructed for many types of methods, including Kirchhoff, one-way wavefield extrapolation or two-way reverse time migration. The main applications of ADCIGs are in migration velocity analysis (MVA) and amplitude versus angle analysis (AVA).

Generally speaking, there are two classes of ADCIGs: those computed in the data space, which produce images with reflectivity described as a function of offset ray parameter $p_h$ (de Bruin et al., 1990; Prucha et al., 1999) and those computed in the image space, which produce images with reflectivity function of scattering angle $\gamma$ (Weglein and Stolt, 1999; Sava and Fomel, 2003).

The mechanics of computing angle-gathers with either of these two methods are similar, since both involve slant-stacks or radial-trace transforms at various stages of the wavefield extrapolation migration, before or after imaging.

The main advantages and disadvantages of the two methods are discussed by Sava and Fomel (2003). The most important advantage of the image space method over the data space method is its versatility: the same transformation can be used for images produced by shot-geophone (S-G) migration (Sava and Fomel, 2003), shot-profile migration (Rickett and Sava, 2001), reverse-time migration (Biondi and Shan, 2002), and even for migrated images of converted waves (Rosales and Rickett, 2001). In all cases, we obtain at every point in the image

1email: paul@sep.stanford.edu
a 1-D description of the reflection magnitude function of the scattering angle ($\gamma$), measured with respect to the normal to the reflector.

However useful, this ADCIG transformation is incomplete. There are situations when by observing how reflectivity changes with the scattering angle, we cannot distinguish among totally different geologic scenarios. A striking comparison is that between a reflector and a diffractor which are both kinematically represented by flat gathers.

This paper presents a simple extension to the angle-gather transformation. By employing similar techniques as the ones used for the more traditional approach, we can create ADCIGs where reflectivity is described by two parameters in 2-D, or by four parameters in 3-D. These angles represent the structural dip, acquisition azimuth and scattering angles. Using simple examples, I show that this more complex decomposition is capable of highlighting new and useful information. Possible applications of this decomposition include illumination compensation due to limited acquisition, or dip-dependent migration amplitude corrections (Sava et al., 2001).

2-D THEORY

The derivation in this section follows the one of Fomel (1996). Assuming that a reflection event in the extrapolated wavefield is described by the function $t(z, s, r)$, we find from the Snell’s law the following derivatives:

$$\frac{\partial t}{\partial s} = \frac{\sin(\alpha - \gamma)}{v}, \quad (1)$$

$$\frac{\partial t}{\partial r} = \frac{\sin(\alpha + \gamma)}{v}, \quad (2)$$

where $s, r$ stand for the source and receiver spatial coordinates, $v$ is the wave velocity, $\alpha$ is the dip angle, and $\gamma$ is the reflection angle (Figure 1). The traveltime derivative with respect to the depth of the observation surface $z$ has contributions from the two branches of the reflected

![Figure 1: A sketch of reflection rays in an arbitrary-velocity medium.](paul3-local)
ray, as follows:

\[ \frac{\partial t}{\partial z} = \cos(\alpha - \gamma) \frac{v}{u} + \cos(\alpha + \gamma) \frac{v}{u}. \tag{3} \]

Equation 3 corresponds to the well-known double-square-root equation (Claerbout, 1985). This equation simply reflects the fact that the traveltime increases with increasing depth of the reflector.

Transforming Equations (1-3) to the midpoint and half-offset coordinates, we obtain

\[ \frac{\partial t}{\partial x} = \frac{\partial t}{\partial s} + \frac{\partial t}{\partial r} = \frac{2 \sin \alpha \cos \gamma}{v}, \tag{4} \]
\[ \frac{\partial t}{\partial h} = \frac{\partial t}{\partial r} - \frac{\partial t}{\partial s} = \frac{2 \cos \alpha \sin \gamma}{v}, \tag{5} \]
\[ \frac{\partial t}{\partial z} = - \frac{2 \cos \alpha \cos \gamma}{v}. \tag{6} \]

At a fixed image location \( x \), we can transform the derivatives of \( t(z, x, h) \) to the derivatives of \( z(t, x, h) \) by applying the implicit function theorem. Using Equations (5-6), we obtain

\[ \frac{\partial z}{\partial h} = - \frac{\frac{\partial t}{\partial h}}{\frac{\partial t}{\partial z}} = -\tan \gamma, \tag{7} \]

and using Equations (4-6), we obtain

\[ \frac{\partial z}{\partial x} = - \frac{\frac{\partial t}{\partial x}}{\frac{\partial t}{\partial z}} = -\tan \alpha. \tag{8} \]

Sava and Fomel (2003) use Equation (7) to compute angle-domain common image gathers for images obtained by wavefield extrapolation. This formula can be implemented either as a slant-stack in the space domain, or as a radial-trace transform in the Fourier domain. In both cases, we obtain at every point in the image the reflection strength as a function of scattering angle, independent of the reflector dip.

Similarly, we could employ formula (8) for another decomposition of the migrated image function of the structural dip \( \alpha \). As for equation (7) which represents a slant-stack in the \( x - h \) plane, equation (8) represents a slant-stack in the \( x - z \) plane (Figure 2).

Figure 2: Angle decompositions. Panel (a) corresponds to slant-stack angle-decomposition in the \( z - h \) plane. Panel (b) corresponds to slant-stack angle-decomposition in the \( z - x \) plane. In both cases, a dipping segment maps at a particular angle.
3-D THEORY

Equations (7) and (8) are derived with the assumption of a 2-D earth. As pointed out by Tisserant and Biondi (2003), the Fourier-domain angle-gather transformation

$$\tan \gamma = -\frac{k_h}{k_z}$$

is not valid in general in 3-D and needs to be corrected for the crossline dip component of the structure.

Tisserant and Biondi (2003) show that we can make this 3-D correction by re-writing the angle-gather transformation as

$$\tan \gamma = -\frac{|k_h|}{k_z} \frac{1}{\sqrt{1 + \left(\frac{k_{mx}}{k_z} \sin \beta + \frac{k_{my}}{k_z} \cos \beta \right)^2}}$$

where $\beta$ is the acquisition azimuth, $|k_h|$ is the absolute value of the offset wavenumber, and $k_{mx}, k_{my}$ are the components of the midpoint wavenumber. If we introduce the notation $\tan \delta_x = \frac{k_{mx}}{k_z}$ and $\tan \delta_y = \frac{k_{my}}{k_z}$, we can write

$$\tan \gamma = -\left[\frac{1}{\sqrt{1 + \left(\tan \delta_x \sin \beta + \tan \delta_y \cos \beta \right)^2}}\right] \frac{|k_h|}{k_z}.$$  

$\delta_x$ and $\delta_y$ represent the projection angles of the normal to the reflection plane onto the Cartesian coordinate system. Equation (11) gives the relation between the four angles describing the reflectivity decomposition in 3-D, namely the structural dip angles ($\delta_x, \delta_y$), the recording azimuth ($\beta$), and the scattering angle ($\gamma$).

The 3-D decomposition algorithm is based on the following scheme:

- Begin from a 3-D migrated image

$$R(z, h_x, h_y, m_x, m_y).$$

(12)
Transform it to polar coordinates in the $h_x - h_y$ plane

$$R(z, h, \beta, m_x, m_y),$$

where, by definition, $\tan \beta = -\frac{k_{h_y}}{k_{h_x}}$ and $h = \sqrt{h_x^2 + h_y^2}$.

- Decompose the image by dip and dip azimuth, or equivalently by the projections of the normal to the reflection plane on the Cartesian axes ($\delta_x, \delta_y$):

$$R(z, h, \beta, \delta_x, \delta_y),$$

where $\tan \delta_x = -\frac{k_{m_x}}{k_z}$ and $\tan \delta_y = -\frac{k_{m_y}}{k_z}$.

- At every $\beta, \delta_x, \delta_y$, decompose the image function of scattering angle $\gamma$

$$R(z, \gamma, \beta, \delta_x, \delta_y)$$

using Equation (11).

The four angles describing reflectivity are just one particular and convenient choice. Any other convenient quartet of angles can be used, although such alternatives yield no new information.

**EXAMPLES**

The first example, Figures 4 and 5, is a fairly simple synthetic model populated with reflectors at various dips and a few diffractors. One goal of this model is to demonstrate the ability of the transformation described in this paper to distinguish reflections from diffractions.

Figure 5 shows on the left a typical ADCIG. Clearly, this panel does not allow us to distinguish reflectors from diffractors. However, the dual angle decomposition presented in the right panels for two particular depths at the same location show a clear distinction between reflectors and diffractors: either the event concentrates around the dip direction (reflections), or it maps uniformly to all dips (diffractions).

Figure 6 is a summary of the multi-angle decomposition for the entire synthetic model. Each rectangle represents an image point located roughly at the same location of the rectangle. In each box, the horizontal axis corresponds to the dip angle $\alpha$, and the vertical axis corresponds to the scattering angle $\gamma$.

The second example concerns the well known Marmousi model. Figures 7 depicts the migrated image, and the vertical line indicates the location of the ADCIG under analysis. Figure 8 shows on the left a standard 1-D ADCIG, and on the right the multi-angle decomposition for two particular depths at the same location. Each one of the right panels shows events dipping in opposite directions at different angles. In this example, the dip angle decomposition is not as sharp as in the preceding model due to the limited resolution of the space-domain slant-stacks. Nevertheless, the selected events are clearly located at different structural dip angles.
The next example corresponds to a 3-D seismic image (Figure 9) obtained by narrow-azimuth migration (Biondi, 2003). The figure depicts the zero offset $h_x = 0, h_y = 0$ image, although in reality the cube has the full 5 dimensions of 3-D prestack data.

I decompose the reflectivity at $m_x = 1000, m_y = 800$ function of the four angles introduced in the preceding section. This 4-D image decomposition is practically impossible to fully illustrate on paper. A reasonable alternative is to display subsets of the decomposition, by appropriate summation over some of the angles. For example, Figure 10 shows an usual angle gather, obtained by summation over the reflector dip angles $\delta_x, \delta_y$ and acquisition azimuth $\beta$. The left panel depicts an angle gather computed using the 2-D formula, and the right panel shows an angle gather at the same location obtained by summation of the three angles after the 3-D image decomposition. As also seen by Tisserant and Biondi (2003), not much changes for the upper flat reflector, but the event for the bottom reflector shrinks according to the crossline structural dip.

The more interesting plots, however, concern other partial summations over the four an-
Figure 6: Multi-angle decomposition for the image in Figure 5. In each box, the horizontal axis corresponds to the dip angle $\alpha$, and the vertical axis corresponds to the scattering angle $\gamma$.

gles. Figure 11 shows two such partial summations: on the left, the panels correspond to summation over the scattering and azimuth angles ($\gamma$ and $\beta$); on the right, the panels correspond to summation over the projections of the normal to the reflector ($\delta_x$ and $\delta_y$). From top to bottom, each pair of panels corresponds to different depths $z = 1000, 1150, 1380$ at the same location in the image $m_x = 1000, m_y = 800$.

The panels on the left simply indicate the dip and azimuth direction of the normal to the reflection plane, measured by its projections on the Cartesian coordinate system. The right panels, however, show the reflectivity variation with scattering angle and acquisition azimuth. The energy is unevenly distributed function of scattering angle and azimuth. Furthermore, the energy decays at higher scattering angles. The effect is similar to that observed by Sava et al. (2001) for traditional 2-D angle-gathers. We can also observe amplitudes decaying with depth, which is consistent with smaller angular coverage for a fixed acquisition geometry.

CONCLUSIONS

An extension to a widely used transformation allows descriptions of migrated images in components corresponding to various dip, scattering and azimuth angles. Two angles are enough to fully describe the imaged reflectivity in the case of a 2-D experiment. In the more general 3-D case, images are described by four angles.

This image transformation has many applications, among which amplitude compensation (Sava et al., 2001) for AVA analysis, and illumination normalization for limited and irregular data acquisition. We could also imagine many other ways of filtering our migrated images in various sub-panels of the 3-D decomposition, to correct or eliminate specific events.
Figure 7: Marmousi model: migrated image.

Figure 8: Image decomposition by scattering angle $\gamma$ (left). Image decomposition by scattering angle $\gamma$ and dip $\alpha$ angle (right) at the depth levels marked on the left panel.

Finally, I emphasize that neither the standard nor this extended multi-angle ADCIG transformation add any new information that does not already exist in the migrated image. Both represent decompositions of the image in subsets that are useful and simply more convenient for different applications.

REFERENCES


Figure 9: 3-D narrow-azimuth migrated image. The figure depicts the zero offset \((h_x = 0, h_y = 0)\) image out of a cube with 5 dimensions.

Figure 10: Angle gather obtained by the traditional ADCIG equation (left). 3-D angle gather obtained by summation over the reflector dip angles \(\delta_x\), \(\delta_y\) and acquisition azimuth \(\beta\) (right).


Fomel, S., 1996, Migration and velocity analysis by velocity continuation: SEP–92, 159–188.


Sava, P., Biondi, B., and Fomel, S., 2001, Amplitude-preserved common image gathers by


![Figure 11: 3-D multi-angle decomposition of migrated images. On the left, the panels correspond to summation over the scattering $\gamma$ and azimuth $\beta$ angles. On the right, the panels correspond to summation over the projections of the normal to the reflector, $\delta_x$ and $\delta_y$. From top to bottom, each pair of panels corresponds to different depths $z = 1000, 1150, 1380$ at the same location in the image $m_x = 1000, m_y = 800$.](image-url)

[ref1] Paul3-Slic3d [ER]
Multiple realizations and data variance: Successes and failures

Robert G. Clapp

ABSTRACT
Geophysical inversion usually produces a single solution to a given problem. Often it is desirable to have some error bounds on our estimate. We can produce a range of models by first realizing that the single solution approach produces the minimum energy, minimum variance solution. By adding appropriately scaled random noise to our residual vector we change the minimum energy solution. Multiple random vectors produce multiple new estimates for our model. These various solutions can be used to assess error in our model parameters. This methodology strongly relies on having a decorrelated residual vector and, previously, was used primarily on the model styling portion of our inversion problem because it came closer to honoring the decorrelation requirement. With an appropriate description of the noise covariance, multiple realizations can be estimated. Examples of perturbing the data fitting portion of the standard inversion are shown on a 2-D deconvolution and 1-D velocity estimation problems. Results indicate that methodology has potential but is not well enough understood to be generally applied.

INTRODUCTION
Risk assessment is a key component to any business decision. Geostatistics has recognized this need and has introduced methods such as simulation to attempt to assess uncertainty in their estimates of earth properties (Isaaks and Srivastava, 1989). The problem is that the geostatistical methods are generally concerned with local, rather than global, solutions to problems and therefore can not be easily applied to global inversions problems that are common in geophysics.

In previous works (Clapp, 2000, 2001a,b), I showed how we can modify standard geophysical inverse techniques by adding random noise into the model styling goal to obtain multiple realizations. In Clapp (2002) and Chen and Clapp (2002), these multiple realizations were used to produce a series of equiprobable velocity models. The velocity models were used in a series of migrations, and the effect on Amplitude vs. Angle (AVA) was analyzed. In previous papers on the subject, the concentration was on modifying the model styling goal and only briefly mentioned that it should be feasible to apply the same methodology to the data fitting goal.

In this paper, I show the data fitting goal can also be changed to produce equiprobable

1email: bob@sep.stanford.edu
results. I show that it is a much more difficult problem than modifying our model styling goal because of the difficulties in building a realistic noise covariance operator. I begin by reviewing the methodology of multiple realizations with operators. I introduce a simple inverse filtering example to illustrate that for simple cases we can create pseudo-datasets with realistic noise distribution. These datasets can be used as input to an inversion process to get some preliminary boundaries on model errors. I conclude by showing an example of creating multiple reasonable interval velocity models from a single velocity scan.

**REVIEW**

Inverse problems obtain an estimate of a model \( m \), given some data \( d \) and an operator \( L \) relating the two. We can write our estimate of the model as minimizing the objective function in a least-squares sense,

\[
 f(m) = \|d - Lm\|^2. \tag{1}
\]

We can think of this same minimization in terms of fitting goals as

\[
 0 \approx r = d - Lm, \tag{2}
\]

where \( r \) is a residual vector.

Bayesian theory tells us (Tarantola, 1987) that convergence rate and the final quality of the model is improved the closer \( r \) is to being Independent Identically Distributed (IID). If we include the inverse noise covariance \( N \) in our inversion our residual becomes IID,

\[
 0 \approx r = N(d - Lm). \tag{3}
\]

A regularized inversion problem can be thought of as a more complicated version of (3) with an expanded data vector and an additional covariance operator,

\[
 0 \approx r_d = N_{\text{noise}}(d - Lm) \quad 0 \approx r_m = \epsilon N_{\text{model}}(0 - Im). \tag{4}
\]

In this new formulation

\( r_d \) is the residual from the data fitting goal,
\( r_m \) is the residual from the model styling goal,
\( N_{\text{noise}} \) is the inverse noise covariance,
\( N_{\text{model}} \) is the inverse model covariance,
\( I \) is the identity matrix, and
\( \epsilon \) is a scalar that balances the fitting goals against each other.
Normally we think of $N_{\text{model}}$ as the regularization operator $A$. Simple linear algebra leads to a more standard set of fitting goals:

$0 \approx r_d = N_{\text{noise}}(d - Lm)$
$0 \approx r_m = \epsilon Am$.  

(5)

The problem with this approach is that we never know the true inverse noise or model covariance and therefore are only capable of applying approximate forms of these matrices.

Model Variability

What we choose for $A$ can have a significant affect on our model estimate. In theory $A$ should be a matrix of size $nm$ by $nm$ (where $nm$ is the number of model elements). People use numerous approximations for $A$. Some of the more common

• a Laplacian or some type of symmetric operator,
• a stationary Prediction Error Filter (Claerbout, 1999),
• a steering filter (Clapp, 2001a), or
• a non-stationary PEF (NSPEF) (Crawley, 2000).

The first option makes the assumption that our model is smooth and stationary. The second option still assumes stationarity, but allows for much more sophisticated covariance descriptions. The third option allows for non-stationarity but is only valid when a model has a single dip at each location and requires some \textit{a priori} knowledge about the model. The last is the best representation of the inverse model covariance matrix. Unfortunately, it requires a field with the same properties as the model in order to be constructed. In addition, it faces stability problems (Rickett, 2001).

At least the first three methods, and possibly all four (depending on the field we use and the filter description we choose to estimate the NSPEF) are limited to describing second order statistics. In addition we normally try to describe the inverse covariance through a filter with only a few coefficients. In terms of the covariance matrix, we are putting non-zero coefficients along only a few diagonals. Finally, all of these approaches assume that the main diagonal of the inverse covariance matrix is a constant.

The problems with these approximations are demonstrated in Figure 1. The left panel shows measurements of the sea depth for a day. A PEF is estimated at the known locations and used as $A$ with fitting goals,

$0 \approx r_{\text{noise}} = d - Jm$
$0 \approx r_{\text{model}} = \epsilon Am$. 

(6)

where $d$ is shown in the left panel of Figure 1 and $J$ is a selector matrix (1 at known locations, 0 at unknown). The center panel of Figure 1 shows the estimated model. Note how the estimated
Figure 1: The left panel shows the result of recording the sea depth for one day. The center panel shows the interpolation result using a filter estimated from the known portion of the data and then solving (6). Note the difference in variance in the known and unknown portions of the model. The right panel is the filter response of the PEF used for interpolation.

portion of the model doesn’t have the right texture. The variance of the model is not the same as the variance of the data. This is due to the combination of two factors.

- Our covariance description has a limited range. The right panel of Figure 1 show the result of applying the inverse PEF (or more correctly \((A^{-1})'A^{-1}\)) to a spike in the center of the model. Note how the response tends towards zero.

- Our inverse problem will give us a minimum energy solution, which means it is going to want to fill the residual vectors with as small numbers as possible.

In previous papers (Clapp, 2000, 2001a), I showed how we can change what the minimum energy solution will be by introducing an initial condition to our residual vector filled with random numbers. In terms of our fitting goals (6) we are replacing the zero vector of our model styling fitting goal with with standard normal noise vector \(\mathbf{n}\), scaled by some scalar \(\sigma_m\),

\[
0 \approx \mathbf{r_{noise}} = \mathbf{d} - \mathbf{Jm} \quad (7)
\]

\[
\sigma_m \mathbf{n} \approx \mathbf{r_{model}} = \epsilon \mathbf{Am}.
\]

For the special case of missing data problems, where \(\mathbf{L}\) is simply a masking operator \(\mathbf{J}\) delineating known and unknown points, Claerbout (1999) showed how \(\sigma_m\) can be approximated by first estimating the model through the fitting goals in (6), then by solving

\[
\sigma_m = \frac{1' \mathbf{Jr_{model}}^2}{1' \mathbf{J1}}, \quad (8)
\]

where \(\mathbf{1}\) is a vector composed of 1s. Figure 2 shows the model using three different \(\mathbf{n}\) vectors. Note how the variance of the model is now similar to the variance of the data. A good check of this is whether the original recording path can be seen.

Using equation (8) to estimate \(\sigma_m\) makes the assumption that our covariance description is correct. When our assumptions on stationarity are incorrect or filter shape is insufficient to correctly describe the covariance, this approach will be ineffective.
Data Variability

As fitting goals (4) indicate, our fitting goals are in many ways symmetric. As a result, we should be able to replace the zero vector of our data fitting goal with another scaled random vector. We now have a second scaling factor $\sigma_d$ and a new set of fitting goals:

$$\sigma_d \eta \approx r_d = N_{\text{noise}}(d - Lm)$$
$$\sigma_m \eta \approx r_m = \epsilon Am.$$  \hspace{1cm} (9)

This new set of fitting goals offers the opportunity to see how the uncertainty in our data estimates affects our uncertainty in our model estimate. In the next section I will discuss some of the difficulties in making this formulation practical.

PROBLEMS

Getting fitting goals (9) to work correctly requires a good handle on several elements that are not necessarily easy or even possible to achieve. To demonstrate these problems, I am going to set up a simple deblurring inversion problem. The left panel of Figure 3 shows the model $m$ that we are going to attempt to invert for. The right panel of Figure 3 shows the data $d$, the model blurred by a simple known filter. If we add random noise to the problem (left panel of Figure 4), we can quickly get an unreasonable model estimate (right panel of Figure 4). If we add an isotropic regularizer, our estimate improves substantially (Figure 5). We will speed up the conversion of our problem by preconditioning the problem $A^{-1}$ with a symmetric operator and start with

$$0 \approx r_d = N_{\text{noise}}(d - LA^{-1}p)$$
$$0 \approx r_m = \epsilon p.$$  \hspace{1cm} (10)

as our fitting goals.
Figure 3: The left panel is the model that we are going to attempt to invert for. The right panel is the data, a blurred version of the model.

Figure 4: The left panel is the data with Gaussian random noise added. The right panel is the resulting model estimate.

Figure 5: The model estimated from the data shown in the left panel of Figure 4 using an isotropic regularization operator.
IID Residuals

A more difficult, but at times more realistic, challenge is when our noise is not Gaussian. The left panel of Figure 6 shows smoothed Gaussian noise, of approximately the same amplitude as the random noise used above, that we will add to our data (right panel of Figure 3). The resulting model estimate (right panel of Figure 6) shows a clear imprint of the noise pattern. If we look at the $r_d$ vector (left panel of Figure 7), we see that we have disobeyed both of the IID requirements. The residual is definitely correlated and if we look at histogram of the residuals (right panel of Figure 7), the identically distributed requirement looks suspect.

Figure 6: The left panel shows correlated noise that we are going to add to our data. The right panel shows the resulting inversion. Note how the imprint of the noise is visible in the model.

Figure 7: The left panel shows the residual ($r_d$) associated with the model in Figure 6. The right panel is a histogram of the residual. Note the structure in the residual and the deviation from Gaussian of the residual.

Up to this point we have ignored the inverse noise covariance matrix $N$. Claerbout (1999) and Guitton (2000) suggested using a PEF for $N$ estimated from the residual. If we do this we
get an improved result (left panel of Figure 8) and a more decorrelate residual (right panel of Figure 8). Remember that $N$ should be the inverse of the noise spectrum. Estimating it from the residual does not necessarily give us the same information. Since this is a synthetic example, we have another option. We can estimate $N$ directly from our noise estimate. Figure 9 shows the result of the inversion using this filter and the $r_d$ vector associated with the inversion. The resulting estimate is generally of higher quality.

Figure 8: The left panel shows the inversion result using a PEF estimated from the residual as $N$. The right panel shows the resulting $r_d$. Note the improved result compared to Figure 7.

Figure 9: The left panel shows the inversion result using a PEF estimated from the known noise as $N$. The right panel shows the resulting $r_d$. Note the improved result compared to Figure 7. Compared to Figure 8 we see an improvement in image clarity at some expense of a boosting of the noise.

Types of noise

Looking at the residual in Figures 7-9 demonstrates another problem with satisfying the requirements for multiple realizations to work correctly. There are two different structures that
we see in the residual. The first is the low spatial frequency structure that is associated with
the noise that we added to the original data. This is the actual noise in our problem and what
we hope to attempt to exploit when doing multiple realizations.

The second type of error has a higher frequency component, the outline of the objects
in the photo are an example of this type of noise. This is a more disturbing component of
the residual. It is actually caused by our problem formulation. Our regularization operator
\( A \) makes an assumption of model smoothness that isn’t valid everywhere. Our data and data
fitting operator \( L \) indicate that the model should not be smooth at these locations. These two
conflicting pieces of information result in additional structure in our residual. If we estimate
our noise covariance operator directly from our data residual, the resulting noise covariance
operator will attempt to represent both types of noise and we will be introducing additional
unwanted structure into our noise when doing multiple realizations.

**Inverse noise covariance**

If we apply fitting goals (9) in an attempt to create multiple models with about the same level
and type of noise, we obtain the models seen in Figure 10. These models have about the
right amplitude of noise but have a higher frequency noise component than our synthetic test.
The reason for this discrepancy is that the noise filter does not effectively describe the noise
spectrum in our synthetic. Figure 11 shows the spectrum of the noise added in our synthetic
test (the noise seen in the left panel of Figure 6). Note how the spectrum outside a very
small band is zero. Emulating this behavior with a filter is nearly impossible and requires a
prohibitively large filter. As a result, we are limited in what type of noise we can describe by
the practicality of building a filter that accurately captures its spectrum.

**Variance**

In the above example, the assumption that the variance of our noise is spatially invariant is
correct. In a more general problem, this will not be the case. As a result, a more appropriate
formulation for our noise covariance is

\[
N = N_c N_v. \tag{11}
\]

In this formulation, \( N_c \) is a description of the relation between points. It will take the form of
a PEF or one of the other operators described above. The second operator \( N_v \) will attempt to
normalize the variance of the various data errors to the same level and should be equal to

\[
N(i) = \sqrt{\frac{1}{V(i)}}, \tag{12}
\]

where \( V(i) \) is the variance at a given data location \( i \).
Figure 10: Four realizations using the fitting goals (9). Note how the amplitude of the noise is consistent with our synthetic experiment but the spectrum shows more energy and large wavenumbers. [bob5-inc] [ER,M]

Figure 11: The spectrum of the noise added in the synthetic problem shown in Figures 6-9. [bob5-noise] [ER]
A final problem is choosing appropriate values for $\sigma$ and $\epsilon$. Often we have a good idea of what our data variance is, but what we are actually adding random noise to is in the output space of our inverse noise covariance operator. How the variance in the data space translates to a variance in our noise covariance operator is far from obvious. In addition, if our $\sigma$ value is different from the variance of solving our estimation without noise added to the residual, we will need to modify our value of $\epsilon$ to achieve the same balance between our data fitting and model styling goal.

### 1-D SUPER DIX

A relatively simple, but more realistic, example is estimating interval velocities $v_{\text{int}}$ from RMS velocities $v_{\text{rms}}$. Clapp et al. (1998) did this by taking advantage of the linear relation between $v_{\text{rms}}^2$ and $v_{\text{int}}^2$. We can keep our interval velocities relatively smooth by adding a roughening operator $D$. The fitting goals then become

$$
0 \approx r_n = T v_{\text{rms}}^2 - C v_{\text{int}}^2 \\
0 \approx r_m = \epsilon D v_{\text{int}}^2,
$$

where $C$ is causal integration and $T$ is the result of causal integration with a vector of ones.

Figure 12 shows the result of applying this procedure on a simple CMP gather. The left panel shows the initial CMP gather and the center panel shows the stack power of various $v_{\text{rms}}$ values. We use the maximum within a reasonable fairway (the solid lines overlaying the stack power scan) as our data (dashed lines). The right panel of Figure 12 shows: our auto-picked $v_{\text{rms}}$ (solid line), our inverted $v_{\text{int}}$ (dashed lines), and our interval velocity converted back to RMS velocity (dotted line).

Fitting goals (13) again assume a constant variance in our data. This is an incorrect assumption in this case for two very obvious regions. First, the $T$ operator applied to our data means that late times are going to be given a much larger weight in our inversion. A solution to this problem is introduce a weighting operator $D_1$, which is simply $\frac{1}{T}$. A second error in the assumption of constant variance is that we know that not all our data ($v_{\text{rms}}$ measurements) are of the same quality. The center panel of Figure 12 shows that there are areas where there are no significant reflectors. In addition, there are areas where our stack power results show an obvious maximum at a given $v_{\text{rms}}$ value and other areas where the maximum is much less clear. To try to take into account both of these phenomena I calculated a weighted variance within the fairway shown in the center panel of Figure 12,

$$
v(i) = \frac{\sum_{j=b(i)}^{\sigma(i)} (v(j) - v_{\text{max}}(i))^2 s(i,j)^4}{\sum_{j=b(i)}^{\sigma(i)} s(i,j)^4},
$$

where
Figure 12: The left panel shows the initial CMP gather. The center panel shows the stack power of various $v_{\text{rms}}$ values. Overlaid is a fairway (the solid lines overlaying the stack power scan) that we used for automatically picking RMS values (dashed lines). The right panel of Figure 12 shows: our auto-picked $v_{\text{rms}}$ (solid line), our inverted for $v_{\text{int}}$ (dashed lines), and our interval velocity converted back to RMS velocity (dotted line).

$b(i)$ is the beginning sample of the fairway at a given sample $i$, 
$e(i)$ is the ending sample of the fairway at a given sample $i$, 
$v(j)$ is the $v_{\text{rms}}$ at a given stack power location, 
$v_{\text{max}}(i)$ is the velocity associated with the maximum stack power value (our data), and 
$s(i,j)$ is the semblance value at time sample $i$ and some $v_{\text{rms}}$ value $j$.

The left panel of Figure 13 shows our stack power scan overlaid by $v_{\text{rms}}$ (dashed line) and $v_{\text{rms}} + -\sqrt{v}$ (dashed lines). Note how at areas with a sharp stack power blob the variance is small, while when the stack power blob is wide, where we have little coherent energy, the variance is large. We can now estimate new interval velocity model using,

$$0 \approx r_n = N(T v_{\text{rms}}^2 - C v_{\text{int}}^2)$$

$$0 \approx r_m = \epsilon D v_{\text{int}}^2.$$  \hspace{1cm} (15)

where $N$ is $V D_1$. The right panel of Figure 13 shows our data $v_{\text{rms}}$, $v_{\text{int}}$, and $C v_{\text{int}}$. 
Multiple \( v_{\text{int}} \)

Now we have almost all of the pieces to create multiple realistic interval velocity models from our \( v_{\text{rms}} \) measurements. The left panel of Figure 14 shows the residual after estimating the model using (15). We can see that there is a low frequency component to the residual. We can estimate a PEF from this residual and resolve (15) using

\[
N = HVD_1, \tag{16}
\]

where \( H \) is convolution with our PEF calculated from the residual. The resulting residual (right panel of Figure 14) is now white.

We can now use our multiple realization machinery to produce multiple interval velocity models that have approximately the correct covariance. The right panel of Figure 15 shows 80
interval velocity realizations. The left panel shows those realizations converted back to $v_{\text{rms}}$. Note how the $v_{\text{rms}}$ functions stay within variance bounds. In addition, as we expected, we see larger variance where our bounds are wider. If we look at the mean and variance of our interval and rms (Figure 16) estimates we get another interesting result. Areas of large variance are not always correlated.

Figure 15: The right panel shows 80 interval velocity realizations. The left panel show those interval velocities converted back to $v_{\text{rms}}$ overlaying the stack power for the CMP gather.

Figure 16: The left panel shows the mean of the $v_{\text{rms}}$ (solid line) and $v_{\text{int}}$ (dashed line) of 80 realizations overlaying the stack power. The right panel shows the variance of $v_{\text{rms}}$ and $v_{\text{int}}$ for the 80 realizations.

**CONCLUSIONS**

In order to see the effect of data uncertainty upon model uncertainty, we must have a good understanding of the properties of our noise and the accuracy of our modeling and regularization
operators. If we are able to make reasonable estimates on these properties we can produce multiple, equi-probable estimates. This methodology shows great promise in fields like velocity analysis where we understand the errors in our data but its complex interaction with the model makes inferring model uncertainty difficult. Preliminary work using a simple RMS to interval velocity estimation shows promise.

REFERENCES


Flattening 3-D data cubes in complex geology

Jesse Lomask

ABSTRACT

3-D volumes of data can be efficiently flattened with Fourier domain methods as long as the reflections are continuous and depth invariant. To handle faults (discontinuous reflections), I pose the flattening problem in the time-space (T-X) domain to apply a weight. This ignores fitting equations that are affected by the faults. This approach successfully flattens a synthetic faulted model. Also, the flattening method is applied repeatedly in the T-X domain to flatten a synthetic model that has pinch-outs and structure that varies with depth. There are two possible schemes for handling unconformities. One scheme requires that the unconformity be picked, the data separated into different volumes, flattened individually, and then recomposed. Another scheme is to apply the flattening method which picks travel-times for all horizons at once without being restricted to time-slices. It is expected that this method will be much more computationally intensive but will require less initial picking. Both of these methods need more development and testing.

INTRODUCTION

The flattening method described in Lomask and Claerbout (2002) to automatically flatten entire 3-D seismic cubes with minimal picking, has major difficulties with faults, unconformities, and pinch-outs. The original method resolves local dips into time shifts via a least-squares problem that is solved quickly in the Fourier domain. This is very similar to an approach used by Bienati et al. (1999a,b); Bienati and Spagnolini (2001, 1998), yet here the full data volume is flattened at once. This approach works efficiently with unfaulted and depth invariant reflections. However, if the data contains discontinuities caused by faulting then it has trouble. The dip at the faults are estimated incorrectly and these incorrect dip values are integrated along with the rest of the correct dip values to cause significant errors. Because this method integrates dips along time-slices, it has difficulties flattening cubes where the structure varies with depth. Flattening pinch-outs and unconformities is also problematic because the flattening solution can be non-unique as two horizons maybe compressed into one by flattening.

In this paper, I review the basic flattening process. I also present a time-distance (T-X) least-squares approach for flattening faulted seismic data cubes. Using the T-X domain rather than the Fourier domain permits the application of a weight to throw out fitting equations affected by bad dip estimates at faults. I also discuss solutions for handling pinch-outs. This flattening method can be applied repeatedly to handle the common case of depth-varying
structure. Lastly, I review methods to flatten unconformities. One method proposed by Sergey Fomel does not restrict the integration of dips to time-slices. Although this may be considerably more computationally expensive, this method might be able to flatten data with unconformities with less picking.

**REVIEW OF BASIC FLATTENING METHODOLOGY**

The basic idea is similar to phase unwrapping (Claerbout, 1999), but instead of summing phase differences to get total phase, dips are summed to get total time shifts that are then used to flatten the data. To apply the shifts, the central trace is held constant as a reference and all other traces are shifted vertically to match it.

The first step is to calculate dips everywhere in the 3-D seismic cube. Thus far only a simple dip estimation method has been used. Dip can be easily calculated using a plane-wave destructor as described in Claerbout (1992). For each point in the data cube two components of dip, \( p_x \) and \( p_y \), are estimated in the \( x \) direction and \( y \) direction, respectively. These can be represented everywhere on the mesh as vectors as \( p_x \) and \( p_y \). For the dip in the \( x \) direction of a seismic cube with a wave field represented by \( u(x, y, t) \), I calculate the following locally:

\[
p_x = -\frac{u_x \cdot u_t}{u_t \cdot u_t}, \tag{1}
\]

where \( u_x \) is a vector with components \( \partial u / \partial x \) taken on a mesh in \((x,t)\) and \( u_t \) is a vector with components \( \partial u / \partial t \). Because I am calculating local dips, it is necessary to smooth the dips. I apply a triangle filter to both the numerator and denominator of equation (1). Since the dip estimation technique described in equation (1) is very dependent on the amplitude from trace-to-trace, amplitude variations along the horizons will create inaccuracies in dip estimation. This will, in turn, affect the quality of the flattening result. Therefore, an improved dip estimator will be very beneficial. For example, plane wave destructor filters (Fomel, 2001) will provide more robust dip information as they are less sensitive to aliasing and may require less smoothing than the dip estimation technique in equation (1).

The most basic flattening approach is to integrate dips on each time-slice in the data cube to get total time shifts \((t = t(x, y, z))\). The gradient \((\nabla = (\partial, \partial, \partial))\) of the time shifts can be related to the estimated dip \((p = (p_x, p_y))\) in an overdetermined system with the following regression:

\[
\nabla t \approx \begin{bmatrix} \frac{\partial t}{\partial x} \\ \frac{\partial t}{\partial y} \end{bmatrix} \approx \begin{bmatrix} p_x \\ p_y \end{bmatrix}, \tag{2}
\]

The dips are summed to find a total time shift vector using:

\[
\nabla^2 t \approx \nabla' p \tag{3}
\]

where \( \nabla^2 \) is the Laplacian and \( \nabla' \) is the divergence. Solving this equation for time-slices in both the Fourier domain and time-space domain is the basic flattening method described in the following sections.
Integrating dips in the Fourier domain

Using the integration method described below (Lomask and Claerbout, 2002), I first apply the divergence (\(\nabla'\)) to the dip (\(p\)). I then convert to Fourier space where I integrate twice by dividing by the Laplacian. Then I convert back to the time domain. The resulting \(t\) can be thought of as the time shifts to apply to each point in the data to flatten it.

The flattening corrections for a 3-D volume can be generated in two different ways. In 2-D integration, the dips across each time-slice are integrated separately. In 3-D integration, the dips for all slices are integrated at once.

**Integrating time slices independently:** Beginning with input dip information across each horizon I have:

\[
p = (p_x, p_y).
\]  

(4)

The analytical solution to equation (3) is found with:

\[
t \approx \text{FFT}_2^{-1}\left[\frac{\text{FFT}_2[\nabla'p]}{-Z_x^{-1} - Z_y^{-1} + 4 - Z_x - Z_y}\right]
\]  

(5)

where \(Z_x = e^{iw\Delta x}\) and \(Z_y = e^{iw\Delta y}\).

The chief stumbling block for this approach is that the zero frequency component is neglected from the denominator of equation (5). This means that each time slice has a constant shift applied to it. It works out that this shift is equal to the average absolute time value. The danger here is if there is any noisy dip values in one slice not present in adjacent slices, the time correction to flatten a volume of data may actually swap data values. One way to prevent this is to regularize in 3-D by integrating all time-slices at once. Another way would be to adequately smooth the dip values.

**Integrating all time slices at once:** Beginning with my input dip data:

\[
p = (p_x, p_y, p_t)
\]  

(6)

where \(p_t\) is all ones for smoothness in time (explained below).

The analytical solution is found with:

\[
t \approx \text{FFT}_3^{-1}\left[\frac{\text{FFT}_3[\nabla'p]}{-Z_x^{-1} - Z_y^{-1} - Z_t^{-1} + 6 - Z_x - Z_y - Z_t}\right]
\]  

(7)

where \(Z_x = e^{iw\Delta x}\), \(Z_y = e^{iw\Delta y}\), and \(Z_t = e^{iw\Delta t}\).

The denominator is the Z-transform of the 3-D Laplacian. The zero frequency term of the Z-transform of the denominator is neglected. This means that the resulting surface in space will have an unknown constant shift applied to it. However, by adding the \(t\) dimension and assuming the gradient in the \(t\) direction to be all ones, I am ensuring that the integrated time varies smoothly in the \(t\) direction.
Integrating in three dimensions enforces vertical smoothness. In this case, the dip in the $t$ direction is all ones. However, it isn’t necessary to use ones. If zeros were used instead then this would ensure that the time-shifts do not change much in the $t$ direction. By integrating in 3-D, I prevent my method from swapping sample positions in time. Unfortunately, by preventing swapping of sample values this method cannot flatten data with overturned reflections. As will be discussed later, having a smooth output is beneficial if I plan to iterate on the result to remove any residual structure caused by dip changing with depth.

A major drawback to this approach rather than the 2-D approach is it involves taking a 3-D FFT. This is more computationally taxing and may restrict the size of the data volumes being integrated.

**FAULTS**

To handle faults, I will have to leave the Fourier domain behind. The Fourier based method will estimate erroneous dips across faults. It will then try to honor these erroneous dips creating a result that behaves erratically. However, in the time-space domain, I should be able to handle all faults that have at least one half of the fault tip-line within the data cube. My approach is to create a masking operator ($W$) that will throw out dip estimates along faults. The method will try to remove all deformation except at the faults where it will allow complete slippage.

I want to find the time shifts, $t(x,y)$, such that their gradient ($\nabla$) is the dip, $p(x,y)$. This sums across time-slices and is similar to equation (5). A time-space equivalent of equation (7) has also been implemented. I assume the dip, $p(x,y)$, is not a function of the unknown $t(x,y)$ and write the fitting goal:

$$\nabla t \approx p.$$

This is multiplied by the masking operator ($W$) to throw out fitting equations at the faults as:

$$W\nabla t \approx Wp.$$

The time shifts ($\hat{t}$) can be found in a least-squares sense with:

$$\hat{t} \approx (\nabla W^2 \nabla)^{-1} \nabla W^2 p.$$

**Faulted model - test case**

The synthetic model seen in Figure 1 has a fault starting from the center. The dip is constant with depth. Surfaces within this model gradually climb in the clockwise direction, much like a parking garage. This model is flattened using a Fourier space method in Figure 1. Because of erroneous dips at the fault, it does a poor job of flattening. Figure 2 shows the results of applying the T-X space approach using conjugate gradients and a weight that throws out fitting equations at the fault as in equation (10). Notice it is now well flattened. Similarly, using the 3-D integration version in T-X space, shown in Figure 3, the results are adequately flattened.
Figure 1: Faulted model. Left is unflattened. Right is flattened. It does a poor job because it calculates erroneous dips at the fault.

Figure 2: Same as Figure 1 except solved in T-X space rather than Fourier. This allows a weight to be applied to remove fitting equations corrupted by bad dips at the fault. Notice it does a much better job at flattening.
Although completely separated fault blocks present an obvious challenge to this method because the dip is unknown at fault discontinuities, I have some plausible solutions. The fault slip displacement would be the perfect dip information at a fault. If you have a fault block that is bounded on all sides by faults within the 3-D cube, then there is really no way to remove the deformation across the faults without at least knowing two points that correlate on each side of the fault. From an interpretation standpoint it will be necessary to restore both sides of the fault to see the best geological picture. Under such circumstances it maybe necessary to first calculate the fault contours of the fault using a method similar to that described in Lomask (2002) and put the values of the fault slip into the dip cube as input into the flattening process. This fault contour procedure basically just cross-correlates both sides of each fault.

The weights applied to the residual, equation (9), to throw out fitting equations at faults and to weaken fitting equations with low dip semblance or high noise can be created from rough fault models or coherency cubes. Additionally, this method still can be integrated with automatic fault tracking schemes. The output from these programs can be passed to my flattening scheme and used as fault weights. The fault contours, which are an easy output from the flattening method, can be used to quality check and improve the automatic fault tracking output.

**PINCH-OUTS, DIP IS CONSTANT WITH DEPTH**

Pinch-outs can be thought of as a single horizon that splits into a double horizon. If the reference trace for flattening goes through the single horizon, then the shifts from the flattening process should collapse the double horizon into one and therefore the solution is non-unique. On the other hand, if the reference trace goes through the double horizon, then the flattening
process should result in replicating or stretching the single horizon into two.

**Warped pinch-out model - test case**

This pinch-out model was created from a series of flat reflectors convolved with wavelets that gradually increase in dominant frequency from one corner of the model to the other. These flat reflectors were then warped with a cosine function. The 2-D and 3-D flattening results are shown in Figures 4 and 5, respectively. Here, again, both 2-D [equation (5)], and 3-D [equation (7)], integration methods do a similar job. There still is some residual warping in both. Since these flattening methods are operating on time-slices, the subtle pinch-out shifts are too small to be removed with one pass of this method. Also, the smoothing of the dips reduces sensitivity to the pinch-out.

In summary, the pinch-outs in this model are not severe enough to determine how these flattening methods deal with the issues non-unique and replicating solutions. These issues need further evaluation with models that have more dramatic pinch-outs.

![Figure 4: Pinch-out model warped by a cosine function. Left is unflattened. Right is flattened using the 2-D integration method described in equation (5).](jessel-pinchout_cos.flat_comp2D.png)
Thinning model - test case

This pinch-out model was created by taking planar reflection coefficients and gradually increasing dip with depth. The reflection coefficients were then convolved with the same wavelet. This creates the numerous pinch-outs seen in Figure 6a. This model should cause inaccuracies in the flattening method because I assume dip is not a function of time. Surprisingly, both methods in equations (5) and (7) are able to remove most of the deformation in one iteration. The results of one iteration of equation (7) is shown in figure 6b. To implement this iteratively, it is necessary to use equation (9) to apply weights where gaps were created by the previous flattening iteration.

Since the structure is changing with depth, one iteration will not necessarily provide sufficient information for flattening the data. This is mainly because I am assuming that dip is not a function of time by integrating on only time-slices. In this case, the output of the flattening method will need to be flattened again, possibly several times. Repeated application will also improve deficiencies in the flattening caused by inaccuracies in the dip estimation.

Figure 6 shows the results of five iterations of the flattening method. The image shrinks with each subsequent iteration because the weight is growing to account for the smoothing of dips. This can be rectified by adding a regularization term to the fitting goal in equation (9) to fill empty spaces. Notice that after five iterations, the horizontal view is almost all white indicating that it is flat.
Figure 6: (a) The thinning model. (b) One iteration of flattening applied. (c) Two iterations. (d) Three iterations. (e) Four iterations. (f) Five iterations. Notice that the horizontal view is almost all white indicating that it is flat.
UNCONFORMITIES

Angular unconformities present a major challenge for this methodology. An angular unconformity is a horizon-like feature that has two dips calculated everywhere. One dip for the layers above and another dip for the layers below. These can be handled in different ways.

Perhaps the simplest way to handle unconformities is to break up the cube. If an unconformity is already identified, then the data can be broken up into different cubes, flattened separately, and then recombined.

Analytical solution for calculating time shifts

Another possible solution would be to solve the following equation to find the absolute time \( t(x,y) \) that minimizes (Fomel, 2002, personal communication)

\[
J(t) = \int \int \left[ \left( p_x(t,x,y) - \frac{\partial t}{\partial x} \right)^2 + \left( p_y(t,x,y) - \frac{\partial t}{\partial y} \right)^2 \right] dx \ dy
\]

(11)

where \( p_x \) is the dip in the \( x \) direction and \( p_y \) is the dip in the \( y \) direction.

Although this is for one horizon, it will be useful to solve this equation for all horizons simultaneously. This would allow the problem to be regularized and its solution to be truly global. In particular, it would no longer be necessary to iterate to remove residual structure in cases of the dip changing with time.

The Euler-Lagrange equation is used in calculus of variations (Farlow, 1993) to find a function \( t \) that will minimize a functional \( J \). It is analogous to the calculus expression for finding \( x \) to minimize the function \( f(x) \) in:

\[
\frac{\partial f(x)}{\partial x} = 0.
\]

(12)

The Euler-Lagrange equation is applied to equation (11) to find (a simplified 2-D development is shown in Appendix A):

\[
\frac{\partial^2 t}{\partial x^2} + \frac{\partial^2 t}{\partial y^2} = \frac{\partial p_x}{\partial x} + \frac{\partial p_y}{\partial y} + \frac{1}{2} \frac{\partial (p_x^2 + p_y^2)}{\partial t}.
\]

(13)

Figure 7 is a 2-D example summarizing this flattening process for a single horizon. The 2-D version of equation (13) is:

\[
\frac{\partial^2 t}{\partial x^2} = \frac{\partial p_x}{\partial x} + \frac{1}{2} \frac{\partial p_x^2}{\partial t}.
\]

(14)

Figure 7a is a cartoon image of the horizon to be flattened. Figure 7b shows the dip field of that horizon. Figure 7c is the output time field. In this case, we are looping over the output field. Each value in Figure 7c contains time values of where to sample the data in order to flatten it. Figure 7d shows the time shift values from the output time field for this horizon.
Equation (13) is non-linear and therefore tricky to solve because dip functions, $p_x$ and $p_y$, are dependent on the unknown, $t$, and the last term is taking a derivative with respect to the unknown, $t$.

Incidentally, dropping the last term from equation 13, the equation can be simplified to:

$$\frac{\partial^2 t}{\partial x^2} + \frac{\partial^2 t}{\partial y^2} = \frac{\partial p_x}{\partial x} + \frac{\partial p_y}{\partial y}. \quad (15)$$

This is another way of writing equation (3), except that here the dip is a function of the output, $t$. Repeated applications of this equation partially accommodates for the dropping of the last term.

CONCLUSIONS

Overall, the ability of these flattening schemes to flatten 3-D volumes of data in complex geology is steadily developing. A method for handling faults has been defined but needs more testing. The issues related to unconformities and pinch-out are being addressed but still need more development and testing.

The T-X domain approach was able to flatten the faulted model. This was possible because half of the fault’s tip-line was encased within the data cube. If the fault were to cut all the way
across the data cube then it would be necessary to first calculate the slip distribution along the fault and use the slip values as dips.

Multiple iterations of the flattening method were able to remove the residual structure from the test case with structure varying with depth. The data cube was essentially flat after four iterations. The anticipated issues of multiple data points mapping into a single point associated with pinch-outs have not yet been addressed because the dip becomes too small to be sensitive to this.

The two possible solutions for dealing with angular unconformities, need to be tested. The first solution is to merely identify the unconformity, break the data into different cubes, flatten separately, and then recombine. The second solution requires solving a challenging differential equation but may require less picking.

The present procedure for applying the shifts to cubes is very simplistic and needs enhancement. The current approach is to hold the central trace constant and shift all other traces vertically to match it. However, a procedure for actually applying the shifts in the presence of numerous non-vertical faults, over-turned beds, pinch-outs, and unconformities needs to be developed.

ACKNOWLEDGMENTS

I would like to thank Jon Claerbout and Sergey Fomel for many useful insights.

REFERENCES


**APPENDIX A**

**Euler-Lagrange solution for flattening**

The Euler-Lagrange equation can be used to find the absolute time \( t(x,y) \) that minimizes (Fomel, 2002, personal communication)

\[
J(t) = \int \int \left[ (p_x(t,x,y) - \frac{\partial t}{\partial x})^2 + (p_y(t,x,y) - \frac{\partial t}{\partial y})^2 \right] dx \, dy
\]  
(A-1)

where \( p_x \) is the dip in the \( x \) direction and \( p_y \) is the dip in the \( y \) direction.

This can be simplified for the 2-D case to find the absolute time \( t(x) \) that minimizes

\[
J(t) = \int \left[ (p_x(t,x) - \frac{\partial t}{\partial x})^2 \right] dx
\]  
(A-2)

The Euler-Lagrange equation (Farlow, 1993) is used to find the function \( t(x) \) that minimizes the equation of this form

\[
J(t) = \int F(x,t,t') \, dx \approx 0
\]  
(A-3)

where the unknown \( t \) is a function of \( x \). The Euler-Lagrange equation is

\[
\frac{\partial F}{\partial t} - \frac{d}{dx} \left[ \frac{\partial F}{\partial t'} \right] = 0
\]  
(A-4)

To apply this equation I find

\[
\frac{\partial F}{\partial t} = 2 \left[ p_x - \frac{\partial t}{\partial x} \right] \frac{\partial p_x}{\partial t}
\]  
(A-5)

and

\[
\frac{\partial F}{\partial t'} = -2 \left[ p_x - \frac{\partial t}{\partial x} \right]
\]  
(A-6)

Substituting equations (A-5) and (A-6) into equation (A-4) and simplifying, I get:

\[
\frac{\partial^2 t}{\partial x^2} = \frac{\partial p_x}{\partial x} + \frac{1}{2} \frac{\partial p_x^2}{\partial t}
\]  
(A-7)

It is straightforward to extend to the 3-D case.
Amplitude balanced PEF estimation

Antoine Guitton

ABSTRACT
Inverse theory teaches us that the residual, or misfit function, should be weighted by the inverse covariance matrix of the noise. Because the covariance operator is often difficult to estimate, we can approximate it with a diagonal weight that can be more easily computed. This paper investigates the possible choices of weighting functions for the data residual when prediction-error filters are estimated. Examples with 2-D and 3-D field data prove that it is better to weight the residual than to weight the data before starting the inversion.

INTRODUCTION
In seismic processing, we often have to estimate filters in order to, for example, improve the result of tomography (Clapp, 2001), interpolate data (Spitz, 1991; Crawley et al., 1999) or perform signal-noise separation (Soubaras, 1994; Abma and Claerbout, 1995). The filter estimation can be done in the time or frequency domain. In general, we first start by estimating the filters and then use them for a particular geophysical task. One general assumption when we estimate filters is that the time series from which we estimate the filters is wide-sense stationary. Often with real data, however, this assumption is violated. The non-stationarity can be related to a different moveout or a different amplitude behavior throughout the seismic record. The filter estimation in the first case can be improved by introducing patches. We then estimate one filter per patch assuming that the data are locally stationary. The boundary conditions for patching technology are rather difficult to handle and a better solution is to use non-stationary filters (Crawley et al., 1999). The amplitude aspect of non-stationarity is usually tackled before processing by applying, for example, an Automatic Gain Control (AGC) on the data. One problem with this approach is that it is a non-linear process that can damage important amplitude information. In a setting where more attributes are extracted from seismic data to derive rock properties, treating amplitudes accurately becomes a major challenge in seismic processing.

In this paper, I show strategies for estimating filters when amplitude problems exist with seismic data. My claim is that (1) any weight on the data should be applied as late as possible in the processing work-flow, preferably for display purposes only and (2) if weights are needed, they should be incorporated inside our processing scheme without letting any footprint in the final image (Claerbout, 1992). Inverse theory provides us with tools to handle amplitude problems which are usually not used by the industry. Theoretically, from a statistical view-

1email: antoine@sep.stanford.edu
point (Tarantola, 1987), the residuals should be weighted during the filter estimation by the inverse covariance matrix of the noise. This requirement might be quite difficult to meet because we might not know the noise a-priori and/or the covariance operator might be too expensive to compute. In the context of filter estimation, I will assume that the covariance operator is a diagonal weight. Here I investigate the filter estimation problem for signal/noise separation. I will show with 2-D and 3-D field data that it is important to weight the residual during the filter-estimation step to obtain the best noise attenuation results.

In the first section following this introduction, I will describe the theory of prediction-error filters (PEF) estimation when the data are weighted and when the residuals are weighted. Then, I will show with a 2-D example with stationary filters and a 3-D example with non-stationary filters that a weighted-residual PEF estimation leads to the best noise attenuation results, as predicted by inverse theory.

**THEORY OF PEF ESTIMATION**

This section is largely inspired by chapter 7.5 in Claerbout (1992). I show that the proper way of dealing with amplitude problems with seismic data for the PEF estimation is to weight the residual and not the data itself.

I will now describe the theory of PEF estimation for a three coefficients filter \( a = (1, a_1, a_2)' \) from a time series \( y = (y_0, y_1, y_2, \ldots, y_6)' \). Helical boundary conditions (Claerbout, 1998) make this analysis valid for multidimensional PEFs as well. What we want is to minimize the energy of the residual \( r \) according to

\[
0 \approx r = YK a + r_0
\]

which can be rewritten

\[
0 \approx r = YK a + r_0
\]

where \( Y \) is the convolution matrix with \( y \), \( K \) is a masking operator and \( r_0 \) the first column of the convolution matrix \( Y \). The next step is to find \( a \) such that

\[
f(a) = \|r\|^2 = \|YK a + r_0\|^2
\]

is minimum. We can solve this problem with an iterative solver such as a conjugate-direction method (Claerbout and Fomel, 2002). The least-squares inverse of \( a \) becomes

\[
\hat{a} = -(K'Y'YK)^{-1}K'Y'r_0.
\]

It is common practice to weight the data before doing any processing in order to correct for
amplitude anomalies. Doing so, the fitting goal in equation (1) becomes

\[
0 \approx r = \begin{bmatrix}
w_2 y_2 & w_1 y_1 & w_0 y_0 \\
w_3 y_3 & w_2 y_2 & w_1 y_1 \\
w_4 y_4 & w_3 y_3 & w_2 y_2 \\
w_5 y_5 & w_4 y_4 & w_3 y_3 \\
w_6 y_6 & w_5 y_5 & w_4 y_4 \\
\end{bmatrix} \begin{bmatrix}
0 & \cdots & 0 \\
0 & \cdots & 1 \\
0 & \cdots & 2 \\
0 & \cdots & 3 \\
0 & \cdots & 4 \\
\end{bmatrix} \begin{bmatrix}
a_1 \\
a_2 \\
\end{bmatrix} + \begin{bmatrix}
w_2 y_2 \\
w_3 y_3 \\
w_4 y_4 \\
w_5 y_5 \\
w_6 y_6 \\
\end{bmatrix}
\]

(5)

where \( w = (w_1, w_2, \ldots, w_6)' \) are weights for a particular point of the time series \( y \). This is the wrong approach to correct for amplitude anomalies but it has practical values. First, equation (5) keeps the Toeplitz structure of the Hessian in equation (4), allowing fast computation of \( \hat{a} \). Then, the PEF can be easily estimated in the Fourier domain. However, inverse theory teaches us that we should be weighting the residual instead such that

\[
0 \approx W r = \begin{bmatrix}
w_2 y_2 & w_1 y_1 & w_0 y_0 \\
w_3 y_3 & w_2 y_2 & w_1 y_1 \\
w_4 y_4 & w_3 y_3 & w_2 y_2 \\
w_5 y_5 & w_4 y_4 & w_3 y_3 \\
w_6 y_6 & w_5 y_5 & w_4 y_4 \\
\end{bmatrix} \begin{bmatrix}
0 & \cdots & 0 \\
0 & \cdots & 1 \\
0 & \cdots & 2 \\
0 & \cdots & 3 \\
0 & \cdots & 4 \\
\end{bmatrix} \begin{bmatrix}
a_1 \\
a_2 \\
\end{bmatrix} + \begin{bmatrix}
w_2 y_2 \\
w_3 y_3 \\
w_4 y_4 \\
w_5 y_5 \\
w_6 y_6 \\
\end{bmatrix}
\]

(6)

with

\[
W = \begin{bmatrix}
w_2 & 0 & 0 & 0 & 0 \\
0 & w_3 & 0 & 0 & 0 \\
0 & 0 & w_4 & 0 & 0 \\
0 & 0 & 0 & w_5 & 0 \\
0 & 0 & 0 & 0 & w_6 \\
\end{bmatrix}
\]

(7)

The difference between equations (5) and (6) is that in the first case, we weight the data points and that in the second case, we weight the regression equations. In equation (6), we weight each row independently and leverage the PEF estimation globally. With the weighted residual, the Hessian loses its Toeplitz structure making the estimation of \( a \) less computationally efficient. In addition, the estimation of \( a \) in the Fourier domain becomes much more difficult.

It might not be clear why one method is better than the other. The choice of a weighting function can also be tricky. In the next section, I present guidelines and results on how to choose \( W \) and what it changes for two signal/noise separation examples.

**SIGNAL-NOISE SEPARATION WITH WEIGHTED PEFs**

In this section, I test the idea of weighting the residual for the PEF estimation as it is done in equation (6). I present guidelines on how to choose the weighting function. I illustrate my method with a 2-D and a 3-D data example for stationary and non-stationary PEFs estimation, respectively.
A brief theory of signal/noise separation

I present one possible formulation of the signal/noise separation problem as exemplified by Soubaras (1994); Abma and Claerbout (1995); Brown and Clapp (2000); Guitton et al. (2001). To simplify, we have the two following fitting goals:

\[ 0 \approx r_d = N(s - d) \]
\[ 0 \approx c r_s = c S s, \] (8)

where the data \( d \) is equal to the sum of the noise \( n \) and signal \( s \), \( N \) is the PEF for the noise \( (N n \approx 0) \), \( S \) is the PEF for the signal \( (S s \approx 0) \), and \( c \) a constant. Solving for \( s \) we have

\[ \hat{s} = (N'N + \epsilon^2S'S)^{-1}N'Nd. \] (9)

The signal-noise attenuation is then separated into two distinct steps: first we estimate the noise and signal filters \( N \) and \( S \), then we estimate the signal \( s \) based on the fitting goals in equation (8).

A 2-D example

Now I show how to estimate PEFs with data having amplitude anomalies. Figure 1a display a 2-D shot gather from a land survey. This gather shows high amplitudes at short offsets as indicated by the red color. The noise we want to attenuate is the low velocity/low frequency event visible throughout the section. I estimated a noise model in Figure 1b by transforming the data in the Radon domain and muting the velocities corresponding to the fastest events. The signal model in Figure 1c is obtained by subtracting Figure 1b to Figure 1a. This gather is particularly interesting because it illustrates very well the problem of amplitude balancing with real data. Indeed, it is clear from Figure 1 that the amplitude is not uniform across offset and time. That will cause problems for the PEF estimation and the signal-noise separation steps.

I show in Figure 2 the result of the signal noise separation when no weight is applied, i.e, equation (1). Figures 2a and 2c show the estimated signal and noise, respectively. The noise attenuation clearly failed here. Figures 2b and 2d display the impulse responses of the signal PEF and the noise PEF, respectively. In theory, Figure 2b should look like the signal (or at least have the same spectrum) and Figure 2d should look like the noise. The signal PEF is clearly wrong here since we do not retrieve the signal spectrum very well. This mismatch comes from the high amplitudes at short offset in Figure 1a that bias the estimation of the filter.

To make things right, a weighting function is needed for the residual. My choice of a weighting function is based on statistical considerations. I apply a weighting function on the data and look at the histogram. If the histogram has a Gaussian distribution, I keep it and use it during the estimation of the noise and signal PEFs. I keep this weighting function unchanged during the iterations, although I could reestimate it as it is done with Iteratively Reweighted Least Squares methods. Figure 3 shows four histograms corresponding to different weighting functions. The Amplitude Gain Control (AGC) produces the most satisfying result in term of
Figure 1: One shot gather from a 2-D land acquisition survey. The clipped values are shown in red. The input data on the left show high amplitudes at short offset. The middle and right panels display the noise and signal model, respectively.

Figure 4 displays the final noise attenuation result with the weighted PEF estimation. Figure 4a shows the estimated signal and Figure 4c the estimated noise. The noise has been successfully attenuated. Note that some signal has leaked in the noise in Figure 4c around 0.5 second. This is because I estimate only one noise and signal filter for the whole gather. But still, the noise is correctly attenuated. Figures 4b and 4d show a spike divided by the signal and noise PEFs respectively. Figure 4b demonstrates that the signal PEF correctly represents the quasi-flat signal. Figure 4d looks very similar to the noise with the correct amplitude behavior, as expected.

A last necessary comparison is by estimating the noise and signal PEFs from the weighted data [equation (5)] and look at the noise attenuation result. Figure 5 displays the noise attenuation result with the weighted data for the PEF estimation. The estimated signal in Figure 5a is satisfying with artifacts above 0.5 second (also visible in the estimated noise in Figure 5c). Figure 6 shows the difference between Figures 4a and 5a. In addition to the artifacts above 0.5 second, the noise is better removed at short offset in Figure 4a. The inverse PEF for the signal in Figure 5b is very similar to Figure 4b, but the amplitude decay is less important. The
Figure 2: (a) Signal estimated with no weights for the PEF estimation (equation (1)). (b) Spike divided by the signal PEF. (c) Estimated noise. (d) Spike divided by the noise PEF.
Figure 3: (a) Histogram of the input data. The amplitude of the data is not very well distributed. (b) Histogram of the data after geometrical spreading correction. The distribution is still not satisfying. (c) Histogram of the data after AGC. The data are very well balanced as indicated by the bell-shaped function. (d) Histogram of the data after envelope scaling. The distribution of the amplitudes is not as good as the AGC result.

Inverse PEF for the noise in Figure 5d is very different from Figure 4d. With the weighted residual, the filter is minimum phase whereas with the weighted data, it is not because the amplitude increases constantly with offset and time in Figure 5d.

Therefore, weighting the data or the residual for the PEF estimation produces very different filters affecting the final noise attenuation result. The best results are obtained when the residual is weighted to balance the relative importance of the regression equations. In the next section, I exemplify the weighted PEF estimation with a 3-D example and non-stationary filters.
Figure 4: (a) Signal estimated with weighted residual for the PEF estimation (equation (6)). (b) Spike divided by the signal PEF. (c) Estimated noise. (d) Spike divided by the noise PEF.
Figure 5: (a) Signal estimated with weighted data for the PEF estimation (equation (5)). (b) Spike divided by the signal PEF. (c) Estimated noise. (d) Spike divided by the noise PEF.
Figure 6: Difference plot between Figures 4 and 5. Both results are very similar but the weighted data strategy leads to a less efficient noise attenuation at short offset and leaves artifacts on top of the gather.

A 3-D land data example

I illustrate the weighted PEF estimation on a near-offset section from a 3-D land survey dataset. Figure 7 displays the 3-D data. The noise comes in different flavors. First, we have missing traces, second, we have large amplitude differences from trace to trace (common with land data), and then we have surface waves with strong amplitude variations along the time axis. I first design a weighting function that takes into account the missing traces and correct for the amplitude problems (Figure 8). This weighting function is going to be utilized for the PEF estimation. With the time domain formulation, there is no limit in the complexity of the weighting operator since it can mix zero traces with weights of any kind. This would be much more difficult, if not impossible, in the Fourier domain.

For this 3-D example, I estimate non-stationary 3-D filters. These filters are constant within a patch. For this dataset, I have a total of 90 patches with ten patches in the time axis and three in the crossline and inline directions. More details on the non-stationary filters can be found in Crawley (2000) and Claerbout and Fomel (2002). The first step consists in estimating the noise and signal non-stationary PEFs. The noise in Figure 7 is mainly ground-roll. A good noise model can be then obtained by low-passing the data and a signal model by high-passing the data.

Figures 11 and 12 show the noise attenuation results with and without residual weighting, respectively. However, a mask has been applied for the PEF estimation in Figure 11 to not include the missing traces. Figure 12 shows better results in the lower part (below 0.42 s.) of the section where some low-frequency noise is still visible. The difference is particularly obvious in the time section where a channel is clearly visible in Figure 12. One problem with the input data is that the amplitude decreases with time, especially for the noise (Figure 7). Therefore, during the signal and noise PEFs estimation steps, most of the solver efforts are directed toward the PEFs where the noise and signal is the strongest, i.e, the upper part. Thus, we end-up with “good” PEFs in the top, and “bad” PEFs in the bottom where the noise separation is the less efficient. Figure 9 displays the estimated PEFs for the noise and signal with and without residual weight. Note that in Figures 9a and 9c the coefficients (especially the first two) vary a lot with the patch number. They almost go to zero for the PEFs at the bottom. On the contrary, in Figures 9b and 9d, the PEFs coefficients are much much uniform.
across time and offset, as expected with the weighted PEF estimation technique.

Figure 7: A near-offset section of a 3-D land survey. Some signal is visible near 0.42 s. This section is contaminated with ground-roll. The amplitude varies across time and offset with missing traces as well.

To finish with the 3-D data example, I show in Figure 13 the noise attenuation result when the noise and signal models are weighted prior to the PEF estimation. In that case, the residual is not weighted, but only the data are, i.e., equation (5). Note that the same inversion parameters ($\epsilon$ in equation (8), number of iterations, patch geometry, PEF sizes) are used for both cases. This result is also satisfying but the signal is not as well preserved as it is in Figure 12. I show a difference plot between the two results in Figure 14. Looking closely at the time slices (upper panel) of Figures 12 and 13, we see at large black area between 6000 and 4000 meters in the crossline direction and between 2670 and 3670 meters in the inline direction. We can see the same black shape in the difference plot of Figure 14, thus demonstrating that more noise has been subtracted in Figure 12. The same conclusion holds for the white area above the black shape in Figures 12 and 13. These differences prove that weighting the residual is the correct way of handling amplitude problems with seismic data.

**CONCLUSION**

Estimating filters is an important step of many processing techniques like signal/noise separation, data interpolation, missing data restoration or regularized tomography. The stationarity of the data is at the heart of many filter estimation techniques. In the case where the data have strong amplitudes variations, I show that a weighting function that leverages the fitting
Figure 8: Weighting function for the non-stationary PEFs estimation. Whiter means larger weight. The weight is zero where traces are missing. [ER,M]

equations during the filter estimation step must be used for improved results. In other words, the residual should be weighted and not the data points. Test with 2-D and 3-D field data with stationary and non-stationary filters for two signal-noise separation problems prove that a weighted residual PEF estimation scheme give the best estimated noise and signal while preserving the amplitudes.

REFERENCES


Figure 9: PEFs estimated for each patch for the noise and signal. The vertical axis represents the PEF coefficients (the leading one coefficient is omitted) and the horizontal axis represents a patch number (Figure 10). (a) Noise PEFs with unweighted residual. (b) Noise PEFs with weighted residual. Note that the noise PEFs are very close to a 1-D second derivative (1, -2, 1). (c) Signal PEFs with unweighted residual. (d) Signal PEFs with weighted residual.

Figure 10: Geometry of the patches. The numbers correspond to the vertical axis of Figure 9.
Figure 11: Signal estimated without weight for the signal and noise PEFs estimation. Some noise remains below 0.42 s.

Figure 12: Signal estimated with weight for the signal and noise PEFs estimation. The noise is well attenuated. The time slice displays the channel more clearly than in Figure 11.
Figure 13: Signal estimated with a weighted noise and signal model for the PEFs estimation.

Figure 14: Difference between Figure 12 and Figure 13. The signal is better preserved with the weighted residual than with the weighted noise and signal models.


Coherent noise suppression in electroseismic data with non-stationary prediction-error filters

Seth Haines and Antoine Guitton

ABSTRACT
Non-stationary prediction-error filters (PEF’s) provide an effective means for separating signal from coherent noise in electroseismic data. The electroseismic signal is much weaker than the noise, so we can not rely on windowing, transforms, or other alterations of the original data to create models for the PEF estimation. Instead, we design signal PEF’s using the physical predictability of the phenomena, and estimate noise PEF’s using portions of the original data. This technique is effective on synthetic and real data.

INTRODUCTION
Electroseismic phenomena produce two forms of energy: the interface response signal, and the coseismic noise (Haines and Guitton, 2002). Electroseismic data processing must attenuate the coseismic noise in order to reveal the interface response. In short, the signal is composed of flat events (the interface response has virtually zero moveout) while the noise (coseismic energy) has moveout similar to seismic data. The object is to remove the curved energy so that it does not contaminate the final stack of the gather. We employ the signal/noise separation technique described by Guitton (2003) to test its effectiveness in electroseismic processing. The basic approach is to estimate non-stationary prediction-error filters (PEF’s) for the signal and the noise, and to use these PEF’s in an iterative signal/noise separation following Guitton et al. (2001).

The electroseismic signal is far weaker than the noise so we can not hope to obtain an adequate model for the estimation of signal PEF’s by muting in the parabolic radon transform (PRT) domain, or other alterations of the original data. We can, however, take advantage of the fact that the amplitude pattern of the signal can be easily modeled. It is the pattern of the potential \( V \) of a dipole field:

\[
V(x, z) = \frac{qd}{4\pi \epsilon_0} \frac{z}{(x^2 + z^2)^{3/2}},
\]

as measured at a horizontal offset \( x \) from a dipole at depth \( z \), where \( q \) is the magnitude of the electrical charges, \( d \) is the distance between the two separated charges, and \( \epsilon_0 \) is the electrical permittivity of the Earth. Using a velocity model to provide the relationship between depth

\[1\text{email: shaines@pangea.stanford.edu, antoine@sep.stanford.edu}\]
and travel time, and making basic assumptions about the size of the Fresnel zone producing the dipole, we can compute a model of the relative amplitude measured at various locations for events corresponding with any given travel time. This amplitude pattern may be used directly in the estimation of one-dimensional (in the offset direction) PEF’s (since such a one-dimensional PEF contains no wavelet information), or we can use the amplitude pattern to scale synthetic wavelets to be used as models for PEF estimation. Thus if we have a velocity model for a particular study area, we can estimate non-stationary signal PEF’s to target any interface response events that may be in the data without the need for a priori knowledge of their arrival times. We use simple physics [equation (1)] to design general signal PEF’s, and use components of the original data to design noise PEF’s. We show that these non-stationary PEF’s provide an effective means for separating the interface response signal from the coseismic noise in synthetic and real electroseismic data.

SYNTHERETIC ELECTROSEISMIC DATA

We begin by testing the processing approach on simple synthetic data containing three interface response events, three coseismic arrivals, and a small amount of random noise (Figure 1a). For our first processing example we chose models for the PEF estimation (Figure 1c and d) that simply contain amplitude-normalized versions of the events in the synthetic data, plus random noise. For the noise model, this choice is not entirely unrealistic; horizontal geophone data collected with electroseismic data closely resembles the coseismic noise in the electroseismic record (Garambois and Dietrichz, 2001) and could be used in this capacity. For the signal model, this choice is rather unrealistic, as we would generally be looking for previously unknown interface response signal that might be entirely obscured by coseismic and background noise. The purpose of this first example is to verify that the method is effective, and this is demonstrated by the final result shown in Figure 1b. Virtually all of the coseismic noise has been removed, leaving only the interface response energy.

Our second synthetic processing example (Figure 2a) employs the same data and noise model as Figure 1, and a more general signal model. We use the amplitude pattern predicted by equation (1) to produce a model for signal PEF estimation (Figure 2c). We normalize the amplitude pattern in the time direction so that the PEF estimation equally considers deeper and shallower parts of the amplitude pattern. Because we use one-dimensional PEF’s (Haines and Guitton, 2002), this choice of signal model is quite reasonable. If we were to use two-dimensional signal PEF’s, we could not use this model as the PEF’s would be trying to model waveform information that is not present in the model. An alternative option would be to use the amplitude pattern (Figure 2c) to scale synthetic wavelets, and then to estimate 2-D signal PEF’s on that model. But this approach would lose some generality as we would have to assign particular arrival times to those arrivals, and is not necessary since we find one-dimensional PEF’s to be at least as effective as two-dimensional PEF’s. The final result (Figure 2b) is nearly as good as that of Figure 1b, and shows that the generality of this choice of signal model does not bring with it a significant degradation of the final result.

Our third synthetic example (Figure 3) adds an important element of realism to the syn-
Figure 1: a) Simple synthetic data, with horizontal interface response events created using equation (1) and an arbitrary velocity function. Coseismic noise is created with the same velocity function. b) Result after non-stationary PEF signal/noise separation. c) Normalized version of synthetic interface response, used as model used for the estimation of the signal PEF's. d) Normalized version of coseismic noise, used as model for the estimation of the noise PEF's.

The coupling of these electrodes with the ground is hardly uniform and results in amplitude variations between adjacent traces. We simulate this coupling variation by multiplying each trace of the synthetic by a random scalar (between 0.7 and 1.0), producing the data shown in Figure 3a. We use the same signal and noise models as in the previous example (Figure 2), and obtain the result shown in Figure 3b. This result contains more remnant coseismic noise than the previous examples, but the interface response energy is significantly stronger, and would dominate a stack of the gather.
Figure 2: a) Synthetic data, same as Figure 1a. b) Result after signal/noise separation. c) Model used for estimation of signal PEF’s, based on equation (1). d) Normalized version of coseismic noise, used as model for estimation of noise PEF’s.

REAL ELECTROSEISMIC DATA

We next test non-stationary PEF’s on real data (Figure 4a). This 48-channel shot gather was constructed by interleaving two 24-channel gathers that were collected according to the methods described by Haines et al. (2003) and Haines and Guitton (2002). The shot point is the same for the two records, and the receiver positions are shifted by half the receiver spacing for the second record. Thus the resulting 48-channel record has half the original receiver spacing, and provides a test gather that can be processed more easily than the original 24 channel records. The shot point is in the center of the dipole receiver line, in the geometry that would be used for a typical survey. The gather shown in Figure 4a has been pre-processed to remove 60 Hz energy and eliminate high- and low- frequency background noise.

Using the coseismic energy as a guide, we establish a basic velocity function to be used with equation (1) for the determination of the model to be used for signal PEF estimation (Figure 4c). For estimation of the noise PEF we use a windowed version of the original data.
Figure 3: a) Synthetic data, same as Figure 1a, but each trace is multiplied by a random number to simulate the imperfect electrode coupling that impacts electroseismic data. b) Result after signal/noise separation. c) Model used for estimation of signal PEF’s, based on equation (1). d) Normalized version of coseismic noise, used as model for noise PEF estimation.

This choice of noise model is effective, as shown by the final result (Figure 4b), but is somewhat unrealistic, as we would ideally be looking for signal within the noisy part of the data and would not want our noise PEF estimation to be impacted by any signal in the estimation model. A better choice would be to use horizontal geophone data, but the data collected to complement this record is of insufficient quality to be used as a model for PEF estimation. Though not shown here, results using geophone data as a model for noise PEF estimation contain considerably more coseismic noise. The final result (Figure 4b) shows that the signal noise separation effectively removed the bulk of the coseismic energy, leaving two interface response events in the first 0.01 seconds of the record, and predominantly random noise in the lower part of the record.
CONCLUSIONS AND FUTURE WORK

We have shown that non-stationary PEF’s provide an effective means for the electroseismic signal/noise separation. We demonstrate that the physics of the problem can be used to develop models for PEF estimation in this case where the signal may be so much weaker than the noise that alterations of the data cannot produce suitable models. The generality of this approach is important in electroseismic signal processing, as it enables us to attenuate the strong coherent noise and enhance previously obscured signal, without any *a priori* knowledge of the signal. These results are far superior to those obtained with f-k filtering, stationary PEF’s (Haines and Guitton, 2002), or other means. The real data example shown here is encouraging, and indicates that the development of an effective, fully general, algorithm for processing electroseismic data is an attainable goal. The next major step toward a generally applicable algorithm is the development of a better model for noise PEF estimation. One likely option is horizontal geophone data, as it should closely resemble the coseismic noise. The surprisingly
poor quality of the horizontal data collected as part of this project makes it unsuitable for such a use, but is likely due to the poor condition of the geophones. We plan to collect additional data using better quality geophones, and hope to find that these data can be used for noise PEF estimation.

Assuming that horizontal geophone data can effectively be used for PEF estimation, we suggest the following processing scheme for electroseismic data:

1. Pre-processing (60 Hz removal, frequency filtering).
2. Develop rough velocity model using coseismic and/or geophone data.
3. Estimate signal PEF on dipole amplitude pattern, using velocity model.
4. Estimate noise PEF on horizontal geophone data.
5. Use these non-stationary PEF’s in the signal/noise separation outlined here.

ACKNOWLEDGMENTS

We are grateful to Steve Pride, Biondo Biondi, Jerry Harris, Simon Klemperer, and Art Thompson for guidance in this project. The work would not be possible without the generosity of the Geological Society of America, the American Association of Petroleum Geologists, the Stanford McGee Fund, the Achievement Rewards for College Scientists Foundation, Jim and Carolyn Pride, and the Stanford Exploration Project.

REFERENCES


Short Note

Subtraction versus filtering for signal/noise separation

Antoine Guitton

Introduction

In Guitton (2001) I presented an efficient algorithm that attenuates coherent noise based on the spatial predictability of both the noise and the signal. I called this algorithm the subtraction method. This algorithm was first presented by Nemeth (1996) and works generally better than the standard projection filtering technique (Soubaras, 1994; Abma, 1995).

The main motivation in writing this paper is to better understand why the subtraction method attenuates the noise better than the filtering approach (Guitton and Claerbout, 2003). This is difficult to answer since both methods, although having their roots in the inversion theory framework, follow two different philosophies. The subtraction method models simultaneously the noise and the signal components with a noise and signal modeling operator, respectively, whereas the filtering method approximates the noise covariance operator with filters. It is not clear from these definitions why both approaches should lead to different results. This short note attempts to fill this gap.

In the first section following the introduction, I present the subtraction method as developed by Nemeth. Then I show that the subtraction method approximates the inverse noise covariance operator with a projection filter, as opposed to a single filter, e.g., a prediction-error filter, for the filtering approach. To illustrate the difference between both methods, I present a 3-D land data example and show that the subtraction approach gives the best noise attenuation result.

The subtraction method

In this section, I show that starting from the least-squares inverse of the subtraction method, I can retrieve a fitting goal that resembles the fitting goal of the filtering technique. I first assume that the data \( d \) is the sum of the signal \( s \) and the noise \( n \) as follows:

\[
d = s + n
\]
Now I assume that I have two modeling operators $L_s$ and $L_n$ for the signal and noise respectively such that
\[
L_s m_s = s, \\
L_n m_n = n,
\] (2)
where $m_n$ and $m_s$ are unknown model parameters. In this paper I assume that the two modeling operators are orthogonal, meaning that they model different regions of the data space. Inserting the modeling operators into equation (1) I have
\[
d = L_s m_s + L_n m_n. 
\] (3)
I then want to estimate $m_n$ and $m_s$ so that
\[
0 \approx r_d = L_s m_s + L_n m_n - d. 
\] (4)
In a least-squares sense, I have to minimize the following objective function:
\[
f(m_s, m_n) = \|r_d\|^2 = \|L_s m_s + L_n m_n - d\|^2. 
\] (5)
The least-squares inverse of the model parameters is given by (Guitton, 2001)
\[
\begin{pmatrix}
\hat{m}_n \\
\hat{m}_s
\end{pmatrix} = \begin{pmatrix}
(L_n^\prime \overline{R}_s L_n)^{-1} L_n^\prime \overline{R}_s \\
(L_s^\prime \overline{R}_n L_s)^{-1} L_s^\prime \overline{R}_n
\end{pmatrix} d, 
\] (6)
with
\[
\overline{R}_s = I - L_s (L_s^\prime L_s)^{-1} L_s^\prime, \\
\overline{R}_n = I - L_n (L_n^\prime L_n)^{-1} L_n^\prime. 
\] (7)
One can easily recognize in the definition of $\overline{R}_s$, the identity matrix minus the signal resolution operator and for $\overline{R}_n$, the identity matrix minus the noise resolution operator. Therefore, $\overline{R}_s$ and $\overline{R}_n$ are signal and noise filtering operators respectively with
\[
\overline{R}_s s \approx 0 \quad \text{and} \quad \overline{R}_n n \approx 0. 
\] (8)
An interesting property of $\overline{R}_s$ and $\overline{R}_n$ is that $\overline{R}_n \overline{R}_n = \overline{R}_n$ and $\overline{R}_s \overline{R}_s = \overline{R}_s$: they are called projectors. It is easy to verify that they are Hermitian operators as well (Guitton, 2001). Now, looking at the first row in equation (6), I have
\[
\hat{m}_n = (L_n^\prime \overline{R}_s L_n)^{-1} L_n^\prime \overline{R}_s d. 
\] (9)
This expression is the least-squares inverse for the following objective function
\[
f(m_n) = (L_n m_n - d)^\prime \overline{R}_s (L_n m_n - d) 
\] (10)
which can be rewritten
\[
f(m_n) = (L_n m_n - d)^\prime \overline{R}_s \overline{R}_s (L_n m_n - d) 
\] (11)
or
\[
f(m_n) = \| \overline{R}_s (L_n m_n - d) \|^2 
\] (12)
using the properties of $\overline{R}_s$. Therefore, equation (12) is the objective function for the fitting goal
\[
0 \approx r_d = \overline{R}_s (L_n m_n - d). 
\] (13)
Similarly, if I look at the second row of equation (6), I end up with
\[
0 \approx r_d = \overline{R}_n (L_s m_s - d). 
\] (14)
Why is it working?

Remember that \( L_n m_n - d \) is something that should resemble the signal. If I only solve

\[
0 \approx r_d = L_n m_n - d, \tag{15}
\]

I intrinsically assume that the signal has minimum energy. Inverse theory teaches us that I should weight the residual so that its spectrum is white (Tarantola, 1987; Claerbout and Fomel, 2001). \( R_s \) in equation (14) does exactly that since \( R_s s \approx 0 \) by definition. Thus, although not obvious from the fitting goal in equation (4) and the pseudo-inverse in equation (6), the subtraction method works very well because it approximates the covariance operators with the projection filters \( R_s \) and \( R_n \).

The filtering approach is identical except that I have

\[
0 \approx r_d = A_s (L_n m_n - d) \tag{16}
\]

instead of

\[
0 \approx r_d = R_s (L_n m_n - d), \tag{17}
\]

where \( A_s \) is usually a prediction-error filter for the noise such that \( A_s s = 0 \). So the difference between the two methods is numerically embedded within the weighting operators \( A_s \) and \( R_s \) in the fitting goals of equations (14) and (16).

So why is it better to have \( R_s \) instead of \( A_s \)? First, \( R_s \) is a projection filter meaning that it sets to zero every signal component and keeps the rest basically unchanged (depending on the orthogonality of the noise and the signal components). Therefore, I think that \( R_s \) is a better signal filter than \( A_s \) and has fewer impact on the norm (the same way that inversion is better than filtering). Then, the definition of \( R_s \) shows that the conditioning number of the Hessian should be better with \( R_s \) than with \( A_s \). This property has also been established by Guitton (2002) where I showed that both approaches were related by preconditioning transformations. Finally, another advantage of \( R_s \) is that the modeling operator can be anything I want, as long as \( L_s m_s = s \) in equation (2).

Therefore, the new interesting idea is that the prediction-error filter might not be the best approximation for the data covariance operator. A projection filter seems to be a better choice. I now illustrate the difference between the two examples with a 3-D field data example.

**A 3-D field data example**

I show with a 3-D field data example that the subtraction method works better than the filtering approach. The dataset is also used in Guitton (2003). Figure 1 shows the input data with an overwhelming amount of noise. For the subtraction and filtering methods, I use 3-D non-stationary prediction-error filters (PEFs) to approximate the modeling and inverse covariance operators, respectively. These filters have been estimated with a weighted PEF estimation scheme (Guitton, 2003) to take into account the amplitude variations in the data. The PEFs are estimated from a noise and signal model that I computed by band-passing the data: 0. to
45 Hz filtering for the signal model and 35 to 125 Hz. filtering for the noise model. Note that with the subtraction method, the modeling operators are computed with inverse PEFs. Unfortunately, these inverses are not guaranteed to be stable (Rickett, 1999). Therefore, with PEFs, the subtraction method might not be always feasible.

Figures 2 and 3 show the noise attenuation results for the filtering and subtraction method, respectively. Note that the PEFs and the patches for the two methods are identical. In addition, the convergence of both methods during the noise attenuation phase is very similar. It is interesting to notice that the subtraction method gives a cleaner panel, with more continuous reflectors everywhere. The amplitudes are also stronger with the subtraction method, as seen in the time slice section.

Figure 1: A near-offset section of a 3-D land survey. Some signal is visible near 0.42 s. This section is contaminated with ground-roll. The amplitude varies across time and offset with missing traces as well.
Conclusion

The subtraction and filtering methods are two different approaches for attenuating the noise present in seismic data. These two methods effectively approximate the covariance operator with a single filter, for the filtering approach, or a projection filter, for the subtraction method. The projection filter seems to improve (1) the conditioning of the Hessian and (2) the filtering properties of the covariance operator. Therefore, subtracting the noise might be the method of choice whenever possible.

REFERENCES


Figure 3: Estimated signal with the subtraction method. The signal is stronger and more continuous than in Figure 2.


Short Note

Enhanced random noise attenuation

Ioan Vlad\footnote{email: nick@sep.stanford.edu}

INTRODUCTION

Spatial prediction filtering attenuates random noise uncorrelated from trace to trace, while preserving linear, predictable events. The prediction is formulated as a least-squares problem in either the \( t-x \) or the \( f-x \) domain. The methods are casually known as "\( f-x \) decon" and "\( t-x \) decon." Although established by common practice, the name "decon" is not appropriate in these cases, because it suggests a similarity with the much better-known deconvolution of the signal along the time axis. However, deconvolution removes the predictable information (wavelet + multiples) and keeps the unpredictable (the reflectivity function), while \( f-x \) and \( t-x \) decons keep the predictable along the space axis (linear events), and remove the unpredictable (random noise). A very good explanation of spatial prediction filtering, with many examples, is given on page 960 of Yilmaz (2001). \( F-x \) decon was introduced by Canales (1984). Noise suppression in the \( t-x \) domain was developed by Abma (1995), within the framework of which the SEPlib programs tFx2d and tTxdec were developed. I will show how to achieve Enhanced Random Noise Attenuation (ERNA) by building upon these existing spatial prediction filtering methods.

IMPROVING ON EXISTING METHODS

When doing prediction filtering, the noise is defined as whatever remains after applying a signal-annihilation filter to the data. The filter must be designed on a sample of the data that has as little noise as possible. However, totally noise-free data is not available, so both \( f-x \) and \( t-x \) decon pass some noise along with the signal. Most often they “find” quasi-coherent patterns in the noise. The lower panels of Figure 2 are a good example of this phenomenon.

A close examination of the spurious coherent patterns passed by \( f-x \) and by \( t-x \) decon reveals that they are different. Simply averaging the results will increase the signal/noise ratio.\footnote{The maximum possible s/n ratio increase is \( \sqrt{2} \), when the noise in the two datasets being summed is uncorrelated and random.} Moreover, the potential for improvement does not stop here. The two semi-coherent
noise patterns passed by each method have interfered with each other, so their coherence is much reduced after averaging. Noise in the averaging result will therefore be vulnerable again to attenuation by spatial prediction filtering. I found in practice that a $t\times$ decon with a smaller filter than the first one used produces good results. The Enhanced Random Noise Attenuation (ERNA) flowchart is displayed in Figure 1. Figures 2, 3 and 4 show the result of applying this technique on real poststack seismic data, real prestack seismic data “spiked” with synthetic noise and Ground-Penetrating Radar (GPR) data. The results of $f\times$ and $t\times$ decons are also displayed alongside in the bottom panels for a comparison.

![One iteration flowchart](nick2-skema_bw)

**PRACTICAL ISSUES**

The improvement brought about by ERNA increases with the amount of random noise up to a point, after which it decreases as the signal becomes totally dominated by noise. As with any noise removal process, the residual (eliminated noise) should be routinely examined to see whether any meaningful signal has been thrown away too. Residuals should ideally look like pure white noise. If they do not, then substitute the residual for the input data and iterate again.

---

3 Arco data used in Figure 5.3. of Abma (1995).
4 Made available to SEP by Gulf Science and Technology Company and used by Kjartansson (1979) and many other SEP report papers.
5 Courtesy of Ran Bachrach, Michigan State University.
(see Figure 1). If signal is still seen in the last residual, and additional iterations do not bring improvement, then modification of the parameters of the f-x and t-x decons may be necessary.

Sometimes a very small amount of signal will be left in the residuals even after extensive adjustment of the parameters. While this may signify that the optimal combination of parameters has not been attained yet, in practice the result is better than either f-x or t-x decon. Given the way the method works (averaging the results of f-x and t-x decons), in order for signal to be overlooked by ERNA, it has to be overlooked by both types of decon. The left panels of Figure 5 show the final residuals after applying several ERNA iterations to each of the three datasets presented in Figures 2, 3, and 4. Hints of signal can be distinguished, especially in the prestack seismic residual. However, it must be kept in mind that the amplitude in the panels has been gained for visualization. The Root Mean Square (RMS) values of the residuals throughout the iterations (normalized to the value of the input data) are shown in the right side panels in the same Figure 5. The first iteration removes most of the noise; the rest only extract leftover signal from the residuals. After a small number of iterations ERNA converges to its practical limit.

Experience with data strongly affected by coherent noise shows that the method does not react well to crossing dips, especially when one set of events is less coherent than the other. Applying the method to such data results in a patchwork of small regions in which a single dip dominates, much like the results of a dip field estimator that attempts to find the most energetic local dip. Another instance when ERNA may not function optimally is the case of very simple synthetics, which contain extended surfaces of constant (especially zero) values between events. This is because f-x and t-x "decons" may have trouble with such areas. "Spiking" very clean real data with noise (Figure 3) can be a good alternative to using a simple synthetic.

$\text{tFx2d}$ is cheaper to apply than $\text{tTxdec}$, the cost of which increases with the cube of the filter size. $\text{tTxdec}$ can be applied on a 3-D cube, doing true 3-D spatial prediction, while with $\text{tFx2d}$ one has to resort to a surrogate involving multiple passes. The cost of 3-D t-x decon can be prohibitive nonetheless, so a 2-D decon may have to be used. Simply looping 2-D noise attenuation operators along the third dimension of a cube will eliminate data that is coherent along the third dimension, but is not coherent in the filtering plane. This is easily visible in the residual. To simulate the effects of true 3-D noise attenuation one has to take the residual (not the result) of the first 2-D filtering along the first dimension, and to filter it along another dimension, obtaining a second result and a second residual. This second residual is then filtered along the third dimension to obtain a third filtering result and a third residual. The final result is the sum of the three filtering results.

A potential objection to the method is that the noise passed by the spatial prediction filtering programs consists only of coding artifacts due to the different size of the windows. The response is that when varying the parameters, including window sizes, inside the same method, the artifacts remain similar enough as not to destruct their coherence when superimposed. However, the patterns passed by any of the f-x runs are different from those passed by the t-x, and they interfere to destroy their coherence. The two spatial prediction filtering methods employed are supposed in theory to produce similar results, by working in two different domains. However, we see that in practice they produce different artefacts. A more quantita-
tive analysis of the properties of the noise passed by \( f-x \) and \( t-x \) decon may be warranted.

My experience showed that ERNA can benefit velocity analysis in the case of 2D prestack seismic data when automatic velocity analysis is affordable for each midpoint. Denoising common-offset planes greatly improved the ability of automatic velocity picking programs to output results that are consistent across midpoint.

\( F-x \) decon is commonly implemented in seismic processing packages, and it also exists in SEPlib, as does \( t-x \) decon. There is therefore no need to write any code in order to implement the ERNA technique. Everything can be accomplished in a Makefile, a shell script or any other form of batch file.

**CONCLUSIONS**

The spurious coherent noise patterns passed by the two spatial prediction filtering methods are different enough that their coherence is destroyed when added together. A formal mathematical investigation on why this holds true in the vast majority of cases may yield useful insights on optimally choosing the parameters for the two methods so that the final signal-to-noise ratio is maximized. However, even on a heuristic basis, Enhanced Random Noise Attenuation works. It is simple, fast, intuitive, and does not require any coding if \( f-x \) and \( t-x \) decons are already implemented. Without being more unstable than either of these techniques, it builds on both of them in order to produce a better result.

**ACKNOWLEDGMENTS**

Ioan Vlad is supported by a Burt and Deedee McMurtry Stanford Graduate Fellowship.

**REFERENCES**


Figure 2: **Top, left:** original GPR data. **Top, right:** Enhanced Random Noise Attenuation result. **Bottom, left:** f-x decon result. **Bottom, right:** t-x decon result. The encircled area has fewer artifacts in the ERNA result. Differences between panels are much more visible when viewed electronically.
Figure 3: **Top, left:** original prestack seismic data. **Top, right:** Enhanced Random Noise Attenuation result. **Bottom, left:** $f$-x decon result. **Bottom, right:** $t$-x decon result. In the ERNA result, the upper encircled area has fewer artifacts, and the lower encircled area shows more continuous reflectors (less “checkerboard” noise). Differences between panels are much more visible when viewed electronically. [nick2-kja][ER]
Figure 4: **Top, left:** original poststack seismic data. **Top, right:** Enhanced Random Noise Attenuation result. **Bottom, left:** f-x decon result. **Bottom, right:** t-x decon result. The encircled area at the bottom of the figure shows more continuous reflectors (less “checkerboard” noise). Differences between panels are much more visible when viewed electronically.
Figure 5: Final ERNA residuals and the RMS values of successive residuals, normalized with respect to the input data (shown as iteration 0 with a RMS of 1). **Top panels:** GPR. **Middle panels:** Seismic prestack. **Bottom panels:** Seismic poststack.
Wave-equation migration velocity analysis by inversion of differential image perturbations

Paul Sava and Biondo Biondi

ABSTRACT

Wave-equation migration velocity analysis is based on the linear relation that can be established between a perturbation in the migrated image and the corresponding perturbation in the slowness function. Our method formulates an objective function in the image space, in contrast with other wave-equation tomography techniques which formulate objective functions in the data space. We iteratively update the slowness function to account for improvements in the focusing quality of the migrated image. Wave-equation migration velocity analysis (WEMVA) produces wrong results if it starts from an image perturbation which is not compliant with the assumed Born approximation. Other attempts to correct this problem lead to either unreliable or hard to implement solutions. We overcome the major limitation of the Born approximation by creating image perturbations consistent with this approximation. Our image perturbation operator is computed as a derivative of prestack Stolt residual migration, thus our method directly exploits the power of prestack residual migration in migration velocity analysis.

INTRODUCTION

Migration velocity analysis based on downward continuation methods, also known as wave-equation migration velocity analysis, is a technique designed as a companion to wave-equation migration (Biondi and Sava, 1999; Sava and Fomel, 2002). The main idea of WEMVA is to use downward continuation operators for migration velocity analysis (MVA), as well as for migration. This is in contrast with other techniques which use downward continuation for migration, but traveltime-based techniques for migration velocity analysis (Clapp, 2001; Liu et al., 2001; Mosher et al., 2001).

WEMVA is closer to conventional MVA than other wave-equation tomography methods (Noble et al., 1991; Bunks et al., 1995; Forgues et al., 1998) because it tries to maximize the quality of the migrated image instead of trying to match the recorded data. In this respect, our method is related to Differential Semblance Optimization (Symes and Carazzone, 1991) and Multiple Migration Fitting (Chavent and Jacewitz, 1995). However, with respect to these two methods, our method has the advantage of exploiting the power of residual prestack migration to speed up the convergence, and it also gives us the ability to guide the inversion by geologic
WEMVA benefits from the same advantages over traveltime estimation methods as wave-equation migration benefits over Kirchhoff migration. The most important among them are the accurate handling of complex wavefields which are characterized by multipathing, and the band-limited nature of the imaging process, which can handle sharp velocity variations much better than traveltime-based methods (Woodward, 1992). Complex geology, therefore, is where WEMVA is expected to provide the largest benefits.

WEMVA is based on a linearization of the downward-continuation operator using the Born approximation. This approximation leads to severe limitations on the magnitude and size of the anomalies that can be handled. It, therefore, cannot operate successfully exactly in the regions of highest complexity. Other methods of linearization are possible (Sava and Fomel, 2002), but neither one allows for arbitrarily large anomalies.

In our early tests (Biondi and Sava, 1999), we construct the image perturbation using Prestack Stolt Residual Migration (PSRM) (Stolt, 1996; Sava, 2000). In summary, this residual migration method provides updated images for new velocity maps that correspond to a fixed ratio ($\rho$) of the new velocity with respect to the original velocity map. Residual migration is run for various ratio parameters, and finally we pick the best image by selecting the flattest gathers at every point.

The main disadvantage of building the image perturbation using PSRM is that, for large velocity ratio parameters ($\rho$), the background and improved images can get more than $\pi/4$ out of phase. Therefore, the image perturbation computed by the forward operator and the one computed by residual migration are fundamentally different, and can have contradictory behaviors when using the Born-linearized WEMVA operator for inversion.

Alternative methods can be used to create image perturbations for WEMVA, in compliance with the Born approximation and computed directly from the background image (Sava and Biondi, 2001). We use an analytic differential procedure starting from the background image and leading to image perturbations similar to the ones created using the forward WEMVA operator.

**WAVEFIELD SCATTERING**

In migration by downward continuation, the wavefield at depth $z + \Delta z$ is obtained by phase-shift from the wavefield at depth $z$:

$$W(z + \Delta z) = W(z) e^{-ikz\Delta z}.$$  

Using a Taylor series expansion, the depth wavenumber ($k_z$) depends linearly on its value in the reference medium ($k_{zo}$) and the laterally varying slowness $s(x, y, z)$ in the depth interval under consideration

$$k_z \approx k_{zo} + \frac{dk_z}{ds} \bigg|_{s = s_o} (s - s_o).$$  


$s_o$ represents the constant slowness associated with the depth slab between the two depth intervals. $\frac{dk}{ds}\bigg|_{s=s_o}$ represents the derivative of the depth wavenumber with respect to the reference slowness and can be implemented in many different ways, e.g. by the Fourier-domain method of Huang et al. (1999).

The wavefield downward continued through the background slowness $s_b(x,y,z)$ is

$$W_b(z + \Delta z) = W(z)e^{-i \left[ k_{z_o} + \frac{dk}{ds}\bigg|_{s=s_o}(s_b-s_o) \right] \Delta z},$$

and the full wavefield $W(z + \Delta z)$ depends on the background wavefield $W_b(z + \Delta z)$ by

$$W(z + \Delta z) = W_b(z + \Delta z)e^{-i \left[ k_{z_o} + \frac{dk}{ds}\bigg|_{s=s_o} \Delta s \Delta z \right]},$$

where $\Delta s$ represents the difference between the true and background slownesses $\Delta s = s - s_b$.

Defining the wavefield perturbation $\Delta W(z + \Delta z)$ as the difference between the wavefield propagated through the medium with correct velocity, $W(z + \Delta z)$, and the wavefield propagated through the background medium, $W_b(z + \Delta z)$, we can write

$$\Delta W(z + \Delta z) = W(z + \Delta z) - W_b(z + \Delta z)$$

$$= W_b(z + \Delta z) \left[ e^{-i \left[ k_{z_o} + \frac{dk}{ds}\bigg|_{s=s_o} \Delta s \Delta z \right]} - 1 \right].$$

Equation (5) represents the foundation of the wave-equation migration velocity analysis method. One major problem with Equation (5) is that the wavefield $\Delta W$ and slowness perturbations $\Delta s$ are not linearly related. For inversion purposes, we need to find a linearization of this equation around the reference slowness, $s_o$. Biondi and Sava (1999) linearize Equation (5) using the Born approximation ($e^{i\phi} \approx 1 + i\phi$). With this choice, the WEMVA Equation (5) becomes

$$\Delta W(z + \Delta z) = W_b(z + \Delta z) \left[ -i \left. \frac{dk}{ds} \bigg|_{s=s_o} \Delta s \Delta z \right] \right].$$

The wavefield perturbation $\Delta W$ in Equation (7) turns into an image perturbation $\Delta R$ after we apply an imaging condition, e.g. summation over frequencies. The WEMVA objective function is

$$\min_{\Delta s} \| \Delta R - L\Delta s \|$$

where $L$ is a linear operator defined recursively from Equation (7) at every depth level and frequency. We estimate the slowness update by minimizing this objective function through an iterative conjugate gradient optimization technique.

**DIFFERENTIAL IMAGE PERTURBATION**

Residual migration can be used to create image perturbations ($\Delta R$). In its simplest form, we can build $\Delta R$ as a difference between an improved image ($R$) and the reference image ($R_b$)

$$\Delta R = R - R_b,$$
where $\mathcal{R}$ is derived from $\mathcal{R}_b$ as a function of the parameter $\rho$, which is the ratio of the original and improved velocities. Of course, the improved velocity map is not known explicitly, but is described indirectly by the ratio map of the two velocities.

If we denote by 1 the velocity ratio that corresponds to the background velocity model and define $\Delta \rho = \rho - 1$, we can also write the discrete version of the image perturbation as

$$\Delta \mathcal{R} \approx \frac{\mathcal{R} - \mathcal{R}_b}{\rho - 1} \Delta \rho,$$

which can also be written in differential form as

$$\Delta \mathcal{R} \approx \frac{d \mathcal{R}}{d \rho} \bigg|_{\rho=1} \Delta \rho,$$

or, equivalently, using the chain rule, as

$$\Delta \mathcal{R} \approx \frac{d \mathcal{R} \, dk_z}{dk_z \, d\rho} \bigg|_{\rho=1} \Delta \rho,$$

where $k_z$ is the depth wavenumber defined for PSRM.

Equation (12) offers the possibility to build the image perturbation directly. We achieve this by computing three elements: the derivative of the image with respect to the depth wavenumber, and two weighting functions, one for the derivative of the depth wavenumber with respect to the velocity ratio parameter ($\rho$), and the other one for the magnitude of the $\Delta \rho$ perturbation from the reference to the improved image.

Firstly, the image derivative in the Fourier domain, $\frac{d \mathcal{R}}{dk_z}$, is straightforward to compute in the space domain as

$$\frac{d \mathcal{R}}{dk_z} \bigg|_{\rho=1} = -iz \mathcal{R}_b.$$

The derivative image is the imaginary part of the migrated image scaled by depth.

Secondly, we can obtain the weight representing the derivative of the depth wavenumber with respect to the velocity ratio parameter ($\frac{dk_z}{d\rho}$) starting from the double square root (DSR) equation written for prestack Stolt residual migration (Sava, 2000):

$$k_z = k_{zs} + k_{zr} = \frac{1}{2} \sqrt{\rho^2 \mu^2 - |k_s|^2} + \frac{1}{2} \sqrt{\rho^2 \mu^2 - |k_r|^2},$$

where $\mu$ is given by the expression:

$$\mu^2 = \frac{\left[4(k_{zo})^2 + (|k_r| - |k_s|)^2\right] \left[4(k_{zo})^2 + (|k_r| + |k_s|)^2\right]}{16(k_{zo})^2},$$

$(k_{zs}, k_{zr})$ are the depth wavenumbers and $(|k_s|, |k_r|)$ are the spatial wavenumbers for the sources and receivers, respectively.
The derivative of \( k_z \) with respect to \( \rho \) is
\[
\frac{dk_z}{d\rho} = \rho \left( \frac{\mu^2}{4k_{zs}} + \frac{\mu^2}{4k_{sr}} \right),
\]
therefore we can write
\[
\left. \frac{dk_z}{d\rho} \right|_{\rho=1} = \frac{\mu^2}{2\sqrt{\mu^2 - |k_s|^2}} + \frac{\mu^2}{2\sqrt{\mu^2 - |k_r|^2}}.
\]

Finally, \( \Delta \rho \) can be picked from the set of residually migrated images at various values of the parameter \( \rho \) (Sava, 2000). One criterion that could be used to estimate \( \Delta \rho \) is the flatness of the angle-domain image gathers, which can be measured through derived parameters such as stack power, semblance or differential semblance.

SYNTHETIC EXAMPLE

We demonstrate the method on a synthetic example consisting of several dipping reflectors embedded in laterally varying slowness. Figure 1 shows from top to bottom the correct slowness model, the stacked reflectivity model and a few selected angle-gathers. We use this model to create a synthetic dataset.

Figure 1: Correct model: slowness (top), stacked image (middle) and selected angle-gathers (bottom).

Figure 2 shows from top to bottom the background slowness model, the stacked reflectivity and a few selected angle-gathers. Since we do not use the correct slowness, the angle gathers are not flat and the image is distorted.

Figure 3 shows from top to bottom the slowness perturbation between the true and the background slowness models, and the image perturbation created using the forward linear
WEMVA operator: the stacked image in the middle panel and a few selected angle-gathers (bottom). This image is, by definition, consistent with the Born approximation. In practice, however, we need to go backward and compute a slowness perturbation from an image perturbation.

The problem with the Born approximation is illustrated in Figure 4. The image perturbation obtained as a difference between the background image and an improved version of it is presented in the bottom two panels. The phase difference between corresponding events is larger than a fraction of the wavelet, and clearly violates the Born approximation. The inverted slowness anomaly (top panel) shows the characteristic sign changes usually seen in wave-equation tomography when the limits of the Born approximation are violated.
Figure 5 illustrates our method for computing the image perturbation from the background data. We run residual migration as indicated in the preceding section and then pick at every location in the image the best velocity ratio $\rho$ which corresponds to flat gathers. We also compute and image derivative according to Equation (12) without the $\Delta \rho$ scaling. Figure 5 shows the stacked image derivative and a few selected angle-gathers. The shape of this image is similar to that of the background image, with some phase and amplitude differences introduced by the derivative process.

We use Equation (12) and the $\Delta \rho$ weight (Figure 5) to create the differential image perturbation (Figure 6). This image perturbation is comparable in shape and magnitude to the ideal perturbation (Figure 3). This indicates that we have succeeded in computing from the
background image an image perturbation which is consistent with the Born approximation, and, therefore, we can use the linearized WEMVA operator without the danger of divergence due to images going out of phase.

For comparison, Figure 7 shows, from top to bottom, the image perturbations computed with the forward WEMVA operator, the one computed as a simple difference between the background image and an improved version of it, and the one computed by our differential procedure. We observe in the middle panel events which are largely out of phase, indicating that we cannot use the Born linearization. In contrast, the differential image perturbation is fully consistent with the one computed by the forward operator.
We use the image perturbation depicted in the bottom two panels of Figure 6 to invert for the corresponding slowness perturbation. We use 10 linear iterations for this example, with only one non-linear iteration.

Finally, Figure 8 shows from top to bottom the updated slowness model, and the updated stacked image in the middle panel and a few selected angle-gathers in the bottom panel, which are much flatter than the ones of the background image (Figure 2).

The key to the success of WEMVA inversion is in the appropriate definition of the image perturbation. The linear WEMVA operator does not account for reflector movement away from the reference image. Therefore, if the updated image incorporates such movement, we risk breaking the Born limitations quite easily, as illustrated in Figure 4. The differential image perturbation operator, however, does not pose such a risk, since we construct the image perturbation on top of the reference image, in compliance with the Born approximation.

Figure 8: Estimated model after WEMVA: slowness (top), stacked image (middle) and selected angle-gathers (bottom).

FIELD DATA EXAMPLE

We also exemplify our method with a field data example. The data corresponds to a complicated North Sea salt environment (Vaillant and Sava, 1999), although the region we have selected for our initial analysis is away from the main salt body. We have selected a 2-D line from the 3-D data, although our 2-D assumption for this region is only partially correct (Clapp, 2001).

Figure 9 depicts the migrated image obtained using our benchmark velocity model. We use this image to relate all our velocity analysis results. Since this velocity model is not perfect, the migrated image is not perfect, either. Various gathers show substantial moveout, particularly inside the “bowl” (around depth 2500 – 3000 m).
Since our benchmark velocity model is not perfect anyway, and given the intrinsic non-linearity of migration velocity analysis, we decided to back away from this model and use a heavily smoothed version of it as our background model. Figure 10 shows the smoothed model and the corresponding migrated image. The angle-gathers clearly indicate slowness inaccuracies which we try to correct using our WEMVA with differential image perturbations method.

Following the strategy used for our synthetic model, we run residual migration for a wide range of velocity ratios, and then pick at every location the value which corresponds to the flattest gathers. We compute the image perturbation using the differential equation (12), and scale it with the picked residual migration ratio. We use the image perturbation in Figure 11 to invert for the slowness perturbation. We run 1 non-linear iteration and 9 linear iterations to obtain the slowness perturbation presented in the right panel of Figure 11.

Finally, we take the inverted image perturbation in Figure 11 and update the background slowness in Figure 10 to obtain the slowness model in Figure 12. This figure also shows the image obtained by migrating the data using this updated slowness model. Comparing the background and improved slowness models in Figures 10 and 12, we observe improved flatness in the upper part of the image (around depth 1500 – 2500 m). We also observe better
definition of the “bowl” (around depth 2500 – 3000 m). However, the bottom-right corner of the image degrades slightly after inversion, possibly as a result of boundary effects or of poor picking during the residual migration step.

Comparing our improved slowness model and the benchmark model in Figures 12 and 9, we also observe a few interesting differences. Again, the “bowl” (around depth 2500 – 3000 m) is better defined using our improved slowness model, although the upper part of the model is still flatter in the image obtained with the benchmark model.

**CONCLUSIONS**

We present an extension of our recursive wave-equation migration velocity analysis method operating in the image domain. Our method is based on the linearization of the downward continuation operator that relates perturbations in slowness to perturbations in image. The fundamental idea is to improve the quality of the slowness function by optimizing the focusing of the migrated image, and not by fitting the data recorded at the surface directly.
We construct the image perturbations by a differential operator applied to the reference image. In this way, we ensure that we do not violate the inherent Born approximation made in our method. This method directly constructs the image perturbation from the background image, and is always compliant with the Born approximation which is the underlying assumption of WEMVA. We show that we can obtain slowness anomalies that are fully consistent with those obtained by the application of the forward and adjoint WEMVA operators.

REFERENCES


Figure 12: Updated model. Migrated image (left) and slowness model (right).


Huang, L., Fehler, M. C., and Wu, R. S., 1999, Extended local Born Fourier migration method: Geophysics, 64, no. 5, 1524–1534.


Short Note

Steering filters in 3-D: Tubes rather than planes

Robert G. Clapp and Marie L. Clapp

INTRODUCTION

An operator composed of non-stationary plane-wave destruction filters, called a steering filter, has practical applications to many problems. Clapp et al. (1997) demonstrated their use for the missing data problem. Fomel (2000) showed how they could be used for signal-noise separation. In several papers (Clapp and Biondi, 1998, 2000; Clapp, 2001), they are used for regularizing tomography. Prucha and Biondi (2002) used them to better handle the null-space in wave equation migration.

Several different methods have been suggested for constructing the 2-D representation of these filters. Clapp et al. (1997) proposed three methodologies: approximating a given dip by interpolating a triangle centered at a given dip or doing Lagrange and Pade interpolation for a given dip. The first method suffered from significant dispersion while all three were severely limited in the angular range that they could accurately describe. To address these and other issues, Fomel (2000) suggested building the filters by doing a Taylor series expansion around zero frequency. The resulting filters were not limited in the angles they could represent and had significant better dispersive properties but the filters were not causal, therefore they could not be used to precondition a model to speed up convergence (Fomel et al., 1997).

Constructing a 3-D steering filter poses additional problems. Clapp (2000) suggested cascading two 2-D steering filters. The resulting response had approximately the right dip in both the x-z and y-z directions but was fairly inaccurate at other angles (Fomel, 2001). Fomel (2001) suggested that a better method was to form the autocorrelation of the combined filter then use Wilson-Burg factorization (Sava et al., 1998) to obtain a causal representation of the filter. Combining these filters produced a 3-D steering filter. This methodology is effective but extremely computationally intensive.

In this paper, we suggest another way to construct a specific class of 3-D steering filters, ones with a significant angular bandwidth. These filters are ideal for tomography or other slow-varying functions. The key idea behind our approach is to transform our filter’s Cartesian coordinate system to its spherical equivalents. We then estimate the filter coefficients given our desired dip and azimuth bandwidth. The resulting filters are able to handle almost any

\[email: bob@sep.stanford.edu, marie@sep.stanford.edu\]
dip-azimuth combination, with minimal dispersion and angle errors. In this paper we begin by showing how to build these classes of filters in 2-D and 3-D. We then use the filters in a simple missing data problem.

**METHODOLOGY**

The purpose of a non-stationary dip filter is to describe the covariance of a field that has a single dip at each location but may vary spatially. We can think of the dip at each location in terms of its normal and describe that normal in our choice of coordinate systems. An obvious choice for our coordinate system is the Cartesian mesh of our model. Another choice is polar coordinates (2-D) or spherical coordinates (3-D).

**2-D construction**

Let us begin by simplifying our problem and think of how to construct a 2-D filter that will describe a specific dip. We need to begin by transforming our filter into polar coordinates. Figure 1 shows a filter in Cartesian coordinates. Think of each grid cell as representing a vector offset \( x \) from the center of our fixed filter coefficient to the center of the given grid cell. We can easily transform the individual grid cells into corresponding \( \theta_i \) values. We can build a filter that tends to create a trend at a given \( \theta \) value by finding the \( \theta_i \) values that bound our desired \( \theta \) value (\( \theta_i^+ \) and \( \theta_i^- \)) and then define our filter coefficient amplitudes \( f \) by,

\[
 f(\theta_i^-) = -\frac{\theta - \theta_i^-}{\theta_i^+ - \theta_i^-} \\
 f(\theta_i^+) = -\frac{\theta_i^+ - \theta}{\theta_i^+ - \theta_i^-}. \tag{1}
\]

The problem with this construction is that the angular bandwidth of the filter will vary wildly. At 0, 45, 63, .. the inverse filter response will be basically a line while for other dips it will have significant aperture. Clapp et al. (1997) shows that by choosing a higher order interpolation scheme, the angular bandwidth problem can be reduced. We can also solve much of this problem by focusing on our goal to construct a specific type of a filter, one for velocity estimation. Velocity is generally a slow-varying function and we aren’t going to have perfect knowledge of its gradient. As a result we want a filter with some angular bandwidth that we have some control over. Let us imagine we want the angular bandwidth to be some quantity \( \alpha \). Instead of interpolating a filter for a single dip we will interpolate for a range of dips between \( \theta - \alpha \) to \( \theta + \alpha \) scaled by a function, we choose \( \cos(\theta_i - \theta)^2 \), that gives prominence to the closest to our desired dip. Our filter construction then becomes a loop over \( \theta_k \) values from \( \theta - \alpha \) to \( \theta + \alpha \) range at some \( \Delta \theta \) step. At each \( \theta_k \) value we find the bounding dip locations so that

\[
 f(\theta_i^+) = f(\theta_i^-) + \frac{\theta_k - \theta_i^-}{\theta_i^+ - \theta_i^-} \cos(\theta_k - \theta)^2 \tag{3}
\]
Figure 1: An initial filter shape is designed to include the fixed coefficient (1) and several potential filter coefficients. The center of each filter is then used to assign a dip value for each potential coefficient.

\[
f(\theta_i^+) = f(\theta_i^+) + \frac{\theta_i^+ - \theta_k}{\theta_i^+ - \theta_i^-} \cos(\theta_k - \theta)^2. \tag{4}
\]

We then normalize the filter coefficients by

\[
f(\theta_i) = \frac{f(\theta_i)}{\sum_{\theta_i} f(\theta_i)}. \tag{5}
\]

Figure 2 shows the filter response of a filter oriented at 30 degrees with lines showing the bounding \(\alpha\) range. Note how the function dies off smoothly towards the boundaries. Figure 3 shows a range of filter responses where \(A\) is polynomial division with the filter. Note how the angular bandwidth is quite consistent in the entire range.

This construction method has another advantage. Larger angles can be effectively represented. Figure 1 shows our initial filter construction. This filter can effectively handle dips between \(-70\) to \(90\) degrees. By extending the second column of the filter a wider angle range can be effectively handled.

### 3-D construction

Up to this stage, everything that has been shown is very similar to the filter construction in previous works (Clapp et al., 1997; Fomel, 2000). In 3-D, we have a fundamental shift in philosophies. In previous works, Clapp (2000, 2001) and Fomel (2001) had the goal to create a filter whose response was a plane oriented at a given direction. Instead we are going to construct a tube. There is a definite disadvantage to this approach. The resulting filter will
not be appropriate for doing interpolation because it is going to be spreading information in a limited area.

The filter construction in 3-D is simply adding an extra dimension to the procedure described above for the 2-D case. Our potential filter coefficients are generally oriented one plane away from the fixed coefficient (Figure 4) and range in the first axis in a similar manner as shown in Figure 1. Our weighting function becomes \( \cos(\theta - \theta_i)^2 \cos(\phi - \phi_i)^2 \), where \( \phi \) is our desired azimuth and \( \phi_i \) is the azimuth at the given filter location. Finally, our linear interpolation is 2-D rather than 1-D. Figure 5 shows the filter response at different dip-azimuth combinations. The response is somewhat deceiving. What we are creating is a tube oriented at a given dip-azimuth combination so the response is only symmetric along the dip-azimuth line.

**EXAMPLE**

To test the methodology we performed two different interpolation problems, one that we expect to succeed, the other to fail. The first is a simple 2-D interpolation problem, the top-left panel of Figure 6 shows our model. We assume some *a priori* knowledge of the dips of the field (top-right panel of Figure 6) and a sparse set of known data traces (bottom-left panel of Figure 6). We apply the fitting goals for a model preconditioned estimation problem described in Claerbout (1999),

\[
d \approx JA^{-1}p. \tag{6}
\]

In this case \( d \) is the known values interpolated on to our model space (bottom-left panel of Figure 6), \( J \) is a diagonal matrix of ones (where we have known data points) and zeros where we do not. \( A^{-1} \) is polynomial division with a steering filter operator constructed using the above methodology. Our preconditioned variable \( p \) is related to our model through \( p = Am \). The bottom right panel of Figure 6 shows the interpolated result. Even with this very sparse dataset we have done a good interpolation of the field.

The 3-D case is a different story. Remember that we are creating tubes rather than planes. The area over which they act is limited. With a sparse set of data points we expect to see tubes
Figure 3: Impulse response of the 2-D filter at several different angles.
Figure 4: The front face of the cube shows where the fixed coefficient are located. The intersection of the As, Bs, and Cs represent the variable coefficient locations.

form around our known data points. The results bear out this hypothesis. Figure 7 shows the known model, Figure 8 our initial data, and Figure 9 our interpolation result. As anticipated what we have ended up doing is forming tubes that curve with our predefined dip and azimuths around the known traces.

CONCLUSIONS

We demonstrated another way to form 3-D plane-wave annihilation filters. The filters do not suffer from the inaccuracies and/or computational expense of previous methods. The main difference from previous constructions is that they form tubes rather than planes. This fact makes them effective regularizers for problems like tomography but inappropriate for interpolation.

REFERENCES


Clapp, R. G., and Biondi, B. L., 2000, Tau tomography with steering filters: 2-D field data example: SEP–103, 1–19.


Figure 5: Impulse response of the 3-D filter at several different angles and azimuths.
Figure 6: A 2-D interpolation example. The top-left panel is the model. Top-right panel is the dip field used to construct the filters. The bottom-left panel is our initial data. The bottom-right panel is our interpolation result. Generally the result is quite accurate.
Figure 7: The correct model for 3-D interpolation problem.

Figure 8: Our starting set of traces.
Figure 9: The final interpolation result. Note how we have formed tubes around our known traces.


**Short Note**

**Semblance picking**

*Robert G. Clapp*\(^1\)

---

**INTRODUCTION**

Tomography in the post-migrated domain is quickly becoming the standard approach for obtaining velocities in complex areas. The standard flow to update the velocity through tomography in the post-migrated domain begins by migrating the data using some initial velocity model. This migration method can be Kirchhoff (Etgen, 1990), downward continuation methods (Clapp, 2001b; Mosher et al., 2001), or any other method that can produce an image as a function of offset or reflection angle. Some type of moveout analysis is then performed on the migrated volume. This moveout analysis can take the form of one (Etgen, 1990; Clapp, 2001b) or two (Biondi, 1990) parameter curvature analysis or something more complex like a residual migration scaling parameter (Etgen, 1990; Clapp, 2002). These moveouts are then converted to time errors. We approximate the non-linear relationship between slowness and travel times by linearizing around an initial slowness model. We can then solve the resulting linear system and update our slowness field.

Until recently, computer power limitations forced some shortcuts in the ideal processing flow. One of these shortcuts was to use a sparse set of gathers rather than the entire volume. Another was to limit moveout analysis was often limited to a set of *a priori* chosen reflectors. Both of these limitations have drawbacks. Etgen et al. (2002) showed that an improved measure of residual moveout can be achieved by calculating moveout at all image locations and then smoothing along reflectors. In addition, Woodward et al. (1998); Clapp (2001a) showed that not being limited to a sparse set of picked reflectors can improve the inversion result and limit the amount of human processing time. These two results indicate that what we would ideally like to do is obtain a field of moveout parameters. The problem is that our data has one more dimension (offset/angle) than our model. This is the same problem faced in NMO velocity analysis (Toldi, 1985). One of the best solution to this type of problem is the Differential Semblance Operator (DSO) approach (Symes and Carazzone, 1991). The problem with a DSO type approach is that it is quite expensive and sensitive to noise.

In this paper I demonstrate an alternate method to estimate the semblance field. The general approach is to solve the non-linear problem through a sequence of simple linear problems.

---

*1email:* bob@sep.stanford.edu
I pick a maximum semblance value at every location. I then form a simple inversion problem to estimate a smooth field of moveout parameters weighted by image and semblance strength. I pick a new maximum in a range around the estimated semblance, estimate a new model and repeat until a satisfactory model is obtained. As the non-linear iterations increase, I decrease the smoothing criteria and range around the estimated semblance.

**REVIEW**

The standard methodology used to update our migration velocity estimate is to start from the fact that the position of a reflection should be independent of the angle at which it is illuminated. As a result, if we look at a common reflection point gather, the reflection depth should be constant as a function of angle (downward continuation methods) or offset (Kirchoff and Gaussian Beam methods). Figure 1 shows the result of migrating a 2-D marine set with a velocity that is significantly in error. When the depth of a reflector is not constant (left panel of Figure 2) as a function of our independent variable (angle/offset) it is indicative of some error in our migration velocity. We set up an inversion problem that relates the deviation from constant depth $dz$ to some change in slowness $\Delta s$ through a linear approximation $T$,

$$dz \approx T \Delta s. \tag{1}$$

We have many choices of how to describe $dz$. We could use an auto-picker that attempts to follow reflections. The problem with this approach is that auto-pickers, especially in complex data, have trouble following events and are susceptible to cycle skip problems. A more common approach is to attempt to describe the moveout as a function of offset/angle by two (Biondi, 1990) or more commonly one parameter (Clapp, 2001b; Etgen, 1990; Biondi and Symes, 2003). The normal procedure is to scan over the parameter space forming semblance gathers (right panel of Figure 2). Normally these semblance gathers are calculated on a sparse set of CRP gathers and the maximum extracted at pre-determined reflector locations.

Selecting a sparse set of CRP gathers and selecting the maximum can give misleading results. If a selected CRP gather has a low signal-noise ratio at the reflector location we can get misleading information. Also, if our CRP analysis points are too sparse we can miss important features in the dataset. Finally, the maximum, especially if automatically selected, can be a dangerous choice. If the semblance has multiple distinct peaks (Figure 3) the maximum might indicate a spurious moveout. Clapp (2001b); Etgen et al. (2002) took the approach that a better solution is to do the semblance analysis at each CRP location and then smooth the semblance values along the reflectors. This approach is a definite improvement over the conventional approach but still does not effectively handle the core problem that our data space, moveout as a function of angle/offset, is one more dimension than our model, maximum semblance, and writing a linear relationship between the two is problematic.
Figure 1: The top panel shows the migrated image using the zero offset imaging condition. The bottom panel shows every 20th angle gather of the same dataset. Note the significant moveout in the angle gathers indicating velocity errors.
Figure 2: The left panel shows a CRP gather from a 2-D dataset shown in Figure 1. Note that the moveout varies as a function of angle indicating an error in the velocity model. The right panel is a semblance gather calculated from the CRP gather in the right panel. Gamma is a scaling parameter applied to the velocity field, a scaling of 1 indicates a flat gather.

Figure 3: Semblance in complicated portion of the data. Note the 3 distinct peaks. In this case the peak with the maximum amplitude is the result of noise in the data rather than the moveout of the reflector.

GLOBAL SOLUTION

This type of estimation problem is well known in the literature for NMO velocity analysis (Toldi, 1985; Symes and Carazzone, 1991). One solution to the NMO velocity analysis problem, presented in Clapp et al. (1998) was to:

- Calculate semblance in a gather.
- Pick the maximum within some reasonable fairway.
- Solve the estimation problem:

\[ 0 \approx W(d - Cm) \]
\[ 0 \approx \epsilon Am, \]  \hspace{1cm} (2)

where \( d \) is \( V_{rms}^2 \), \( m \) is \( V_{int}^2 \), \( C \) is causal integration, \( W \) is the semblance at the given location or stack-power, \( \epsilon \) is our control over smoothness, and \( A \) is some roughening operator.
The advantage of this approach is that it is efficient and the $W$ appropriately gives reflectors with higher coherence more weight in the inversion. However, this approach is still susceptible to erroneous maximum picks.

We can write a similar set of fitting goals for picking the best semblance for our tomography problem,

$$\begin{align*}
0 & \approx W(d - m) \\
0 & \approx \epsilon Am.
\end{align*}$$

In this case $d$ is the gamma value of the maximum semblance at each location and $m$ is some smoothed version of it. Our $W$ operator can be semblance, stack power, or some other measure of data quality. I found that using a combination of the envelope of the stack and semblance gave the best result,

$$W = SE,$$

where $S$ is the maximum semblance at a given location, and $E$ is the amplitude of the envelope of the migration. Figure 4 shows the maximum semblance for a 2-D marine dataset. Note the highly variable nature of the maximums. Figure 5 shows the envelope of the image, the maximum semblance value, and the weight operator (the $E$, $S$, and $W$ arrays in (4)). Figure 6 shows the resulting smooth semblance values. The resulting model is smoother and more reasonable than the unsmoothed field (Figure 4), but still has problems. Specifically, some of our original data picks are suspicious. If you look at the top-left portion of Figure 4 you will note large moveout values at shallow depths which seem unreasonable. In addition, Figure 7 shows the CRP gather at $x = 7$, note the events at about $3\ km$. They look flat, but if we look at both the data (Figure 4) and our model (Figure 6) we see significant moveout. Both areas suffer from the poor data pick problem mentioned above.

We can add another twist to the problem of bad maximums by repicking our maximum semblance values to be within some tolerance range around our estimated semblance values, then resolving the inversion problem. Figure 8 shows our new data picks limiting the range around estimated model. We can repeat this procedure several times continually decreasing the range around which we will accept new data values. This approach has similarities to simulated annealing. We begin by searching the entire range of our space, trying to get close
Figure 5: Left panel is the envelope of the migrated image. The center panel is the maximum semblance at each model location. The right panel is a combination of the two. These are the three quantities in equation (4) used in fitting goals (3).

Figure 6: Model estimated using fitting goals (3), the data shown in Figure 4 and the weight function shown in Figure 5.

Figure 7: A CRP gather at X=7.3km. Note the reflection at 3km. The reflection is flat but the artifacts cause the maximum semblance (Figure 4) and the estimated model (Figure 6) to indicate that residual moveout exists.
to the correct solution. Later we limit our search space and try to get the finer features. Using the same analogy, it makes sense to also have $\epsilon$ be a function of iteration. At early iterations we have a large $\epsilon$ and by reducing $\epsilon$ as a function of non-linear iteration we get the finer features of the model. Figure 9 shows the results after repeating this procedure five times. Note the absence of the problem seen in Figure 6. We have successfully honored the data where we have good semblance and strong reflectors without introducing artifacts in regions where data quality is more questionable.

Figure 8: The result of limiting the range of acceptable data values to around a reasonable range defined from solving fitting goals (3) once.

Figure 9: Our final estimated solving fitting goals (3) several times reducing the range of acceptable data values, and decreasing or smoothing constraints.

CONCLUSIONS

Obtaining reasonable moveout measurements is an essential step in migration velocity updating. The problem is at some level non-linear. A viable solution is to solve the problem iteratively by relinearizing the problem several times. Preliminary results are encouraging.

REFERENCES


Clapp, R., 2001a, Ray-based tomography with limited picking: SEP–110, 103–112.


Realization and analysis of an integration tomography scheme

Weitian Chen

ABSTRACT
I realize and analyze a model-based joint tomography scheme in this paper. Surface reflection seismic data and VSP traveltimes are used simultaneously to invert the velocity by using an integrated inversion scheme. Since more data are used, the integrated tomography can obtain more accurate inversion results with lower uncertainty. Using identity operator as integration operator, I apply this method to a synthetic anticline model. Compared to the surface reflection tomography, the integrated inversion result is better in areas where the VSP ray coverage is good and conversely, the integrated inversion result is poorer in the areas where VSP ray coverage is sparse or inexistent. The results suggest we can obtain improved velocity field by this integration tomography scheme if using a well-designed integration operator.

INTRODUCTION
During the past few years, promising case studies have indicated that better subsurface images can be acquired by integrated processing of borehole and surface seismic data than by processing either type of data alone (Chopra et al., 2002; Kostov et al., 1999). A major advantage of this integration is that it helps determine an improved subsurface velocity model. Among different tomography methods, surface reflection tomography has gained popularity in modern 3D seismic projects for use in large-scale projects (Stork and Clayton, 1991; Stork, 1992, 1994; Clapp, 2001). However, surface reflection tomography suffers from inversion ambiguity because it perturbs both reflector position and velocity together to extract the unknown velocity field. Such velocity-depth ambiguity can be substantially reduced if transmission and reflection seismic data are used together for velocity inversion (Mao and Stuart, 1997).

Researches have been conducted to do integration tomography using VSP and surface seismic data simultaneously (Chiu and Stewart, 1987; Cao et al., 2000). Claims have been made that the integration tomography can resolve the subsurface velocity field better than either tomography technique. However, the frequency bandwidth difference, the correlated errors between the two data types (Brown and Clapp, 2001), the difficulty to balance different tomography operations and the challenge of software design make integration tomography remain a topic of considerable academic and practical interest.

In this paper, a model-based joint tomography scheme is realized and analyzed. Sur-
face reflection data and VSP traveltime are used simultaneously to invert the velocity in this integrated inversion scheme. Using the identity operator as integration operator, this integration tomography scheme is applied to a synthetic anticline model. The inversion result from integration tomography is compared to that from surface reflection and VSP tomography, respectively. Compared to surface reflection tomography, the integrated inversion result is better in areas where the VSP ray coverage is good and conversely, the integrated inversion result is poorer in the areas where VSP ray coverage is sparse or inexistent. The results suggest that we can obtain improved velocity field by this integration tomography scheme if using a well-designed integration operator.

**METHODOLOGY**

**Data Calibration**

Data calibration is a necessary pre-processing step for the integrated inversion since the surface reflection data and VSP data not only have random errors, but also may have correlated errors (Brown and Clapp, 2001). The correlated errors must be removed before the data are used together for velocity inversion. In addition, different data sets may be pre-processed by different processors with different standards, which also require us to calibrate different data sets before they are input to tomography solver.

For the synthetic model used in this paper, the data are generated without introducing noise. Therefore, I do not consider data calibration in this paper and leave it for future work.

**General principles of tomography**

Both reflection tomography and transmission tomography start from the idea that we can use the following fitting goal to invert the unknown slowness model $s$ from traveltime $t$:

$$ t = T(s)s $$

The operator $T(s)$ is usually constructed in terms of rays passing through the slowness model $s$, so it is model dependent and therefore non-linear. By doing a Taylor expansion and ignoring the second and higher order terms, we can linearize this problem:

$$ \Delta t = T_0 \Delta s $$

Here $\Delta s$ and $\Delta t$ are the change of slowness and corresponding change of travel time, respectively. The tomography operator $T_0$ now is model-independent. To use the fitting goal (2) for inversion, we need some prior information about the unknown slowness field to construct an initial model, then construct $T_0$ by ray tracing through it. If the initial model does not seriously deviate from the true slowness field, we can use the operator $T_0$ to invert the change of slowness $\Delta s$ from the change of traveltime $\Delta t$. After we obtain $\Delta s$, we can update the initial model to obtain more accurate slowness field.
**Integrated tomography for slowness inversion**

There are many ways to integrate different tomography types for velocity inversion, such as sequential inversion, joint inversion, or recursive inversion. Compared to other integration schemes, joint inversion is a more complex process but also expected to be more efficient. The complexity of joint inversion comes from the difference between the data sets used for inversion, the difficulty of balancing the different tomography operations, and the challenge of software design. The advantage of joint inversion is that different data sets are used simultaneously to invert the velocity. Therefore, the inversion result will be more accurate with lower uncertainty.

In this paper, I realize a joint inversion scheme with data fitting goal (3) and regularization goal (4):

$$0 \approx N \begin{pmatrix} \Delta t_{\text{ref}} - T_{\text{ref}} \Delta s \\ \Delta t_{\text{vsp}} - T_{\text{vsp}} \Delta s \end{pmatrix}$$

$$0 \approx \epsilon A \Delta s$$

Here, $T_{\text{ref}}$ and $T_{\text{vsp}}$ are the reflection and VSP tomography operators, respectively. $\Delta t_{\text{ref}}$ is the traveltime change for surface reflection and $\Delta t_{\text{vsp}}$ is the traveltime change for VSP tomography. Usually, the traveltime is acquired in different ways in reflection tomography and VSP tomography. In reflection tomography, the reflection points are picked from the migration result. Semblance scans are used to obtain the residual moveout in the angle domain common image gathers. The residual moveout is then converted to the corresponding traveltime change. For VSP tomography, traveltime is usually picked directly from the VSP data manually or automatically.

In order to balance the surface reflection tomography and VSP tomography in the integrated inversion, I introduce an integration operator $N$ into the data fitting goal (3). $N$ is efficiently approximated by the inverse noise covariance (Tarantola, 1987) which is difficult to get in practice. A correct design of operator $N$ is critical to balance the two tomography operators in the final integration result. In this paper, I simply use an identity operator as the integration operator.

The regularization operator $A$ can be best approximated by the inverse model covariance that lacks a priori information in reality. I simply use the Laplacian operator as a regularizer in this paper.

**EXAMPLE**

In this section, we will show the integration inversion result of a synthetic anticline model by using fitting goals (3) and (4). The integration tomography result is then compared to the regular surface reflection and the VSP tomography result. In order to make the results from the different tomography methods comparable, I use the same model styling goal (4).
for all three tomography operations. The techniques developed by Clapp (2001) for reflection tomography, such as performing tomography in the tau domain, steering filter regularization, and preconditioning, are not used in this paper. I plan to extend those techniques to integration tomography in a later work.

**Initial velocity model**

The left panel of Figure 1 shows the synthetic model with 7 reflectors overlapped. The challenge for this model is to determine the low velocity zone under the top of the anticline and to image the bottom of the model. The initial velocity I use is a linear $v(z)$ model, as shown in the right panel of Figure 1. Figure 2 shows the migration result using the initial velocity model. We can see the upper part of the model is well imaged, but the lower part is severely mispositioned. The left three panels of Figure 2 shows three angle domain common image gathers (ADCIG) computed using the initial velocity model. The X positions are 8.0, 10 and 12.0 km respectively. Notice the smiling events in the ADCIGs indicating the initial velocity value is too slow at these locations.

![Figure 1: Synthetic anticline velocity model and the reflectors](chen1-fig1)

**Tomography result**

The panels (a), (b) and (c) in Figure 3 show the VSP, the surface reflection, and the integration tomography result, respectively. The identity operator is used as an integration operator for the integration tomography. Panels (d), (e) and (f) show the ray coverage map for the VSP, the surface reflection and the integration tomography, respectively. The gray scales indicate how many rays pass through the corresponding grids. The surface reflection tomography has
Figure 2: The left image shows the prestack migration result using initial velocity model. The overlapped curves are the true reflectors’ position. The right three panels show three ADCIG gathers. From left to right, the x positions are 8.0, 10.0, 12.0, respectively. 

a much denser ray coverage than that for VSP tomography (There are totally 12674 rays used for surface reflection tomography and only 4708 rays used for VSP tomography). For the VSP tomography, the rays fall into an approximately upside-down triangle area. Out of this area, the VSP tomography gives a highly deviated inversion result. Since surface reflection tomography has a wider and denser ray coverage, its inversion result is much better than VSP tomography. The sum of ray coverage for the surface reflection and the VSP forms the ray coverage map for the integration tomography. In the areas at surface location smaller than 2.0 km or bigger than 18.0 km, both surface reflection and VSP have poor ray coverage, which results in an unsatisfied integration tomography result in these areas. However, in the area within the anticline, where the VSP tomography does invert velocity more accurately than surface reflection, the integration tomography has a better inversion result than surface reflection tomography.

Figure 4 shows the migration results using the velocity field obtained by VSP, surface reflection, and integration tomography. The left panel corresponds to the VSP tomography; the center panel corresponds to surface reflection tomography, and the right panel corresponds to the integration tomography. Compared to using the surface reflection tomography, notice that using the integration tomography for migration can image the bottom reflectors more accurately, as indicated by two circles in Figure 4. However, the image using integration tomography is not as accurate as that using surface reflection tomography at the two sides of the model because of the deleterious contribution from VSP tomography in these areas.

Figure 5 shows the three angle domain common image gathers (ADCIG) extracted at same three locations within the anticline when using VSP, reflection, and integration tomography results for migration. The left three panels correspond to VSP tomography; the center three
Figure 3: The images acquired by using different tomographies and the corresponding ray coverage map. a) The image obtained by VSP tomography. b) The image acquired by reflection tomography. c) The image acquired by integration tomography. d) The ray coverage map for VSP tomography. e) The ray coverage map for reflection tomography map. f) The ray coverage map for integration tomography map.
Figure 4: The migration result using VSP, surface reflection, and integration tomography. The left panel corresponds to the VSP tomography; the center panel corresponds to the surface reflection tomography; the right panel corresponds to integration tomography. From left to right within each group, the corresponding surface locations for three ADCIGs are 8.0, 10.0, and 12.0 km, respectively. We can observe the contributions from both VSP and surface reflection tomography in the ADCIGs for integration tomography.

DISCUSSION

In this paper, I implemented an integrated velocity inversion scheme and applied it to a synthetic model. The surface reflection data and VSP data are used simultaneously to invert the slowness model. To integrate the surface reflection and VSP tomography, I simply use an identity operator as the integration operator. Therefore, the VSP tomography and surface reflection tomography have uniform contributions to their integration. Compared to the surface reflection tomography, the integrated inversion result is better in areas where the VSP ray coverage is good and conversely, the integrated inversion result is poorer in the areas where VSP ray coverage is sparse or inexistent.

The integrated inversion result obtained by using an identity operator suggests that we can improve the result if we can obtain a well-designed integration operator. For example, if the integration operator can afford more weights to the tomography (VSP or surface reflection)
where this tomography result is more accurate, the integration tomography can bring more accurate inversion result throughout the model than either tomography alone.

ACKNOWLEDGMENT

I am very grateful to Bob Clapp for his great help to my work. He gave me original version of his reflection tomography codes, the anticline velocity model and data, and many suggestions for my research. Also many thanks to Guojian Shan for giving me his 3D reflection tomography work done in Phillips Petroleum Company last summer and many helpful discussions. Thanks to Morgan Brown for initiating my work in integration velocity analysis. My greatest thanks go to the Stanford Graduate Fellowships (SGF) Program for financial aid toward my study and work at Stanford University through year 2001 to 2004.

REFERENCES


Cao, D., Hirabayashi, N., Leaney, S., Borland, W., and Hara, K., 2000, An integrated 3-D
Joint tomography


The oddities of regularized inversion: Regularization parameter and iterations

Marie L. Clapp

ABSTRACT
Proper imaging in areas with complex overburdens cannot be done effectively with an adjoint operator such as migration. To image in complex areas, we really want to apply an inverse operator, but most imaging problems can be represented by very large matrices that are difficult to invert directly. Therefore, many schemes to approximate an inverse operator have been developed. Regularized least-squares inversion implemented in an iterative scheme can be very effective in dealing with illumination problems when the imaging and regularization operators are well chosen. However, those aren’t the only decisions that need to be made. Both the choice of regularization parameter ($\epsilon$) which balances the data and model residuals and number of iterations ($n_{\text{iter}}$) can have a significant effect on the quality of the final image. In this paper, I describe some of the issues that must be taken into account when choosing $\epsilon$ and $n_{\text{iter}}$ for an imaging problem with poor illumination. I also examine their effects on a simple synthetic data example. These experiments show that the effects of $\epsilon$ and $n_{\text{iter}}$ are related and must be considered when performing an inversion for imaging.

INTRODUCTION

The quest to obtain the best seismic image in areas of poor illumination is leading us down the tricky path of inversion. Our seismic problems are too large to be directly inverted, so we must turn to iterative schemes.

Although migration is not sufficient to image the subsurface in areas with poor illumination (Muerdter et al., 1996; Prucha et al., 1998), we can use migration as an imaging operator in an iterative least-squares inversion scheme (Nemeth et al., 1999; Duquet and Marfurt, 1999; Ronen and Linder, 2000). In areas with poor illumination, the inversion problem is ill-conditioned, therefore it is wise to regularize the inversion scheme (Tikhonov and Arsenin, 1977). The regularization operator can be designed to exploit knowledge we have about the expected amplitude behavior and dip orientation of events in the image (Prucha and Biondi, 2002).

Selecting proper imaging operators and regularization operators is difficult enough, but there are still two more small but important details. We must select an appropriate value for

1email: marie@sep.Stanford.edu
the regularization parameter $\epsilon$ to balance the data and model residuals and decide how many iterations ($n_{iter}$) to carry out for this problem. There is no well established way to decide on either of these, and their values will vary depending on the data and inversion operators.

In this paper, I will explain my methodology for selecting $\epsilon$ and $n_{iter}$ for a regularized inversion using wavefield continuation migration as the imaging operator and a steering filter as the regularization operator. I will begin by explaining the inversion scheme. I will then explain a criteria for an optimum $\epsilon$ and $n_{iter}$. Finally, I will show the effects of $\epsilon$ and $n_{iter}$ on a simple synthetic with well defined illumination problems.

**REVIEW OF REGULARIZED INVERSION**

Least-squares inversion can be expressed simply as the minimization of this objective function:

$$min\{Q(m) = ||Lm - d||^2\}$$

(1)

where $L$ is a linear modeling operator, $d$ is the data, and $m$ is the model. This minimization can be expressed more concisely as a fitting goal:

$$0 \approx Lm - d.$$  (2)

However, in areas of poor illumination, this problem will have a large null space. The null space is partially caused by the fact that our survey can not have infinite extents and infinitely dense source and receiver grids. Any noise that exists within the null space can grow with each iteration until the problem becomes unstable. Fortunately, we can stabilize this problem with Tikhonov regularization (Tikhonov and Arsenin, 1977). The regularization adds a second fitting goal that we are minimizing:

$$0 \approx Lm - d$$

$$0 \approx \epsilon Am.$$  (3)

The first fitting goal is the “data fitting goal,” meaning that it is responsible for making a model that is consistent with the data. The second fitting goal is the “model styling goal,” meaning that it allows us to impose some idea of what the model should look like using the regularization operator $A$. The strength of the regularization is controlled by the regularization parameter $\epsilon$.

Unfortunately, the inversion process described by fitting goals (3) can take many iterations to produce a satisfactory result. We can reduce the necessary number of iterations by making the problem a preconditioned one. We use the preconditioning transformation $m = A^{-1}p$ (Fomel et al., 1997; Fomel and Claerbout, 2002) to give us these fitting goals:

$$0 \approx LA^{-1}p - d$$

$$0 \approx \epsilon p.$$  (4)

$A^{-1}$ is obtained by mapping the multi-dimensional regularization operator $A$ to helical space and applying polynomial division (Claerbout, 1998). After obtaining $p$ from the fitting goals in (4), it is simple to transform back to $m$. Now that our inversion is defined, we can take a closer look at $\epsilon$ and at the number of iterations ($n_{iter}$).
The question of $\epsilon$ and $n_{iter}$

The intention of regularized inversion is to minimize the residuals of both the data fitting goal and the model styling goal. Generally, we think of accomplishing this by estimating the model $p$ so that it best satisfies fitting goals (4). However, the estimated model itself is dependent on $\epsilon$ and $n_{iter}$. Our choice of $\epsilon$ and $n_{iter}$ can have a significant effect on the model we obtain. To explore this idea, I will first examine $\epsilon$, then $n_{iter}$, showing the relationship between them.

Given the fitting goals (4), the direct solution for our model would be

$$p = ((LA^{-1})'LA^{-1} + \epsilon^2 I)^{-1}(LA^{-1})'d$$

so it is clear that the choice of $\epsilon$ will influence the model. Recall that equation (5) cannot be used to solve our seismic inversion problems because the matrices involved are so large. Selecting an $\epsilon$ that is appropriate for a given problem and model is not a trivial task.

It is common to set $\epsilon = 0$ for preconditioned problems. The inversion will still be affected by the regularization operator because it is present as $A^{-1}$ in the data fitting goal. However, it is known that after a number of iterations, $\epsilon = 0$ will cause the $A^{-1}$ not to have an effect on the data fitting goal. Additionally, Crawley (2000) showed that a non-zero $\epsilon$ can improve the convergence rate of an iterative inversion problem. For these reasons, we would like to find an inexpensive way to choose a non-zero $\epsilon$.

Much work has been done on the idea of selecting $\epsilon$ based on L-curve analysis (Calvetti et al., 1999, 2000). Essentially, an L-curve is created by plotting the residual of the model styling goal versus the residual of the data fitting goal using different values of $\epsilon$. This curve is shaped like an L. The vertex of the L is the point that will minimize both of the residuals. This point tells us what the ideal $\epsilon$ is - assuming that we iterate the problem to convergence. This brings us to the question of the number of iterations ($n_{iter}$).

Ideally, an iterative least-squares inversion should be allowed to iterate to convergence, that is, until the error residual is not getting any smaller. Paige and Saunders (1982) show that when we use a conjugate gradient scheme to minimize our problem, it is guaranteed to converge in no more iterations than there are parameters being inverted for. Preconditioning the problem can reduce this number, but even then our seismic inversions are too large to allow them to iterate to convergence. Therefore, we have to decide how many iterations are needed given our available computer resources. We are forced to decide when a model is “good enough”. Naturally, since we are not iterating to convergence, our L-curve will be different so selecting an $\epsilon$ is now dependent on the choice of $n_{iter}$. In the next section, I will demonstrate the effects of $\epsilon$ and $n_{iter}$ for a synthetic imaging problem.

Effects of $\epsilon$ and $n_{iter}$

In order to understand $\epsilon$ and $n_{iter}$, I decided to experiment with them on a small synthetic seismic inversion problem. In this section I will explain the imaging and regularization operators I choose to use, the synthetic data, and the results of different $\epsilon$ and $n_{iter}$. 
Inversion operators

The imaging operator I chose for this experiment is the downward continuation migration described by Prucha et al. (1999). This operator downward continues the wavefield, slant stacks at each depth step and then extracts an image at zero time. The regularization operator I used was a simple gradient operator along the offset ray parameter axis. This regularization operator assumes that the imaging is being done with the correct velocity model so that the events along the offset ray parameter should be flat. It will also reduce sudden variations in amplitude along the ray parameter axis.

Data

It is difficult to understand the effects of $\epsilon$ and iterations when used for the inversion of real seismic data. Therefore, I chose to use a simple synthetic dataset provided to us by Bill Symes at The Rice Inversion Project (TRIP). It consists of one flat layer beneath a slow velocity lens. The velocity model can be seen in Figure 1. The expected amplitude response from the flat reflector with regard to the offset ray parameter is constant.

Results

Given the velocity model, we expect to see symmetrical shadow zones under the edges of the low velocity lens. They are evident in the result of our downward continuation migration (Figure 2). They are the most clear in the depth slice (the top panel), enclosed by black ovals, as narrow black “V”s. They can be seen as a slight dimming in two places along the event in the common ray parameter section (lower left panel). The common image gather (lower right panel) also shows a gap at high offset ray parameters.

To examine the effects of $\epsilon$ and $n_{iter}$, I ran four experiments. For the first, I set $n_{iter} = 3$ and $\epsilon = 1$. This result can be seen in Figure 3. The shadow zones are still visible inside the black ovals but have begun to fill in. The artifacts at the edges of the event (inside the black circle) are reduced and the imaging artifacts above the flat event are also diminished.
Figure 2: Result of downward continuation migration. The display is a flattened cube in which the top panel is a depth slice, bottom left panel is a common ray parameter section, bottom right is a ray parameter gather. The shadow zones can be seen in the depth slice as thin black “V”s inside the 2 black ovals. The black circle (uppermost) indicates strong artifacts.

Figure 3: Result of inversion with 3 iterations and $\epsilon = 1$. The display is a flattened cube in which the top panel is a depth slice, bottom left panel is a common ray parameter section, bottom right is a ray parameter gather. There is little change in the shadow zones (black ovals) but the artifacts are reduced (black circle).
The second experiment set $n_{\text{iter}} = 3$ and $\epsilon = 10$. This result (Figure 4) shows that the shadow zones are mostly filled in. This combination of $\epsilon$ and $n_{\text{iter}}$ seems to be a good choice.

Figure 4: Result of inversion with 3 iterations and $\epsilon = 10$. The display is a flattened cube in which the top panel is a depth slice, bottom left panel is a common ray parameter section, bottom right is a ray parameter gather. The shadow zones are largely filled in (inside the black ovals). The artifacts inside the black circle are diminished.

The third experiment had $n_{\text{iter}} = 12$ and $\epsilon = 1$. Figure 5 shows an unexpected result. The shadow zones have actually become larger. I believe that the high number of iterations has allowed the data fitting goal to clean up artifacts that creep into the shadow zones because of approximations in the imaging operator. The low $\epsilon$ keeps the regularization operator from trying to fill the shadow zone.

Figure 5: Result of inversion with 12 iterations and $\epsilon = 1$. The display is a flattened cube in which the top panel is a depth slice, bottom left panel is a common ray parameter section, bottom right is a ray parameter gather. The shadow zones are actually getting larger (inside the black ovals) as the high number of iterations cleans up artifacts while the low $\epsilon$ doesn’t help fill the shadow zones.

The final experiment used $n_{\text{iter}} = 12$ and $\epsilon = 10$. Figure 6 shows that this combination has filled the shadow zones nicely. Unfortunately, there is also some noise introduced, faintly visible as a wide vertical stripe in the center of the common ray parameter section.
Figure 6: Result of inversion with 12 iterations and $\epsilon = 10$. The display is a flattened cube in which the top panel is a depth slice, bottom left panel is a common ray parameter section, bottom right is a ray parameter gather. The shadow zones are filled in (inside the black ovals), although this combination of iterations and $\epsilon$ seems to be causing noise away from the reflector.

CONCLUSIONS

I have shown that the regularization parameter $\epsilon$ and the number of iterations $n_{iter}$ can have a significant effect on the result of regularized least-squares inversion. The choice of $n_{iter}$ generally depends on extraneous concerns such as available computer power and the best choice for $\epsilon$ depends on the choice of $n_{iter}$. For the experiments carried out in this paper, the best combination was $n_{iter} = 3$ and $\epsilon = 1$, but it is not likely that this is the “ideal” combination even for this problem, let alone other inversion problems.

ACKNOWLEDGMENTS

I would like to thank Bill Symes and TRIP for the synthetic dataset used in this paper.

REFERENCES


Amplitude and kinematic corrections of migrated images for non-unitary imaging operators

Antoine Guittton

ABSTRACT

Obtaining true-amplitude migrated images remains a challenging problem. One possible solution to address it is iterative inversion. However, inversion is an expensive process that can be rather difficult and expensive to apply, especially with 3-D data. In this paper, I propose computing an image that is close to the least-squares inverse image by approximating the Hessian, thus avoiding the need for iterative inversion. The Hessian is approximated with non-stationary matching filters. These filters are estimated from two images: one is the migration result ($m_1$) and the second is the migration result of the re-modeled data computed from $m_1$. Tests on the Marmousi dataset show that this filtering approach gives results similar to iterative least-squares inversion at a lower cost. In addition, because no regularization was used in the inversion process, the filtering method produces an image with fewer artifacts. Applying this method in the angle domain yields similar conclusions.

INTRODUCTION

It is well known that most of our mathematical operators for seismic processing are not unitary. This means that for any linear transformation $L$, $L'L \neq I$ where $(')$ stands for the adjoint and $I$ is the identity matrix. Having non-unitary operators often results from approximations we make when building those operators. For example, we might not take the irregular and finite acquisition geometry of seismic surveys into account.

For migration, different approaches exist to correct for these defects of our operators. Bleistein (1987), based on earlier works from Beylkin (1985), derived an inverse operator for Kirchhoff migration. A similar path is followed by Thierry et al. (1999) with the addition of non-linear inversion with approximated Hessian. Alternatively, least-squares migration with regularization has proven efficient with incomplete surface data, e.g., Nemeth et al. (1999) and illumination problems due to complex structures, e.g., Prucha et al. (1999); Kuehl and Sacchi (2001). Hu et al. (2001) introduce a deconvolution operator that approximates the inverse of the Hessian in the least-squares estimate of the migrated image. However, this method assumes a $v(z)$ medium which means that the deconvolution operators are horizontally invariant. Recently, Rickett (2001) proposed estimating weighting functions from reference

$^1$email: antoine@sep.stanford.edu
images to compensate illumination effects for finite-frequency depth migration. This method corrects for amplitude effects only and requires some smoothing that can be rather difficult to estimate.

In this paper, I propose a new strategy for approximating the inverse of the Hessian. This approach aims to estimate a bank of non-stationary matching filters (Rickett et al., 2001) between two migrated images that theoretically embed the effects of the Hessian. This approach is implemented after migration and is relatively cheap to apply. It can be applied on images migrated at zero-offset or in the angle domain. I illustrate this method with the Marmousi dataset. I demonstrate that this approach can effectively recover amplitudes similar to the one obtained with least-squares inversion. In addition, these filtered images have less artifacts than the least-squares result without regularization and at a much reduced cost.

THEORY

In this section, I show how the least-squares estimate of a migrated image can be approximated using non-stationary matching filters. In terms of cost, this approach is comparable to one and a half iterations of a conjugate-gradient method (CG), the first “half” iteration being the migration. The cost of estimating the non-stationary filters is negligible compared to the total cost of migration.

First, given seismic data $d$ and a migration operator $L$, we seek a model $m$ such that

\[ Lm = d. \]  

This goal can be rewritten in the following form

\[ 0 \approx r_d = Lm - d \] 

and is called the fitting goal. For migration, a model styling goal (regularization) is necessary to compensate for irregular geometry artifacts and uneven illumination (Prucha et al., 1999). I omit this term in my derivations and focus on the data fitting goal only. By estimating $m$ in a least-squares sense, we want to minimize the objective function

\[ f(m) = \|r_d\|^2 = \|Lm - d\|^2. \]

The least-squares estimate of the model is given by

\[ \hat{m} = (L'L)^{-1}L'd \]

where $L'L$ is the Hessian of the transformation. My goal in this paper is to approximate the effects of the Hessian $L'L$ using non-stationary matching filters.

Approximating the Hessian

In equation (4), I define $L'd$ as the migrated image $m_1$ after a first migration such that

\[ \hat{m} = (L'L)^{-1}m_1. \]
In equation (5), \( \hat{m} \) and \( L'L \) are unknown. Since I am looking for an approximate of the Hessian, I need to find two known images that are related by the same expression as in equation (5). This can be easily achieved by remodeling the data from \( m_1 \) with \( Ld_1 = Lm_1 \) (6)

and remigrating them with \( L' \) as follows:

\[
m_2 = L'd_1 = L'Lm_1.
\] (7)

Notice a similarity between equations (5) and (7) except that in equation (7), only \( L'L \) is unknown. Notice that \( m_2 \) has a mathematical significance: it is a vector of the Krylov subspace for the model \( \hat{m} \). Now, I assume that we can write the inverse Hessian as a linear operator \( B \) such that

\[
\hat{m} = Bm_1
\] (8)

and

\[
m_1 = Bm_2.
\] (9)

Equation (9) can be approximated as a fitting goal for a matching filter estimation problem where \( B \) is the convolution matrix with a bank of non-stationary filters (Rickett et al., 2001). This choice is rather arbitrary but reflects the general idea that the Hessian is a transform operator between two similar images. My hope is not to “perfectly” represent the Hessian, but to improve the migrated image at a lower cost than least-squares migration. In addition in equations (8) and (9), the deconvolution process becomes a convolution, which makes it much more stable and easy to apply. Hence, I can rewrite equation (9) such that the matrix \( B \) becomes a vector and \( m_2 \) becomes a convolution matrix (Robinson and Treitel, 1980):

\[
m_1 = M_2b.
\] (10)

The goal now is to minimize the residual

\[
0 \approx r_{m_1} = m_1 - M_2b
\] (11)

in a least-squares sense. Because we have many unknown filter coefficients in \( b \), I introduce a regularization term that penalizes differences between filters as follows:

\[
0 \approx r_{m_1} = m_1 - M_2b
0 \approx r_b = Rb
\] (12)

where \( R \) is the Helix derivative (Claerbout, 1998). The objective function for equation (12) becomes

\[
f(b) = \|r_{m_1}\|^2 + \epsilon^2\|r_b\|^2.
\] (13)

where \( \epsilon \) is a constant. The least-squares inverse is thus given by

\[
\hat{b} = (M_2'M_2 + \epsilon^2R'R)^{-1}M_2'm_1.
\] (14)

Once \( \hat{b} \) is estimated, the final image is obtained by computing

\[
\hat{m} = m_1 * \hat{b}
\] (15)
where (*) is the convolution operator.

Therefore, I propose computing first a migrated image $m_1$, then computing a migrated image $m_2$ (equation (7)), and finally estimating a bank of non-stationary matching filters $b$, e.g., equation (12). The final improved image is obtained by applying the matching filters to the first image $m_1$, e.g., equation (8). In the next section, I illustrate this idea with the Marmousi dataset. I show that an image similar to the least-squares migration image can be effectively obtained.

**MIGRATION RESULTS**

I illustrate the proposed algorithm with the Marmousi dataset. I use a prestack split-step migration method with one reference velocity. I demonstrate with zero-offset images and angle gathers that the approximation of the Hessian with adaptive filters gives a result comparable to least-squares migration with fewer artifacts.

**Zero-offset prestack migration results**

I first show in Figure 1 a comparison between the migration result of the Marmousi dataset ($m_1$) and the remodeled data ($m_2$). We notice that the migration of the remodeled data (Figure 1b) lowers the amplitudes in the upper part of the model. Therefore, we expect the filters to correct for this difference. Figure 2 displays few estimated filters for the Marmousi result. The filters are ten by ten with 40 patches in depth and 80 along the midpoint axis. I shown only a fifth of these filters in both axes. It is interesting to notice that these filters have their highest value at zero-lag, meaning that we have a strong amplitude correction with few kinematic changes. The zero-lag values are also larger at the top of the model, as anticipated. Looking more closely at these filters, we see that the coefficients follow the structure of the Marmousi model (upper right corner).

Having estimated the filters $b$ in equation (12), I apply them to $m_1$ to obtain an improved image. To validate this approach I show in Figure 3a the result of five conjugate gradient (CG) iterations with the Marmousi data. This results show higher amplitudes at the top but with inversion artifacts. This problem should be addressed with a proper regularization scheme (Prucha et al., 1999). In Figure 3b, I show the corrected image with the approximated Hessian $B$. The amplitude behavior is very similar to Figure 3a, without the inversion artifacts. Additionally, the cost is much lower.

I show in Figure 4 the ratio of the envelope of Figure 3b and 1a. This Figure illustrates that the effects of the non-stationary filters, i.e, the Hessian, are stronger on the top of the model.

**Angle domain results**

Now, I apply the same method in the angle domain as opposed to the zero-offset image domain. The angle panels are created after migration from the offset panels directly (Stolt and Weglein,
Figure 1: (a) Migration result of the Marmousi dataset, i.e., $\mathbf{m}_1$ in equation (9). (b) Migration result of the remodeled data, i.e., $\mathbf{m}_2$ in equation (9).
Figure 2: Each cell represents a non-stationary filter with its zero-lag coefficient in the middle. A fifth of the filters are actually shown in both directions. Each filter position corresponds roughly to a similar area in the model space (Figure 1a). After close inspection of the filter coefficients, these filters seem to follow the structure of the Marmousi model. They are also stronger at the top of the model, as expected.
Figure 3: (a) Model estimated after five iterations of CG. The model is noisy because no regularization has been applied. (b) Model estimated after applying the adaptive filters to $m_1$. The amplitude behavior is similar to (a) without the artifacts and with fewer iterations.
Figure 4: Ratio of the envelopes of Figures 1a and 3b. Brighter colors correspond to higher values. The main effect of the filters is clearly visible at the top.

1985; Weglein and Stolt, 1999; Sava and Fomel, 2000; Sava et al., 2001). I show in Figures 5 and 6 the angle gather at two locations of the model (3050 and 5550 m) for the migration result ($m_1$), the filtering result ($\hat{b} \ast m_1$) and the inversion result after five CG iterations. Again, the angle gather after filtering has less artifacts than with inversion with a similar amplitude pattern throughout the section. In Figure 6, we notice that the filtering approach improves the continuity of some reflectors (between 500 and 600 m.) and that the amplitude increases for large angles. This behavior is similar to what Sava et al. (2001) observed for wave equation migration with amplitude corrections. I display in Figure 7 the estimated filters for the angle gather in Figure 5. Again, we see that we have higher amplitudes for filters at large aperture angles. In addition, we notice a smoothing effects of the filters along the angle axis.

Finally I show in Figure 8 a comparison of stacked images across angle for $m_1$ and $m_2$. The bottom reflectors are stronger than in Figure 1 with fewer migration artifacts, as expected from the stack. The differences between the two stacked-data panels in Figure 8 are similar to what we observe in Figure 1, i.e, stronger amplitudes at the top of the image for $m_1$ and fewer migration artifacts for $m_2$. In Figure 9, I display a comparison between the inversion and filtering approaches for the stacked data. Again, the amplitude pattern is the same for both images with far less artifacts in the filtering result (Figure 9b).
DISCUSSION

In this paper, I presented a method that intends to correct migrated images by approximating the Hessian of the imaging operator with non-stationary matching filters. These filters are estimated from two migration results. One migrated image, \( m_1 \), corresponds to the first migration result. The second image, \( m_2 \), is computed by re-modeling the data from \( m_1 \) and then by re-migrating it. It turns out that the relationship between \( m_1 \) and \( m_2 \) is similar to the relationship that exists between the least-squares inverse \( \hat{m} \) and \( m_1 \). In the proposed approach, this relationship is simply “captured” by matching filters.

I demonstrate with the Marmousi dataset that this approach gives a better image than does least-squares without regularization and at a lower cost. In addition this approach can be used on images migrated at zero-offset or in the angle domain. As opposed to Hu et al. (2001), the correction in the image is completely data driven, does not depend on the velocity, and can be applied with any migration operator. It also works in the poststack or prestack domain without any extra effort. Providing the data and the ability to run at least two migrations to estimate \( m_2 \), this method would be easy to apply with 3-D migrated images.

Compared to Rickett (2001), this proposed approach does not need reference images. In addition, the multi-dimensional filters offer more degrees of freedom for the correction than does a simple zero-lag weight: in that we might also correct for kinematic changes and move energy locally in the image. In the limit case where we choose one filter per point and only one coefficient (zero-lag) per filter, the matching filter approach would theoretically perform better than Rickett’s method because the weights would be optimal in a least-squares sense without ad-hoc smoothing. In the future, it would be valuable to go beyond 2-D filters by extending them to 3-D and to test this 3-D approach with more field data. In addition, these filters could be used as preconditioning operators providing faster convergence for iterative inversion. The stability of the inverse filters still needs to be addressed, however.

ACKNOWLEDGMENTS

I would like to thank Paul Sava for providing the migration codes.

REFERENCES


Figure 5: The cmp location is 3050 m. (a) Migration result. (b) Inversion result. (c) Filtering result ($\hat{\mathbf{b}} \ast \mathbf{m}_1$). The filtering result is comparable in amplitude to the inversion panel with fewer artifacts. [antoine3-comp.ang.100] [CR,M]


Figure 6: The cmp location is 5550 m. (a) Migration result. (b) Inversion result. (c) Filtering result ($\hat{b} \ast m_1$). [antoine3-comp.ang.300] [CR,M]

Figure 7: Sample of the estimated matched filters for Figure 5a. Each filter is 10 by 10. The shape of these filters prove that they tend to smooth energy along the main reflectors. [antoine3-filter-minve0] [CR]
Figure 8: (a) Stack across angle of the migrated image $m_1$. (b) Stack across angle of the migrated image $m_2$. [antoin3-comp.stack.ang1] [CR,M]
Figure 9: (a) Stack across angle of the inverted image. (b) Stack across angle of the filtered image $\hat{b} \ast m_1$. [antoine3-comp.stack.ang2] [CR,M]


Short Note

Directions in 3-D imaging - Strike, dip, both?

Marie L. Clapp

INTRODUCTION

In an ideal world, a 3-D seismic survey would have infinite extents and dense shot and receiver grids over the entire x-y plane. This would provide the best illumination possible everywhere in the subsurface. In our world, our limited source-receiver geometries allow energy to leave the survey and the density of our shot and receiver arrays depends on the equipment available. For 3-D surveys, the geometry leads to limited azimuth ranges dependent on the direction in which the survey is shot. The illumination itself depends on the subsurface structure. For all of these reasons, shooting our surveys in different directions will result in different subsurface illumination.

Some studies have been done comparing strike direction and dip direction surveys, which can be considered to be a special case of any two surveys shot in different directions over the same area. O’Connell et al. (1993) found that for many CMP-based processes, strike direction datasets had advantages over dip directions. Etgen and Regone (1998) showed that there are differences in multiple attenuation and illumination between strike and dip direction surveys.

In this paper, I will examine the results of common azimuth migration of 3-D datasets oriented along the dip and the strike directions of the Amoco 2.5-D Carpathian Mountains overthrusting the North Sea. I will explain how the differences in the images might be used to compensate for illumination problems and theorize that these problems may be overcome by using a regularized inversion of both datasets to produce images that combine the illumination of both.

AMOCO 2.5-D DATASET

Synthetic dataset

The Amoco 2.5-D dataset juxtaposes the complex folding of the Carpathian mountains with the classic salt structures of the North Sea. A 2-D slice of the velocity model can be seen

1email: marie@sep.Stanford.edu
in Figure 1. This slice can be replicated along a third dimension to create the 2.5-D velocity model. Naturally, the second dimension is the dip direction and the third dimension is the strike direction.

The data for this synthetic model were created by finite difference modeling. It was done as a 3-D survey shot in the dip direction. Since this modeling was done over a 2.5-D velocity structure, it is a simple matter to manipulate the data to generate a strike line survey as well, although the inline and crossline geometries will be different from the dip line survey.

Figure 1: Velocity model for the Carpathians over the North Sea.

Migration results

I ran common azimuth migration for both the dip and strike direction 3-D data. Since we are dealing with a 2.5-D subsurface and we know the data was acquired in the true strike and dip directions, the common azimuth assumption is valid. The result of the dip direction common azimuth migration can be seen in the top panel of Figure 2 and the strike direction common azimuth migration is in the bottom panel.

The large difference in amplitudes between the mountain side of the model and the sea side is expected because of the very different impedances that are apparent from the velocity model. For these experiments, this difference in amplitudes should not concern us greatly, as our primary consideration is the illumination, which can be examined in the areas with better impedances. The illumination problems can be seen most clearly along the flat reflector at a depth of 4 km and at the base of the salt.

Comparing the dip direction and strike direction common azimuth migration results is interesting. The strike direction result has stronger artifacts in the area from 1 to 2 km depth under the overthrust, but shows better continuity in the deeper region. Along the flat reflector
at depth=4 km, the strike direction migration shows illumination problems that are different from those seen in the dip direction results. Similar behavior can be seen along the base of the salt, which is patchy in both results but is clear in different regions. The differences in illumination that are evident from these results is what drives the question: can we use these two datasets to get one model that combines the best illumination of both datasets?

![Image](Figure 2: Zero offset ray parameter slices. Top: common azimuth migration of the dip direction data. Bottom: common azimuth migration of the strike direction data.)

**JOINT INVERSION**

It is difficult to combine the information from two surveys shot in different directions to take advantage of the different illumination patterns. If we believe that our results are accurately imaged in space, it is tempting to just add them, including some equalization term to account for amplitude differences. The result of adding the common azimuth results of the dip and strike direction data is in Figure 3. Simply adding the migration results is not wise. Migration
operators essentially sum along complex hyperboloids and the migration result is the summation of all of those hyperboloids. Ideally all of the summations will cancel out artifacts. However, in complex areas, we do not have the data necessary to cancel out all of the artifacts. The differences in artifacts and frequency content in the migration results in Figure 2 have degraded the image. The illumination problems visible on the flat reflector at depth=4 km and along the salt base are still present in the added result (Figure 3).

Figure 3: Result of simply adding the common azimuth results of the dip and strike direction data seen in Figure 2.

The problems that migration encounters in complex areas are well known. These problems can be solved to some degree by imaging through regularized least-squares inversion (Nemeth et al., 1999; Duquet and Marfurt, 1999; Ronen and Liner, 2000). This inversion process can be described by the fitting goals

\[ \mathbf{0} \approx \mathbf{Lm} - \mathbf{d} \]  
\[ \mathbf{0} \approx \epsilon \mathbf{Am} \]  

These fitting goals relate one dataset, \( \mathbf{d} \), to one model, \( \mathbf{m} \), using a linear imaging operator \( \mathbf{L} \). This first goal is the data fitting goal. The second goal is the model styling goal, wherein a regularization operator, \( \mathbf{A} \), acts on the model to help compensate for poor illumination. The regularization parameter \( \epsilon \) allows us to balance the strength of the model styling goal against the data fitting goal. We can replicate these fitting goals to obtain two models from two datasets:

\[ \mathbf{0} \approx \begin{bmatrix} \mathbf{L_1} & \mathbf{m_1} \\ \mathbf{L_2} & \mathbf{m_2} \end{bmatrix} - \begin{bmatrix} \mathbf{d_1} \\ \mathbf{d_2} \end{bmatrix} \]  
\[ \mathbf{0} \approx \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \end{bmatrix} \begin{bmatrix} \mathbf{A_1} & \mathbf{m_1} \\ \mathbf{A_2} & \mathbf{m_2} \end{bmatrix} \]
In these fitting goals, $L_1$ and $L_2$ are linear imaging operators such as the downward continuation migration operator presented by Prucha and Biondi (2002) that relates the individual models, $m_1$ and $m_2$, to the individual datasets $d_1$ and $d_2$. The regularization operators $A_1$ and $A_2$ are individually applied to the two different models and should be designed to compensate for poor illumination. However, these expanded fitting goals do not take advantage of the fact that we are imaging the same areas of the subsurface using two different datasets. We need an additional fitting goal that will allow us to regularize the models based on each other.

One potential scheme for jointly inverting two datasets shot over the same area is based on regularization between stacks of the models. This inversion can be expressed in terms of three fitting goals that combine the two datasets:

\[ 0 \approx \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} - \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \]  \hspace{1cm} (4)

\[ 0 \approx \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} \]  \hspace{1cm} (5)

\[ 0 \approx \epsilon_3 [S_1 - ES_2] \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} \]  \hspace{1cm} (6)

The third fitting goal is the key to regularizing the illumination between the two models. $S_1$ and $S_2$ are stacking operators for the two models. $E$ is a cross-equalization operator that compensates for differences in amplitudes and wavelets between the two models. $E$ can be created from the migration result of each dataset (Rickett et al., 1997; Rickett and Lumley, 1998). The $\epsilon$s are weights that are used to balance the different fitting goals.

These fitting goals will result in two models that have been regularized to help fill in areas that are illuminated differently by the two surveys. Either of these models should have better illumination than the result of migration of the individual datasets. Ideally, if both datasets have the same angular coverage, the stacks of the models should be the same, but the information along the ray parameter axes will be different. This difference is due to the fact that the ray couples for each survey are traveling through different media. The directions of the surveys will cause the rays to have different illumination problems and encounter other problems such as anisotropy, attenuation, and velocity contrasts that may cause evanescence.

**FUTURE WORK**

I plan to apply these fitting goals to the Amoco 2.5-D datasets. Proper construction of the regularization, stacking, and cross-equalization operators is essential. Preconditioning these fitting goals may not help to speed convergence, particularly because of the stacking operators, which will tend to act slowly when “unstacking”. I may choose to develop a different fitting goal to replace the stacking fitting goal. After I have satisfactorily applied my joint fitting goals to this synthetic, I hope to obtain two real 3-D surveys to further test this joint inversion scheme.
ACKNOWLEDGMENTS

I would like to thank BP-Amoco for allowing us to use this 2.5-D dataset and Dennis Yanchak for getting the data to SEP.

REFERENCES


Illumination compensation: Model space weighting vs. regularized inversion

Marie L. Clapp

ABSTRACT

In areas of complex geology, finite surveys and large velocity contrasts result in images full of artifacts and amplitude variations due to illumination problems. Cheap methods such as model space weighting and expensive methods such as regularized least-squares inversion are among the schemes that have been developed to deal with these issues. Model space weighting operators can be obtained by applying a forward modeling and an adjoint migration operator to a user-specified reference model, then applied \textit{a posteriori} to an image. Regularized least-squares inversion applied in an iterative scheme requires the selection of an imaging operator and a regularization operator that will compensate for the illumination problems during the processing itself. Applying each of these methods to the Sigsbee2A dataset, a complex synthetic, shows that model space weighting \textit{a posteriori} can help to equalize amplitudes, but will strengthen artifacts within the image. Regularized least-squares inversion will equalize amplitudes and reduce artifacts, but can be quite expensive.

INTRODUCTION

Imaging the subsurface in areas of poor illumination is an increasingly important goal. Limited survey sizes and lensing caused by large velocity contrasts result in unwanted amplitude variations and artifacts in our images. Although much recent work as been done to improve imaging in these areas with least-squares inversion techniques (Prucha et al., 2000, 2001; Kuehl and Sacchi, 2001; Prucha and Biondi, 2002a,b), these methods are computationally quite expensive. A less expensive method to equalize amplitudes is simple weighting in the model space.

Model space weighting operators can be applied to an image after it has been produced by an imaging operator. One familiar operator used to help bring up amplitudes in seismic images is automatic gain control (AGC). However, AGC does not make use of the information we have about the subsurface and, therefore, the illumination. Since most imaging operators use a velocity model, why not use that velocity model to help design a smarter weighting operator that will do a better job of compensating for illumination? Rickett (2001) suggested using the velocity model and a downward continuation operator to approximate the Hessian,
giving us an appropriate weighting operator.

In this paper, I will first review the method proposed by Rickett (2001) for the construction of a model space weighting operator. I will then apply this weighting operator to a complex synthetic model with clear illumination problems. Finally, I will compare the model space weighted results with the more expensive regularized inversion methods used by Prucha and Biondi (2002b).

METHODOLOGY

The issue of getting accurate amplitudes in areas of poor illumination is a well known problem. It is generally difficult to derive and code even an approximation to the amplitude terms that would compensate for illumination in our imaging operators (Beylkin, 1985; Albertin et al., 1999). Therefore, several schemes have been developed to try to compensate for our inaccurate amplitude terms. These can be divided into expensive schemes, like regularized least-squares inversion, and less expensive schemes, like weighting operators that can be applied \textit{a posteriori}. Rickett (2001) described a simple \textit{a posteriori} diagonal weighting operator \( W_m \).

To approximate \( W_m \), we begin by formulating our imaging operation as

\[
W_m^2 L' d \approx m
\]  

(1)

where \( L' \) is an imaging operator, \( d \) is the data, and \( m \) is the model. When an intelligent weighting operator is applied to the image produced by \( L' d \), it can be a good substitute for iterative inversion in areas with good signal-to-noise ratios (Ronen and Liner, 2000).

The question now is how to get \( W_m \). Claerbout and Nichols (1994) showed that by applying a forward operator and its adjoint to a reference model, we will get a weighting function with the correct physical units. Rickett (2001) took it a step further to show that for illumination, we can find the correct weighting function in this manner:

\[
W_m^2 = \frac{\text{diag}(m_{\text{ref}})}{\text{diag}(L'L m_{\text{ref}}) + \epsilon^2 I}.
\]  

(2)

Here \( m_{\text{ref}} \) is the chosen reference model, \( \epsilon \) is a damping parameter related to the signal-to-noise ratio, and the \( <> \) indicate that we take the smoothed analytic signal envelopes. \( \epsilon \) ensures that we will not be dividing by zero.

APPLICATION

Imaging and illumination

In order to examine this weighting operator, I began by selecting a dataset from an area with poor illumination. I chose to work with part of the Sigsbee2A dataset. This dataset has a salt
Figure 1: Upper left: part of the Sigsbee2A velocity model. Upper right: the flat layer reference model. Bottom: the weighting operator ($W_m^2$) produced using flat layers as the reference model. [CR]
body with strong shadow zones beneath its edges. The velocity model can be seen in Figure 1.

The imaging operator I chose to use is the downward continuation migration described in Prucha et al. (1999a,b). To summarize, this migration is carried out by downward continuing the wavefield in frequency space with the Double Square Root (DSR) equation, slant stacking at each depth, and extracting the image at zero time. In 2-D, this process can be described as:

\[
P(\omega, x_m, x_h; z = 0) \xrightarrow{\text{DSR}} P(\omega, x_m, x_h; z) = P(\omega, x_m, x_h; z)
\]

(3)

\[
P(\omega, x_m, x_h; z) \xrightarrow{\text{Slant stack}} P(\tau, x_m, p_h; z)
\]

(4)

\[
P(\tau, x_m, p_h; z) \xrightarrow{\text{Imaging}} P(\tau = 0, x_m, p_h; z)
\]

(5)

The resulting image has the dimensions of depth (z), common reflection point (CRP), and offset ray parameter (p_h).

The result of carrying out this migration on the synthetic dataset can be seen in Figure 2. The stacked image (top panel) shows the shadow zones that we expect beneath the salt edges. The lower panel shows 10 different ray parameters gathers taken from between CRP = 10km and CRP = 12km. There are definite variations along the flat events that can also be attributed to illumination problems.

**Weighting operator**

To construct a weighting operator for this dataset, we must choose a reference model. Claerbout and Nichols (1994) suggested that the best reference model would be the perfect image, but we never have a perfect image, especially in areas of poor illumination. Rickett (2001) suggested two alternative reference models. One reference model was composed of random noise. This left a “footprint” of the random number field on the weighting function and produced different weighting functions for different random number fields. The final suggested reference model was that of flat layers, essentially the flat layer calibration done by Black and Schleicher (1989). This model produces a weighting function that is noise-free and well-behaved. The flat layer reference model doesn’t rely on the data. It does depend on the velocity model and the survey geometry, which are the primary causes of poor illumination with which we are concerned. I choose to use the flat layer reference model for these reasons.

The flat layer model I constructed can be seen in the top right panel of Figure 1. It has the same dimensions as the migration result from the previous subsection, but is composed entirely of flat layers. I modeled the data that would be produced from these flat layers with the adjoint of the migration scheme described in the previous subsection, using the velocity model from Figure 1. I then migrated that data to get the weighting operator defined in equation (2). The lower panel of Figure 1 shows this weighting operator.

The weighting operator in Figure 1 is displayed as a flattened cube. The front of the cube shows a common ray parameter slice in which the high values of the weighting operator can be seen in the shadow zones. The top of the cube is shown above the common ray parameter slice.
and shows a depth slice. In this depth slice, we can see that the weighting operator indicates that the subsalt illumination along CRPs is becoming worse with larger ray parameters. The side of the cube shows a ray parameter gather taken from a CRP at the salt edge. This view shows how illumination varies along ray parameter and with depth. It also shows some vertical striping that I believe is caused by the way I took the smoothed analytic signal envelope. We must keep this in mind as we apply this weighting function to the image.

Figure 2: Top: stack of migration. Bottom: ray parameter gathers from migration, taken from between CRP = 10km and CRP = 12km.

Result

Now that we have the weighting operator, it can be applied to the image (Figure 2) as defined by equation (1). This result can be seen in Figure 3.

The weighted stacked image in Figure 3 shows improvement over the stacked image in Figure 2. All of the events can be followed further underneath the salt edge after weighting. Events under the salt are stronger, although they now show artificial bright spots. In the ray parameter gathers, the events are stronger and more continuous. However, the shadow zones are now full of noise. It is clear that under the salt we have primarily increased the noise - there
is almost no sign of real events. Overall, we see some improvement just under the salt edge, where the signal-to-noise ratio is still fairly good, but there are serious problems farther under the salt where the ratio is low. Also, even in the areas that show improvement, the artifacts are still strong.

**COMPARISON WITH INVERSION RESULTS**

I have shown that using the *a posteriori* weighting operator can improve amplitudes. However, it does that at the cost of increasing the noise as well. At this point, I would like to compare the result of the weighting operator with that of regularized least-squares inversion. The inversion scheme I chose to use is the geophysical preconditioned scheme explained by Prucha and Biondi (2002a). This iterative inversion uses the imaging operator described above. The regularization operator is a gradient operator along the ray parameter axis. It is designed to improve continuity and minimize amplitude variation due to illumination along the ray
I applied ten iterations of this regularized inversion to the same part of the Sigsbee2A dataset as shown above. The result can be seen in Figure 4. The stacked result in the upper panel shows that we have extended the events even further under the salt edges than the result after model space weighting (Figure 3). The events under the salt are stronger than the result of the migration (Figure 2) but do not have the artificial amplitude variations or the increase in the strength of artifacts that are seen in the result of model space weighting. The fault underneath the salt nose is better focused. The stacked inversion result also has much cleaner shadow zones than both the migration and weighted results. In the ray parameter gathers, the inversion result shows better continuity and amplitude behavior than the migration result and is much cleaner than the weighted result. Overall, the inversion result is better than the model space weighted result. This can be seen even more clearly in Figure 5, which shows an enlarged view of the stacks in the area at the salt edge for the migration (left panel), the model space weighted result (center), and the regularized inversion result (right panel). The inversion result has done a better job of extending the reflectors beneath the salt and has fewer
artifacts than either of the other results. However, the inversion result has a computational cost approximately seven times $^2$ greater than the weighted result.

![Figure 5: Results enlarged from CRP = 9.75km to CRP = 11.25km. Left: stack of migration. Center: stack of model space weighted result. Right: stack of result after 10 iterations of preconditioned inversion.](image)

**CONCLUSIONS**

Although the model space weighting operator does help improve the amplitudes in poorly illuminated areas of seismic images, it does not do as good a job as the least-squares inversion. This is largely due to the fact that the weighting operator increases the amplitude of the noise as well as that of the signal. Regularized least-squares inversion does a better job compensating for poor illumination in areas with low signal-to-noise ratios. Inversion also helps to reduce artifacts. However, the weighting operator is substantially cheaper than the inversion, so the decision of which to use in areas of poor illumination becomes dependent on signal-to-noise ratios and available computing power.

$^2$Note that the cost of generating the model space weighting operator is equal to two migrations and is applied to a migrated image, making the total cost equal to three migrations. The regularized inversion has a cost of two migrations for each iteration, making the total cost equal to twenty migrations.
ACKNOWLEDGMENTS

I would like to thank SMAART JV for providing us with the Sigsbee2A dataset.

REFERENCES


Iteratively re-weighted least-squares and PEF-based interpolation

William Curry

ABSTRACT
Interpolation methods frequently deal poorly with noise. Least-squares based interpolation methods can deal well with noise, as long as it is Gaussian and zero-mean. When this is not the case, other methods are needed. I use an iteratively-reweighted least-squares scheme to interpolate both regular and sparse data with non-stationary prediction-error filters. I show that multi-scale methods are less susceptible to erratic noise than single-scale PEF estimation methods. I also show how IRLS improves results for PEF estimation in both cases, and how IRLS can also improve the second stage of the interpolation, where the unknown data is constrained by the PEF.

INTRODUCTION
Data interpolation is a long standing problem in exploration geophysics, especially in the case of 3-D land data. Many different interpolation methods exist, such as kinematic methods (Chemingui, 1999; Fomel, 2001; Vlad and Biondi, 2001), where the missing data is predicted with operators such as NMO or AMO, Fourier-based methods (Schonewille, 2000) that can now interpolate non-uniformly-sampled data, and prediction-error filter based methods, both in the $f-x$ (Spitz, 1991) and $t-x$ (Claerbout, 1992, 1999) domains.

One of these current methods of seismic data interpolation is a two-stage linear least-squares scheme (Claerbout, 1999), where a prediction-error filter (PEF) is estimated on some known data, and then the newly-estimated PEF is used to regularize the missing data. This method can use non-stationary PEFs, and can also work on coarsely-sampled data (Crawley, 2000). Most recently, methods for PEF estimation on sparse data have been developed for both stationary (Curry and Brown, 2001) and non-stationary (Curry, 2002) PEFs.

Since these methods are based on least-squares inversion, certain assumptions are made about the statistics of the noise, i.e. that the noise has a Gaussian distribution and zero mean. As such, least-squares methods are less successful when dealing with erratic, bursty noise. Erratic data can be dealt with by the use of an $\ell^1$-norm (Claerbout and Muir, 1973), or by removing the data by preprocessing (Abma, 1995; Claerbout, 1999). Existing $\ell^2$-norm-based methods can be adapted to use a variable norm by the use of iteratively re-weighted least-squares (IRLS) algorithms (Fomel and Claerbout, 1995; Guitton, 2000).

1email: bill@sep.stanford.edu
I improve the interpolation of bursty data by using an IRLS-based approach to estimate non-stationary PEFs on both regularly-sampled and sparse data. I then use this PEF to interpolate the data, again using an IRLS-based approach to fill in the missing data. I show how both changes can improve the interpolated result in the presence of erratic noise. I will first show this on a synthetic case with a simple noise model, and then on a 2D land data set.

**BACKGROUND**

A PEF is estimated by minimizing the output of the known data convolved with the PEF, where the first coefficient of the PEF is constrained to be 1 and the other PEF coefficients are unknown. This can be written as

$$W(DKf + d) \approx 0,$$

(1)

where $K$ is a mask that constrains the first filter coefficient to 1, $W$ is a diagonal weighting operator that is equal to 1 only when all filter coefficients lie on known data (and is 0 otherwise), $D$ represents convolution with the data, $d$ is simply a copy of the data, and the unknown PEF is denoted as $f$.

Fitting goal (1) works well for estimating the PEF if there are contiguous data. If the data are sparse enough so that there are an inadequate number of fitting equations, such as case (a) in Figure 1, a multi-scale method is used (Curry, 2002). With this method, more fitting equations are generated by rescaling the data ($d$) to multiple different grid sizes ($d_i$), and then convolving the filter with all of the different scales of data. An example of rescaled data is case (d) of Figure 1. This is done by first performing linear interpolation on the original data in (a), followed by adjoint linear interpolation onto a coarser grid in (c). That is: for each bin, create a data point located at the center of that bin, and then use those data points for linear interpolation onto a coarser grid. All of these different scales of data ($d_i$) can then be convolved ($D_i$) with the single unknown PEF ($f$), which leads to a better-determined system of
equations,

\[
W \begin{bmatrix}
D_0 \\
D_1 \\
D_2 \\
\vdots \\
D_n \\
\end{bmatrix}
\begin{bmatrix}
I \\
P_1 \\
P_2 \\
\vdots \\
P_n \\
\end{bmatrix}
Kf + \begin{bmatrix}
d_0 \\
d_1 \\
d_2 \\
\vdots \\
d_n \\
\end{bmatrix} \approx 0.
\]

The weight \(W\) now includes a weight for missing data in all of the scaled versions of the data.

In the case of a non-stationary prediction error filter, the PEF in fitting goal (1) changes from a vector \(f(ia)\) to a much longer vector \(f(ia, id)\), but the fitting goal looks the same. A full description of the changes in \(K\) and \(D\) in fitting goal (1) are explained elsewhere (Guitton, 2003). To extend fitting goal (2) for non-stationary PEFs, more changes need to be made, since the different scales of data (\(d_i\) and \(D_i\)) have different sizes, and we have a PEF that varies with position. We now need to subsample the PEF to match the different data sizes, so that

\[
W \begin{bmatrix}
D_0 \\
D_1 \\
D_2 \\
\vdots \\
D_n \\
\end{bmatrix}
\begin{bmatrix}
I \\
P_1 \\
P_2 \\
\vdots \\
P_n \\
\end{bmatrix}
Kf + \begin{bmatrix}
d_0 \\
d_1 \\
d_2 \\
\vdots \\
d_n \\
\end{bmatrix} \approx 0.
\]

A sub-sampler \(P_i\) has been introduced that reduces the size of the non-stationary PEF from \(na \times nd\) to \(na \times nd_i\). Since this non-stationary filter is now likely under-determined due to a large increase in the number of unknown filter coefficients, a regularization fitting goal,

\[
\epsilon A f \approx 0,
\]

must also be added, where \(A\) is a regularization operator that roughens common filter coefficients spatially. This improves the stability of the PEF, as well as insures that it will vary smoothly in space.

Once the PEF has been determined, a second minimization problem is solved, where the output of a convolution of the newly found stationary or non-stationary PEF (\(F\) representing convolution with \(f\)) with the partially unknown model \(m\) is minimized. The known points of the model are constrained to their actual values \(m_k\), via a missing data mask, \(K_{data}\). The output of this final step is the interpolated model, \(m\), where

\[
K_{data} m = m_k
\]

\[
F m \approx 0.
\]

All of the above fitting goals are minimized in a least-squares sense, by using a conjugate-gradient solver. Iteratively re-weighted least-squares (IRLS) is a method where a weighting function that varies with iteration is applied to a conjugate-gradient solver, so that different norms can be approximated. I will not go into the details of IRLS here, as they are covered in-depth elsewhere (Scales and Gersztenkorn, 1988; Darche, 1989; Fomel and Claerbout, 1995; Claerbout, 1999; Guitton, 2000), except to mention that several parameters need to be set, including the quantile of the residual used, and the frequency of recalculation of the weights.
IRLS-BASED INTERPOLATION

The use of IRLS has been successful where erratic data is present, and the operators used to model such data are relatively error free. However, in the first stage of this interpolation process, the data \( \mathbf{d} \) is also present in the operator as \( \mathbf{D} \), convolution with the data. Where a single bad data point would only destroy one fitting equation in other problems, in this case it would destroy a number of fitting equations proportional to the size of the filter multiplied by the number of scales.

Since the convolution occurs with multiple scales of data, the noise could be partially attenuated by the scaling process. The multi-scale approach does provide more fitting equations that are less compromised by the noise than with a single-scaled approach, but with the current linear-interpolation-based method this is only partially successful, as the noise is only being attenuated by simple averaging.

The second stage of the interpolation is also not ideally suited to IRLS, but is more likely to benefit from the approach. The errors in the data are constrained to their previous (incorrect) values, but are not directly convolved with the interpolated model. So, instead of destroying the filter, these errors will be more subtle, and centered around the bad data points.

SYNTHETIC EXAMPLE

Figure 2 shows a two-dimensional slice from the qdome model (Claerbout, 1993) with 1 percent of spikey noise (distributed as a delta function) added at roughly 10 times the maximum value of the signal. This data set is ideal for this type of test, since it has varying dips, non-stationarity, aliasing, and bursty noise. Two cases of sub-sampling are shown in Figure 2, one where every second trace in the data was removed, and one where forty percent of the data was removed in a random manner.

![Figure 2: 2D test case. Left: the fully sampled version. Center: every second trace removed. Right: randomly sub-sampled by 60 percent.](bill1-qslice)

The regularly-sampled data in the center of Figure 2, was interpolated using both a non-
stationary PEF estimated from a single scale of data and using a non-stationary PEF estimated from multiple scales of data, shown in Figures 3 and 4, respectively.

The data was first interpolated with the standard least-squares method, first by estimating the non-stationary PEF with fitting goals (3) and (4), then by filling the data using the PEF in fitting goals (5) and (6). The results are in the top-left panels of Figures 3, 4 and 5 for the regularly-sampled single-scale case, multi-scale case, and sparse multi-scale case, respectively.

The errors in the filter around the data spikes are very obvious, and cause very noticeable errors in the interpolation. Dips are incorrect, and the PEF blows up in areas. The single-scale PEF appears to be more susceptible to spikes in the data than the multi-scale PEF.

The IRLS filter estimation results are shown in the center-left panels of Figures 3, 4 and 5. After much trial and error, recalculating the weights every 30 iterations appear to give the most pleasing result, with the same parameters used for all cases. The results for the single and multi-scale cases are very similar, so the IRLS filter estimation has improved the single-scale estimation more than the multi-scale estimation. Overall, the dips captured by the PEF appear to be more correct, and the PEF blows up less.

In the bottom-left panels, an IRLS-based approach was used both on the estimation of the PEF as well as the interpolation of the data. The differences can be more subtle in this case, since this change does not affect things that are quite so obvious, such as the dips of the interpolated result or the stability of the PEF. Instead, the changes here are largely in the amplitude of the interpolated result, especially around the spikes. Differences between these three results are shown on the right side of Figures 3, 4, and 5, where the differences between IRLS PEF estimation and \( \ell^2 \) interpolation and IRLS PEF estimation and interpolation are more clear.

For a proof-of-concept, this example shows limited success. The problems caused by the spikes in the data are definitely reduced but not solved. This is largely due to the noise being present in the operator in the first stage, and the noise being constrained in the second stage. The multi-scale estimation method appears to be less susceptible to the noise, but are still improved by the use of IRLS. Next, a real data example will be examined.

### CASC93 DATA SET

The previous example used noise that presents specific problems with existing \( \ell^2 \) based optimization methods. For a more realistic test of this method, a real data set with severe noise problems is needed. I have now interpolated a single event from the CASC93 (Nabelek and Zandt, 1993) data set. The CASC93 data set is passive seismic data experiment performed in 1993-1994 in central Oregon, with an irregular 2-D array of three-component seismometers. Arrivals from distant earthquakes were recorded, and can be used to image the lower crust of that region (Shragge, 2003). As shown in Figure 6, the data is very noisy, containing broadband noise, as well as artifacts from the preprocessing of the data. The data is on a 2-km grid, with approximately 30 percent of the bins filled.
Figure 3: 2D regularly-sampled case, interpolated with a single-scale PEF. Left side, top: interpolated without IRLS. Middle: the PEF was estimated with IRLS. Bottom: both the PEF and the interpolated model were estimated with IRLS. Right side, top: the difference between a PEF estimated with IRLS and no IRLS used. Middle: the difference between both the PEF and the model estimated with IRLS, and no IRLS. Bottom: the difference between both the PEF and model estimated with IRLS, and just the PEF estimated with IRLS.
Figure 4: 2D regularly-sampled case, interpolated with a multi-scale PEF. Left side, top: interpolated without IRLS. Middle: the PEF was estimated with IRLS. Bottom: both the PEF and the interpolated model were estimated with IRLS. Right side, top: the difference between a PEF estimated with IRLS and no IRLS used. Middle: the difference between both the PEF and the model estimated with IRLS, and no IRLS. Bottom: the difference between both the PEF and model estimated with IRLS, and just the PEF estimated with IRLS.
Figure 5: 2D irregularly-sampled case. Left side, top: interpolated without IRLS. Middle: the PEF was estimated with IRLS. Bottom: both the PEF and the interpolated model were estimated with IRLS. Right side, top: the difference between a PEF estimated with IRLS and no IRLS used. Middle: the difference between both the PEF and the model estimated with IRLS, and no IRLS. Bottom: the difference between both the PEF and model estimated with IRLS, and just the PEF estimated with IRLS.
While the results are definitely more subtle than in the test case, the same conclusions are supported. In the left side of Figure 6, we can see that switching to an IRLS PEF estimation improves the estimation in some areas, most notably in the lower-right corner patch, where the PEF does not benefit as much from regularization due to end effects. Smaller spikes in the difference between only IRLS PEF estimation and IRLS PEF estimation and interpolation are shown in the lower right. As before, it appears that certain spikes appear, although they aren’t as obvious as in the previous example.

CONCLUSIONS

Using the IRLS method showed definite improvement in the interpolation of both test cases presented in this paper. However, there are problems with using the IRLS method. One of them is that it adds to an already large parameter space, requiring a significant amount of trial and error to set the parameters, especially when using IRLS on the second stage of the interpolation. Next, this method does not deal with the underlying problems that are caused by noise in the data, such as errors in the operator in the first step, and the fixing of the bad values in the second step. More sophisticated approaches, such as total least squares (Brown, 2002) that can account for errors in the operator, could be used to further reduce these issues.

We have also seen that the multi-scale PEF estimation appears to be much less susceptible to the noise than a single-scale estimation. This is attributed to some noise attenuation that happens during the rescaling of the data. This benefit of the multi-scale method could be further enhanced by using a more sophisticated rescaling operator than the current linear-interpolation based method, so that the rescaled copies of the data ($d_i$) would provide more reliable information.

Another approach of dealing with bursty noise could be in pre-processing, where the noise is identified and removed prior to the interpolation or the rescaling. This would then avoid the need for overly robust optimization methods, or more sophisticated rescaling methods. Problems arise with this avenue of research, since most signal-noise separation methods require regularly-sampled data.

REFERENCES


Figure 6: The CASC93 land data set. Top right: original data; top left: L-2 interpolated data; mid left: IRLS PEF estimation; bottom left: IRLS PEF estimation and IRLS interpolation; mid right: difference between L-2 interpolation and IRLS PEF estimation; bottom right: difference between IRLS PEF estimation and IRLS PEF estimation and IRLS interpolation.


Curry, W., 2002, Non-stationary, multi-scale prediction-error filters and irregularly sampled data: SEP–111, 327–337.


Schonewille, M., 2000, Fourier reconstruction of irregularly sampled seismic data: Ph.d. thesis: Delft University of Technology.


Short Note

Parameter optimization for multiscale PEF estimation

William Curry

INTRODUCTION

Data interpolation is a long standing and persistent problem in exploration geophysics. Methods range from those based on the known behavior of the kinematics of seismic data (Chemingui, 1999; Fomel, 2001; Vlad and Biondi, 2001), to those that are based on transformations of the data into another domain, such as the Fourier (Schonewille, 2000) or Radon (Trad et al., 2002) domains. Other methods use prediction-error filters (PEFs) in either the $f-x$ (Spitz, 1991) or $t-x$ (Claerbout, 1992, 1999) domains. Claerbout’s method is cast as inverse problem, where the known data remains constant, and the empty bins are regularized by a PEF to constrain the null space. This is done in two stages, where a PEF is first estimated on known data, and is then used to constrain the unknown data by minimizing the output of convolution of the model with the PEF.

When the data are not stationary, a non-stationary filter may be used to fill the unknown data (Crawley, 2000), and in the case of coarsely sampled data, the filter can be stretched over various scales to fit the data. A PEF may be estimated on irregularly sampled data by scaling the data to various grid sizes and simultaneously estimating a single PEF on the various scales of data (Curry and Brown, 2001). This method of PEF estimation has also been shown to work for non-stationary filters (Curry, 2002).

One of the main drawbacks associated with using PEFs is the large number of non-intuitive parameters needed to create them. These include parameters relating to the structure of the PEF, it’s size, shape, and the degree of non-stationarity in the filter. There are also many parameters needed in the PEF estimation process: the number of conjugate-direction iterations used to estimate the filter; and in the case of non-stationary PEFs, the type and amount of regularization used in the estimation.

In addition, the multi-scale method (Curry and Brown, 2001) requires another set of parameters to determine the scales of data to be used. In the case of non-stationary PEF estimation, the possible scales of data are limited by the size of the micropatches.

Currently, the only way to estimate many of these parameters is by trial and error. The

1email: bill@sep.stanford.edu
spatial distribution of the data is known, but is not incorporated in a meaningful way into
the estimation of these parameters. Also, when dealing with a non-stationary scenario, the
importance of having all regions well-constrained by fitting equations is also an unresolved
issue.

In this paper, I explore a general strategy to estimate certain parameters, where the number
of valid fitting equations is counted at each position for every possible combination of param-
eters. A weighted sum of these equation maps may then be calculated, and from those results,
optimal parameters can be obtained.

BACKGROUND

A PEF can be estimated by minimizing (in a least-squares sense) the output of the known
data \(d\) convolved \(D\) with a filter \(f\) that is unknown except for the first coefficient, that is
constrained to 1 by \(K\). This is expressed below, where \(W\) is a diagonal weighting operator that
is equal to 1 when all filter coefficients lie on known data, and is 0 otherwise. This is written
as:
\[
W(DKf + d) \approx 0
\]

When the data are sparsely sampled, \(W\) will be zero everywhere, since there are not enough
contiguous data to estimate a PEF. An example of this is shown elsewhere in this report (Curry,
2003). More fitting equations can be added to this regression by convolving a single filter on
multiple copies of the data that have been rescaled to different grid sizes (Curry and Brown,
2001). This can be written as
\[
W \left( \begin{bmatrix} D_0 \\
D_1 \\
D_2 \\
... \\
D_n \end{bmatrix} Kf + \begin{bmatrix} d_0 \\
d_1 \\
d_2 \\
... \\
d_n \end{bmatrix} \right) \approx 0.
\]

where \(W\) is now a much larger diagonal weight for all of the copies of data \(d_i\), \(D_i\) represents
convolution with \(d_i\), and \(K\) and \(f\) are the same as in fitting goal (1).

Often the covariance of the data is not stationary, so it cannot be adequately described by a
single filter. Non-stationary PEFs may be used to overcome this limitation. These filters vary
with position, so that a filter that looked like \(f(ia)\) now looks like \(f(ia, id)\). These filters
can be estimated in a fitting goal that looks identical to fitting goal (1), except that \(W, K, D,\n\) and \(f\) are all now non-stationary counterparts to those in fitting goal (1). The details of these
changes are documented elsewhere (Guitton, 2003).

Since this non-stationary filter is now likely under-determined due to a large increase in
the number of filter coefficients, a regularization fitting goal,
\[
\epsilon A f \approx 0,
\]

must also be added, where \(A\) is a regularization operator that roughens common filter coeffi-
cients in space. This improves the stability of the PEF, and insures that it will vary smoothly
in space. In order to reduce the number of filter coefficients from $na \times nd$ to something more manageable, the filter coefficients are taken to be constant over a small spatial area, so that the number of coefficients reduces to $na \times nd / np$, where $np$ is the size of this small area, known as a micropatch (Crawley, 2000). Non-stationary PEF estimation can also be performed on sparse data (Curry, 2002), in a multi-scale fashion similar to that used in fitting goal (2). The only changes to fitting goals (1) and (3) are that fitting goal (1) changes to

$$ W \left( \begin{array}{c|c} D_0 & I \\ \hline D_1 & P_1 \\ D_2 & P_2 \\ \vdots & \vdots \\ D_n & P_n \end{array} \right) Kf + \left( \begin{array}{c} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_n \end{array} \right) \approx 0, \tag{4} $$

where $W$, $K$, $D$, and $f$ are all now nonstationary versions of those in fitting goal (2). The only new term in this equation is $P_i$, that sub-samples the non-stationary PEF from $na \times nd$ to $na \times nd_i$.

### SIZE OF THE PARAMETER SPACE

In the multi-scale method of PEF estimation there are a multitude of parameters that must be selected. They can be put into two general categories: those that determine the structure of the PEF, and those that determine how the PEF is estimated.

The first group of parameters includes the size of the PEF; the shape of the PEF; the size of the gap; and in the case of non-stationary PEFs, the size and shape of the micropatches. The size of the PEF presents a trade-off between an improved ability to capture the inverse data covariance, and the number of fitting equations available to estimate the PEF. As the size of the PEF increases, the inverse data covariance is more accurately captured, but the number of fitting equations decreases.

The gap and center parameters both involve the shape of the PEF, and both have a nonintuitive effect on the result. They have a very light impact on the number of fitting equations, and a more pronounced effect on the ability of the PEF to capture the inverse covariance of the data.

The size and shape of the micropatches determine the non-stationarity of the filter. There is another trade-off here, where smaller micropatches allow for greater non-stationarity in the PEF, but also increase the size of the model space, i.e. the number of PEF coefficients.

This next set of parameters is used in the PEF estimation process. In the case of a stationary PEF on regularly-sampled data, no extra parameters are required for the regression except for the number of conjugate-gradient iterations. However, once the data are sparse enough that a PEF cannot be estimated in the traditional manner, rescaling of the data can provide more fitting equations. The choice of these scales is present in the $D_i$ and $d_i$ in fitting goals (2) and (4).
When estimating a non-stationary PEF, quite often the problem will be under-determined; that is, there will be more unknowns in model space than fitting equations. This is overcome by adding a regularization term to the optimization which introduces more parameters, namely the choice of regularization ($A$ in fitting goal 3) and $\epsilon$. The choice of regularization is data-dependent, but it normally operates over common filter coefficients to insure a smoothly varying filter. In the case of common midpoint gathers, the regularization is a radial roughener, since in a CMP gather dips are approximately constant over radial lines (Crawley, 2000). The choice of $\epsilon$ can be handled by either balancing the model residual with the data residual, or by balancing the size of the model and data gradients within the conjugate-direction solver (Claerbout, 1999).

Most of the parameters in this problem are independent. The two exceptions to this are micropatch size and scale choice. Since non-stationary filters are linked to data, if that data is regridded, the non-stationary filter must also be regridded. So, the choice of scales must maintain the aspect ratio of the data and also maintain the spatial location of the micropatch boundaries, such that the same filter coefficients are in the same place regardless of the scale. This means that the choice of scale must cleanly divide all of the data axes as well as all of the micropatch axes. Figure 1 has an illustration of both a valid and invalid scales, given existing data and micropatch sizes.

Figure 1: Plot of data space (grid) with the non-stationary PEF micropatches (overlay). Above: the rescaling doesn’t cleanly re-sample the PEF, so the filter coefficients could be in different places depending on the scale. Below: The appropriate choice of scale maintains both the aspect ratio of the data and of the PEF. [bill2-nsscale] [NR]

There is also a trade-off when optimizing micropatch size and scale choice. When the size of the micropatches is increased, the size of the model space decreases, and the number of possible scales increases. However, the PEF is then able to capture less non-stationarity in the data.

When choosing the micropatch and scale parameters, the goal is to have as many fitting equations as possible. In the non-stationary case, the need to have these fitting equations spatially distributed over a wide area is also important so that there are as few unconstrained micropatches (those with no fitting equations) as possible. In addition to varying scale and micropatch size, we can also vary the origin of the micropatch grid by shifting the positions of the micropatches so that they are more evenly constrained.
METHOD

Since we have three interrelated parameters (scale, micropatch size, and micropatch origin), and a way of measuring the quality of the parameter choices without solving for a model, we can perform a global search in order to find the optimal choices for these parameters. Since these parameters are interdependent, the preferred method is to cycle through possible values for a single parameter, and then for each of the possible values generate all possible combinations of the other two parameters given the first. The most intuitive starting point appears to be micropatch size.

For each possible combination of micropatch size, micropatch grid origin, and scale, we count the number of fitting equations in each micropatch. A valid fitting equation is where all points of the filter fall on known data. Once this is done, we will have a $3N + 1$ dimensional hypercube, where we have $N$ dimensions of freedom in choosing micropatch size and micropatch grid origin. Since the scaling of the data must be isotropic in order to preserve the aspect ratio of the PEF, varying scale only adds one dimension of freedom. Finally, since we are measuring the spatial variance of fitting equations, another $N$ dimensions are added to the hypercube.

Naturally, we don’t want to deal with a $3N + 1$ dimensional space, so we can integrate over certain dimensions, and use the helical coordinate when dealing with others. We can choose the scales with the greatest number of fitting equations (since the cost of the inversion rises by $O(N)$ with the number of scales used) and integrate the total. We can then integrate spatially over micropatches with a weighting function that rewards a wide spatial distribution of fitting equations. Finally, we wrap micropatch size and micropatch grid origin around a helix, so the dimensionality of the problem reduces from $3N + 1$ to 2. However, the derivation of a weighting function for the spatial integration is still an open problem.

RESULTS

In this example, I use a common-offset gather from a sub-sampled version of the Shorncliff 3D land data set from southeastern Alberta (Chemmingui, 1999). I take a single micropatch size and origin, and calculate the number of fitting equations for each scale as a function of micropatch number. Since the time axis is regularly sampled, the number of fitting equations does not change with time (excluding edge effects), and can be ignored. The resulting three-dimensional cube is shown in Figure 3.

Most of the results in Figures 2 and 3 are as expected. At the original scale, no fitting equations were possible. The number of fitting equations increase as we coarsen the mesh, but then as the mesh coarsens further, the edge effects start to dominate, until we reach the bottom slice where the micropatch size corresponds to one scaled bin, leaving each micropatch with a maximum of one fitting equation. Note that the number of micropatches and hence the size of the model space stays constant as we vary the scale.
CONCLUSIONS

This preliminary work toward reducing the size of the parameter space in PEF estimation problems appears to be promising. A flexible framework has been introduced that can incorporate the spatial distribution of the data into the parameter optimization, and can easily be adapted to include weighting for data quality and other issues.

One current outstanding issue is how to up-weight fitting equations with a wide spatial distribution. Also, tests on larger datasets need to be performed in order to judge the effectiveness of this method.

REFERENCES


Figure 3: The number of fitting equations normalized by the number of filter coefficients, shown as a function of position (x and y) and scale (z). By creating many of these cubes for different parameter combinations, criteria can now be established to estimate optimal parameters.

Curry, W., 2002, Non-stationary, multi-scale prediction-error filters and irregularly sampled data: SEP–111, 327–337.


Schonewille, M., 2000, Fourier reconstruction of irregularly sampled seismic data: Ph.d. thesis: Delft University of Technology.


Interpolation of bathymetry data from the Sea of Galilee: A noise attenuation problem

Antoine Guittion and Jon Claerbout

ABSTRACT

We process a bathymetry survey from the Sea of Galilee. This dataset is contaminated with non-Gaussian noise in the form of glitches and spikes inside the lake and at the track ends. Drift on the depth measurements leads to vessel tracks in the preliminary depth images. We derive an inversion scheme that produces a much reduced noise map of the Sea of Galilee. This inversion scheme includes preconditioning and Iteratively Reweighted Least-Squares with the proper weighting function to get rid of the non-Gaussian noise. We remove the ship tracks by adding a modeling operator inside the inversion that accounts for the drift in the data. We then approximate the model covariance matrix with a prediction-error filter to enhance details in the middle of the lake. Unfortunately, the prediction-error filter has the property of degrading the resolution of the depth map at the edges of the lake. Our images of the Sea of Galilee show ancient shorelines and rifting features inside the lake.

INTRODUCTION

In this problem, we are given depth sounding data from the Sea of Galilee. The Sea of Galilee is unique because it is a fresh-water lake below sea-level. It seems to be connected to the great rift (pull-apart) valley crossing east Africa. The ultimate goal is not only a good map of the depth to bottom, but images useful for the purpose of identifying archaeological, geological, or geophysical details of the sea bottom. In particular, we should be able to identify some ancient shorelines around the lake and meaningful geological features inside the lake.

The raw data (Figure 1), irregularly distributed across the surface, is 132,044 triples, \((x_i, y_i, z_i)\), where \(x_i\) ranges over about 12 km and where \(y_i\) ranges over about 20 km. We want to interpolate the data to a regular grid using inversion. The pertinence of this dataset to our daily geophysical problems is three fold. First, we often have to do interpolation of seismic maps (Britze, 1998), potential field data (Guspi and Introcaso, 2000) or other measurements to compensate for the sparseness of acquisition geometries. Second, as seen in the raw data in Figure 1, some noise bursts need to be accounted for in the inversion scheme. This problem is common, for example, in tomography (Bube and Langan, 1997), deconvolution of noisy data (Chapman and Barrodale, 1983) and velocity analysis (Guittion and Symes, 1999) where

\(^{1}\text{email: antoine@sep.stanford.edu, jon@sep.stanford.edu}\)
outliers can degrade the final model if we assume a Gaussian distribution of the noise. And third, the final image of the Sea of Galilee will display the vessel tracks because the measurements on the lake were made on different days, with different weather and human conditions. We can directly link this problem to the goal of removing the acquisition footprint with 3-D seismic data (Duijndam et al., 2000; Schuster and Liu, 2001; Chemingui and Biondi, 2002). Therefore, the interpolation of the data from the Sea of Galilee becomes a spiky noise and a coherent noise attenuation problem. We solve both problems with a finely tuned inversion scheme that should be usable for other geophysical applications.

![Figure 1: Depth of the Sea of Galilee along the vessel tracks. On one traverse across the lake, the depth record is “u” shaped. A few “v” shaped tracks result from vessel turn-arounds.](antoine4-galileedataraw)

There is a long list of students at the Stanford Exploration Project who attempted to produce a satisfying map of the sea bottom. Fomel and Claerbout (1995) introduced the $\ell^1$ norm via Iteratively Reweighted Least Squares (IRLS) to get rid of the noise bursts present in the data. Recently, Brown (2001) attempted to remove acquisition tracks by estimating the systematic error between tracks at crossing points. Karpushin and Brown (2001) used a bank of prediction-error filters (PEFs) to whiten the residual along tracks. However, in most of these results, there is a loss of resolution that hampers our goal of seeing small features in the final image. In this paper, we borrow ideas from these authors with three new approaches. The first important twist is preconditioning (Fomel, 2001), the second one is modeling of the ship’s track instead of filtering it from the residual and the third is estimating the model covariance with a PEF. We show that preconditioning with an IRLS method removes the glitches and noise bursts very well. In addition, the modeling of the ship tracks within the inversion removes almost entirely the acquisition footprint. The PEF unravels small details in the middle of the Sea of Galilee.

We first examine the preconditioning trick along with the IRLS method. Next we introduce
a new fitting equation that takes into account the inconsistency between different tracks inside the inversion. Finally we demonstrate that the model covariance can be estimated with a PEF to better preserve small features at the bottom of the sea.

ATTENUATION OF THE NOISE BURSTS AND GLITCHES

Now, we show our formulation of the regridding problem. Let $\mathbf{h}$ be an abstract vector containing as components the water depth over a 2-D spatial mesh and $\mathbf{d}$ be an abstract vector whose successive components represent depth along the vessel tracks. One way to grid irregular data is to minimize the length of the residual vector $r_d(\mathbf{h})$

$$0 \approx r_d = \mathbf{Bh} - \mathbf{d} \quad (1)$$

where $\mathbf{B}$ is a 2-D linear interpolation operator and $r_d$ is the data residual. This fitting goal simply requires that the gridded data $\mathbf{h}$ take on appropriate values where the data $\mathbf{d}$ was collected. The bin size is 60 by 50 meters. We display a simple binning (without interpolation or inversion) of the raw data (Figure 1) in Figure 2. A coarser mesh would avoid the empty bins but lose resolution. As we refine the mesh for more detail, the number of empty bins grows as does the care needed in devising a technique for filling them. The black lines in Figure 2 are the ship tracks. Notice that some data points are outside the contour of the water. These must represent navigation errors. Figure 3 displays the ship tracks only. The straight lines in the north part of the lake are due to positioning errors. The tracks match almost exactly with the black lines in Figure 2.

Unless data is collected everywhere, and depending on how we parameterize the grid, the regridding will leave holes on the mesh. We can get rid of the holes by adding some regularization, like

$$0 \approx r_d = \mathbf{Bh} - \mathbf{d}$$

$$0 \approx \epsilon r_h = \epsilon \nabla \mathbf{h} \quad (2)$$

where $\nabla = \left( \frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$ and $r_h$ is the model space residual. We then minimize the misfit function

$$f(\mathbf{h}) = \|r_d\|^2 + \epsilon^2 \|r_h\|^2 \quad (3)$$

to estimate the interpolated map of the lake. In theory (Tarantola, 1987), the regularization operator (squared) should be the model covariance operator given an a-priori model $\mathbf{h}_0$. Since we do not have any a-priori model, we simply choose the gradient operator $\nabla$ as a way of saying that the bottom of the lake is smooth. However, as pointed out by Harlan (1995), the regularization and the data fitting goal in equation (2) contradict each other. One equation tends to add details in the final map whereas the second one (the regularization) tends to smooth it. We can more easily balance these two goals by preconditioning the problem (Fomel, 2001).
Figure 2: Simple binning of the raw data in Figure 1. The ship tracks and empty bins are visible and need to be accounted for in the inversion process.

Figure 3: Ship tracks for the Sea of Galilee dataset. The north part of the lake (top) has many navigation glitches which show up as long straight lines. Notice that very is no track going all the way from the north (top) to the south (bottom). Most of the track stop in the middle of the lake.
Preconditioning for accelerated convergence

A generally available preconditioning method is to change variables so that the regularization operator becomes an identity matrix (Claerbout and Fomel, 2002). The gradient \( \nabla \) in equation (2) has no inverse, but its spectrum \(-\nabla'\nabla\), which appears in equation (3), can be factored \((-\nabla'\nabla = HH')\) into triangular parts \(H\) and \(H'\) where \(H\) is known as the Helix derivative. This \(H\) is invertible by deconvolution (Claerbout, 1998). The fitting goals in equation (2) can be then rewritten

\[
0 \approx r_d = BH^{-1}p - d \\
0 \approx \epsilon r_p = \epsilon p,
\]

(4)

with \(p = Hh \approx \nabla h\) and \(r_p\) is the residual for the new variable \(p\). We then minimize the misfit function

\[
f(p) = \|r_d\|^2 + \epsilon^2\|r_p\|^2
\]

(5)

and finally compute \(h = H^{-1}p\) to estimate the interpolated map of the lake. Experience shows that iterative solution for \(p\) converges much more rapidly than iterative solution for \(h\) thus showing that \(H\) is a good choice for preconditioning. There is no simple way of knowing beforehand what is the best value of \(\epsilon\). Practitioners like to see solutions for various values of \(\epsilon\). Of course, that can cost a lot of computational effort. Practical exploratory data analysis is more pragmatic. Without a simple clear theoretical basis, analysts generally begin from \(p = 0\) and then abandon the fitting goal \(0 \approx \epsilon r_p = \epsilon p\) (Crawley, 2000; Rickett et al., 2001). Implicitly, they take \(\epsilon = 0\). Then they examine the solution as a function of iteration, imagining that the solution at larger iterations corresponds to smaller \(\epsilon\), and that the solution at smaller iterations corresponds to larger \(\epsilon\). In all our computations, we follow this approach and omit the regularization in the estimation of the depth maps.

\(\ell^1\) norm

We show how spikes and noise glitches can be attenuated with an approximate \(\ell^1\) norm. One main problem with the Galilee data is the presence of glitches in the middle of the lake and at the track ends. Starting from equation (4), we can introduce a weighting operator that deemphasizes high residuals as follows:

\[
0 \approx r_d = W(BH^{-1}p - d) \\
0 \approx \epsilon r_p = \epsilon p,
\]

(6)

with a diagonal matrix \(W\)

\[
W = \text{diag}\left(\frac{1}{(1 + r_i^2/\bar{r}^2)^{1/4}}\right),
\]

(7)

where \(r_i\) is the residual for one component of \(r_d\), and \(\bar{r}\) a constant we choose a-priori. This weighting operator ranges from \(\ell^2\) to \(\ell^1\) depending on the constant \(\bar{r}\). It is somewhat difficult to evaluate a good \(\bar{r}\) for a particular problem because the transition between the two norms is smooth. We choose \(\bar{r} = 0.01\) cm which is very small. This choice is not based on geophysical considerations: \(\bar{r} = 10\) cm might appear a better choice since the measurements are
recorded to an accuracy of about 10 cm. It is not based on statistical choices either as done by Bube and Langan (1997). This choice of $\bar{r}$ simply gives us the most pleasing results after inversion. Because it is very small, we are essentially simulating a $\ell^1$ norm only. The weighting operator $W$ is kept constant for a number of iterations and then reevaluated. The IRLS method is guaranteed to converge to the $\ell^1$ estimate of the model parameters (Bube and Langan, 1997). The linear steps are computed with a conjugate gradient solver. Abandoning the damping in equation (6), i.e., $\epsilon = 0$, makes the IRLS method very appealing because we focus on the minimization of the data residual. This is only possible with the preconditioning of the problem.

We now test our proposed method to get rid of the outliers. We then use the fitting goals in equations (4) and (6) to produce depth images of the Sea of Galilee. Equation (4) is referred as the $\ell^2$ norm solution and equation (6) as the $\ell^1$ norm solution.

In Figure 4a, we show $p$ estimated with the $\ell^2$ norm. Although $p$ appears to be a variable of mathematical interest only, in fact, the solution $h$ is so smooth that we have difficulty viewing it. We could view the two components of $\nabla h$ but it happens that $p$ is a roughened version of $h$. Hence it is more convenient to view $p$ than the two images $\partial h/\partial x$ and $\partial h/\partial y$. We can see a lot of spurious noise everywhere in the map of Figure 4a. In addition, we can see the vessel tracks in the north part of the map. This first result is obtained after 1000 iterations which means that we essentially simulate a least-squares solution without damping. Therefore, all the noise in the data is inverted for and mapped in the final model. With noisy data, it is common practice to use a damped least-squares to minimize the effects of the noise. We can easily simulate a damped least-squares solution by decreasing the number of iterations, as explained in the preceding section. In Figure 4b, we show the roughened map of the Sea of Galilee after 50 iterations, thus recreating the solution of a damped least-squares problem. Most of the noise has been attenuated but glitches are still present in the northern part of the lake and the $\ell^1$ norm should be utilized.

Figure 4c displays $p$ estimated with the $\ell^1$ norm (e.g., equation (6) with a small $\bar{r}$). Most of the glitches are attenuated showing vessel tracks only. Some ancient shorelines in the west part and south part of the Sea of Galilee are now easy to identify. In addition, we also start to see a “valley” in the middle of the lake that probably represents the on-going rifting in this area. The data outside the sea have been also partially removed.

Figures 5a,b,c show the bottom of the Sea of Galilee ($h = H^{-1}p$) after inversion. The $\ell^1$ result is a great improvement over the $\ell^2$ maps with or without damping. The glitches inside and outside the sea have disappeared. It is also pleasing to see that the $\ell^1$ norm gives us positive depths everywhere. Although not everywhere visible in Figure 5, it is interesting to notice that we produce topography outside the lake. Indeed, the effect of regularization is to produce synthetic topography which is a natural continuation of the lake floor.

We have shown that the combined utilization of preconditioning and IRLS removes the spikes in the depth map of the Sea of Galilee. In the next section, we propose removing the ship tracks by introducing an operator in equation (6) that will model the coherent noise created by different weather and human conditions during the acquisition of the data.
Figure 4: (a) \( p \) estimated with equation (4) in a least-squares sense after 1000 iterations, which simulates a least-squares solution without damping. (b) \( p \) estimated with equation (4) in a least-squares sense after 50 iterations, which simulates a least-squares solution with damping. (c) \( p \) estimated with equation (6) in a \( \ell^1 \) sense. The spikes have been correctly attenuated.
Figure 5: (a) View of the bottom of the lake \((h = H^{-1}p)\) with the \(\ell^2\) norm after 1000 iterations, which simulates a least-squares solution without damping. (b) View of the bottom of the lake with the \(\ell^2\) norm after 50 iterations, which simulates a least-squares solution with damping. (c) View of the bottom of the lake with the \(\ell^1\) norm. Note that with the \(\ell^1\) norm, all the depth values are now positives and that the spikes have been attenuated.
ATTENUATION OF THE SHIP TRACKS

We are now half way to a noise-free image. As seen in Figure 4, the vessel tracks overwhelm possible small details at the bottom of the Sea of Galilee. In this section, we propose a strategy based on the assumption that the inconsistency between tracks comes mainly from different human and seasonal conditions during the acquisition. Unfortunately, we do not have any record of the weather and the time of the year the data were acquired. We presume that the depth differences between different acquisition tracks must be small and relatively smooth in time. That is why we propose introducing a leaky-integration operator (Claerbout, 1992) to model these “secular” data variations within our inversion scheme.

Abandoned strategy for attenuating tracks

An earlier strategy to remove the ship tracks is to filter the residual during the inversion as follows (Fomel and Claerbout, 1995):

\[ 0 \approx r_d = W \frac{d}{ds} (BH^{-1} p - d) \]
\[ 0 \approx \epsilon_p = \epsilon p, \]  \hspace{1cm} (8)

where \( \frac{d}{ds} \) is the derivative along the track. The purpose of the derivative is to remove the drift from the field data while preserving the geological features. One consequence of the derivative is that it creates more glitches and spiky noise at the track ends and at the bad data points. In addition, the use of the derivative might induce a loss of resolution in the final image. Both effects are illustrated in Figure 6 where we display the estimated \( p \) after inversion with the fitting goals in equation (8). We can see that the tracks have been attenuated, as expected. However, we lost important geological features in the middle of the lake and on the sea shores. In addition, the map is more noisy because of the aggravating effect of the derivative on bad data points. We do not fully understand why this approach works badly. One possible explanation is that the conditioning of our problem with the four operators \( W, \frac{d}{ds}, B \) and \( H^{-1} \) worsens, making the optimization very difficult. Theoretically, we could estimate a prediction-error filter (PEF) from the data residual and use it as a data-residual weight within the inversion. Unfortunately, the PEF would probably mix more glitches with good data because it would spread further in space than the derivative.

Recently, Brown (2001) proposed estimating systematic errors between tracks by analyzing measurements at points where the acquisition swaths cross. This approach has the advantage of preserving the resolution of the depth map compared to the derivative along the tracks. Brown (2001) uses this idea as a preprocessing step, however. The comparison of Figure 4b and Figure 6 teaches us two lessons. First, the filtering approach with the derivative along the tracks does not produce a good image of the bathymetry (Fomel and Claerbout, 1995). Second, based on Brown’s idea (2001), we propose introducing an operator that will adaptively model and subtract the systematic shift within the inversion scheme. In the next section, we show that by incorporating a modeling operator for the drift in the data, we can effectively remove the ship tracks without any loss of resolution in the estimated depth map.
A new fitting goal

Now, we show our new idea of removing the tracks by adaptively subtracting them within our inversion scheme. Building on Nemeth et al. (2000), we introduce a modeling operator for the ship tracks inside our fitting goal in equation (6) as follows:

$$0 \approx r_d = W(BH^{-1}p + \lambda Lq - d)$$
$$0 \approx \epsilon_1 r_p = \epsilon_1 p$$
$$0 \approx \epsilon_2 r_q = \epsilon_2 q$$

where \( L \) is a drift modeling operator (leaky integration), \( q \) is a new variable of our inversion, and \( \lambda \) a balancing constant between gridding and noise modeling. We then minimize the misfit function

$$g_2(p, q) = ||r_d||^2 + \epsilon_1^2 ||r_p||^2 + \epsilon_2^2 ||r_q||^2$$

where \( h = H^{-1}p \) estimates the interpolated map of the lake. Again, we set \( \epsilon_1 = \epsilon_2 = 0 \), and we do not iterate to completion. Note that we invert for \( p \) and \( q \) at the same time. Our hope is that by introducing the new variable \( q \) for the drift, we will obtain a cleaner depth map. For the operator \( L \), we choose a leaky integration operator such that \( y = \lambda Lq \) is the portion of data value \( d \) that results from drift. This choice is consistent with the derivative used in the abandoned strategy: since we cannot properly filter the tracks with \( \frac{d}{ds} \), we have to define a new operator, intuitively close to the inverse of the derivative, in order to model the drift in our new approach [equation (9)]. The leaky integration seems to be a good candidate. Consistent with the way we use a rough variable \( p \) to represent the smooth water depth \( h \), we now represent (for the purpose of speeding iteration) \( y \) by a rougher function \( q \). The operator \( L \) has the
following recursive form
\[ y_s = \rho y_{s-1} + q_s \quad s \text{ increasing along the data track.} \] (11)

The parameter \( \rho \) controls the decay of the integration. For \( \rho = 1 \), leaky integration represents causal integration. The operator \( \mathbf{L} \) is then appropriate to model the secular variations implied by the different season and human conditions during the data acquisition. We simply have to choose a value of \( \rho \) that best represents the variations between the different tracks. We have roughly 200 data points per track (Figure 1). With \( \rho = 0.99 \), we have \( \rho^{200} = 0.134 \), which represents a 87% amplitude decay for one track. This seems to be a reasonable decay for what we are trying to model, i.e., the drift. We keep this value of \( \rho = 0.99 \) for our results in the next section. We show that the operator \( \mathbf{L} \) removes most of the vessel tracks present in Figure 4.

The choice of \( \lambda \) in equation (9) is also critical. We tried different values by starting from a very small number and increasing it slowly. We then chose the smallest value that removed enough tracks in the final image (\( \lambda = 0.08 \)). Nemeth et al. (2000) demonstrates that the noise (the tracks) and signal (the depth) can be separated in equation (9) if the two operators \( \mathbf{L} \) and \( \mathbf{BH}^{-1} \) do not model similar components of the data space. The parameter \( \lambda \) helps us to mitigate the possible crosstalk. A similar approach has been used by Guitton (2002) to successfully remove ground-roll on common midpoint gathers.

We display in Figure 9 a comparison of the estimated \( \mathbf{p} \) with or without the attenuation of the vessel tracks. It is delightful that Figure 9b is track-free without any loss of details compared to Figure 9a. The difference plot in Figure 9c between the two results corroborates this and does not show any geological feature.

Comparing Figure 9b and Figure 6, we see that the drift-modeling strategy (equation 9) works much better than the noise-filtering strategy (equation 8). One possible explanation for the difference between the two results is that our modeling approach is more adaptive than the filtering of the residual. Indeed, by introducing the modeling operator, we basically look for the best \( \mathbf{q} \) that models the drift of the data on each track at each point. The price to pay is an increase of the number of unknowns in equation (9). The reward is a surgically removed acquisition footprint. Notice that we can identify the ancient shorelines in the west and east parts of the lake very well.

To better understand what we are doing, we show in Figures 7 and 8 some segments of the input data (\( \mathbf{d} \)), the estimated secular variations (\( \lambda \mathbf{Lq} \)) and the residual (\( \mathbf{BH}^{-1}\mathbf{p} + \lambda \mathbf{Lq} - \mathbf{d} \)) after inversion. We can notice in Figure 7b that the estimated drift seems to have reasonable amplitudes: the average drift is around 15 cm for an accuracy of about 10 cm for the measurements, which is satisfying to us. We also observe that the estimated drift is relatively constant throughout Figure 7b. Now, if we look at the estimated drift for another portion of the data (Figure 8b), we notice that the drift has more variance than in Figure 7b and oscillates between 0 to 2 m, which is a lot. In addition, the estimated drift seems to follow the bathymetry of the lake in Figure 8a.

Looking closely at the residual (Figure 8c), we notice that the drift is large where the data are noisy (Figure 8a). It is possible that the day of acquisition was very windy, which is not a rare weather condition for the Sea of Galilee (Volohonsky et al., 1983). Thus, the wind forces
the water to pile-up on one side of the lake which can explain the lower water level on the other side. In addition, the strong wind in the middle of the lake induces noisy measurements because of the waves and of the erratic movement of the ship. It is also possible that the depth sounder was not working properly that day and had problems to correctly measure the deepest part of the lake. These causes could probably explain the shape and amplitude of the estimated drift in Figure 8b, but we can’t be absolutely sure. It is very unfortunate that no daily logs of the survey were kept in order to better interpret our results, especially for such a noisy dataset.

We have shown that we can effectively subtract the tracks without any loss of resolution by introducing a noise modeling operator within our inversion scheme. In the next section, we go one step further and attempt to improve our result by using a prediction-error filter (PEF) instead of the helix derivative as a preconditioner in equation (9). We show that the PEF helps us to improve the details inside the lake, however, shoreline resolution is degraded.

**APPROXIMATING THE MODEL COVARIANCE WITH A PEF**

It is well known that the ideal regularization operator (squared) is the model covariance matrix (Tarantola, 1987). Estimating this matrix is not straightforward. We often approximate
the model covariance matrix with roughening operators like the Laplacian or the derivative. Claerbout and Fomel (2002) advocate that in principle, an “ideal” regularization operator is a PEF estimated from an a-priori model. In this section, we test the idea of using a PEF instead of the Helix derivative for the regularization operator.

Starting from equation (9), we simply replace the Helix derivative $H$ with a PEF $A_m$ estimated from a model of the lake bathymetry. Since we do not know a-priori the exact $h$, a PEF is computed from the depth map $\hat{h}$ estimated in equation (9) with the Helix derivative. We estimate a 3 by 4 filter. This procedure can be interpreted as a bootstrapping of the model covariance matrix. We bootstrapped the PEF estimation six times before converging to a satisfying result, meaning that we use the last depth map to estimate a PEF and reiterate with the new filter. Including the PEF $A_m$ into equation (9), we have the new fitting goals

$$
0 \approx r_d = W(BA_m^{-1}p + \lambda Lq - d)
$$

$$
0 \approx \epsilon_1 r_p = \epsilon_1 p
$$

$$
0 \approx \epsilon_2 r_q = \epsilon_2 q.
$$

Again, we set $\epsilon_1 = \epsilon_2 = 0$ and estimate $p$ and $q$ with conjugate gradients.

To our great surprise, after our first attempt, the bootstrapping technique did not converge, meaning that we were unable to estimate a correct PEF for the inversion. The final results were extremely low frequency. We were able to stabilize the inversion by adding some random noise.
Figure 9: (a) $p$ estimated with equation (6) with the tracks. (b) $p$ estimated with equation (9) without the tracks. (c) Difference between (a) and (b): no geological feature has leaked in.
to the model $h$ before the PEF estimation. However, the noise level needed was extremely high, around one meter. The filter we estimate from this noisy image is then far from the PEF we are looking for, to our disappointment.

In Figure 10b, we show the estimated $p$ with the PEF as a preconditioning operator after six iterations of bootstrapping with noise added to the model. To increase the contrast inside the lake, we apply a weighting function on Figure 10b that boosts up the low values of $p$ in the middle of the sea and deemphasizes the sea shores. Figure 10c displays the final result. We see that the structure inside the lake is more visible. We can almost follow this pattern up to the north side of the lake (we tried a similar weighting function on the result with the Helix derivative but with no improvement in the middle of the lake.)

Although encouraging, this result has major shortcomings. First, we have lost resolution on the sea shores. The result with the Helix derivative in Figure 10a is much better in this area. Second, it is surprising that the PEF is not able to eliminate the low frequency trend visible throughout the lake in Figures 10b and 10c. Finally, we still do not fully understand why we could not find a PEF without adding noise to the model that would allow us to converge to a satisfying depth map. Maybe we should estimate non-stationary PEFs as opposed to one PEF for the whole model space. Maybe we should try to estimate the PEF and the model at the same time, as done in the missing data problem (Claerbout, 1992). Maybe we should include a starting guess in our inversion to insure better convergence because bootstrapping is essentially a non-linear process. All these suggestions are possible niches of improvement with the Sea of Galilee dataset. Our feeling is that more work needs to be done for the PEF estimation.

CONCLUSION

The interpolation of the Sea of Galilee dataset on a regular grid is addressed as a noise attenuation problem. We saw that the combination of preconditioning and Iteratively Reweighted Least Squares with the proper weighting function greatly reduces glitches and spikes. In addition, the introduction of a noise modeling operator that accounts for the inconsistency between depths measurements on different tracks greatly reduces the acquisition footprint. These two strategies help us to unravel meaningful geological features inside the lake and ancient shorelines. Our last attempt for improving the final image with a prediction-error filter as a preconditioner produces a more detailed map inside the lake but with smoother edges than with the Helix derivative. Unfortunately, our PEF estimation was not completely successful and we think that more work in that direction is desirable.

The lessons we learn from the processing of the Sea of Galilee dataset can be reused in our daily geophysical work. It teaches us that the residual should be always looked at to derive the correct weighting functions. It also seems to teach us that it is better to model and subtract the noise than to try to filter or weight it out of the residual.
Figure 10: (a) Estimated $p$ with the Helix derivative. (b) Estimated $p$ with the PEF. (c) Scaled version of (b) to increase the contrast inside the lake.
REFERENCES


Flexible 3-D seismic survey design

Gabriel Alvarez¹

ABSTRACT
Using all available subsurface information in the design of a 3-D seismic survey, we can better adjust the acquisition effort to the demands of illumination of the target horizon. I present a method that poses the choice of the acquisition parameters as an integer optimization problem. Rays are shot from grid points on the target reflector at uniform opening and azimuth angles and their emergence positions at the surface are recorded. The optimization (an exhaustive search in this example) minimizes the distance between the ray emergence coordinates and the source and receiver coordinates of candidate geometries subject to appropriate geophysical and logistics constraints. I illustrate the method with a 3-D subsurface model that I created featuring a target reflector whose depth changes significantly across the survey area. I show that for this model the standard approach would lead to a design requiring 200 shots/km² whereas the optimum design requires only 80 shots/km² without sacrificing the illumination of the target at any depth or the logistics of acquisition.

INTRODUCTION
For the design of a 3-D seismic survey to be effective, proper illumination of the target reflector and faults must be achieved. The mark of a successful design, however, is that it not only satisfies this condition, but provides for easy acquisition logistics with the lowest possible cost. The standard practice of 3-D seismic survey design assumes implicitly that the subsurface is composed of flat layers of constant velocity. Under this assumption, a set of source-receiver geometries have been devised and used extensively (Stone, 1994). These geometries usually correspond to parallel lines of receivers at fixed distances and to parallel lines of sources, also at fixed distances. The source lines are usually arranged parallel, perpendicular or slanted with respect to the receiver lines. Inasmuch as the assumption of flat parallel subsurface layers is valid, the design process can be standardized and can be seen as a somewhat routine application of known formulas to compute the distances between individual sources and receivers; separation between lines of sources and receivers; number of active channels per source and so on. Input information to the design process is limited to range of target depths and dips, maximum and minimum propagation velocities, and desired fold of coverage.

The key assumption of flat horizontal layers does not honor the complexity often present in the geometry of subsurface layers in areas of great oil exploration or production interest.

¹email: gabriel@sep.stanford.edu
The survey designer usually ignores this discrepancy, however, partly because of mistrust of the available subsurface information and partly because of fear that exploiting that information may lead to ineffective logistics or may bias the results. Maintaining the assumption of flat layers is thus seen as a way to streamline the design process and to guarantee that the design is conservative with respect to our possibly inaccurate knowledge of the subsurface.

Survey designers often choose the source-receiver geometry from among the few standard geometries available (parallel, orthogonal, slanted, zig-zag) on the basis of uniformity of offset and azimuth in the subsurface bins (Galbraith, 1994). Wavefield sampling (Vermeer, 1998) may also be a consideration. In some cases, a 3-D subsurface model obtained from existing 2-D or 3-D data, well logs or geological plausibility is used to compute illumination maps of the reflectors of interest via forward modeling with various candidate geometries (Carcione et al., 1999; Cain et al., 1998). The geometry that provides the least distortion in illumination is chosen as the best design, after maybe tweaking it manually to fine tune its illumination response.

The methodology I propose for the optimization of the survey design poses the selection of the survey parameters as an optimization process that allows the parameters to vary spatially in response to changes in the subsurface. In a previous report (Alvarez, 2002b) I described the basic idea behind the method and illustrated it with a very simple 2-D synthetic model. In this paper I will illustrate the method in 3-D using a subsurface model I created to simulate a land survey. The emphasis will be on the description of the inversion to compute the parameters. I will show that a standard acquisition geometry will either sacrifice the offset coverage of the shallow part of the target horizon or require a large number of sources, which negatively impacts the cost of the survey. For the sake of simplicity, target depth was the only parameter I allowed to influence the spatial change of the geometry. Three geometries were computed according to the depth of the target reflector. Each geometry was locally optimized for uniformity of subsurface illumination. The optimum geometry, being more flexible, relaxes the acquisition effort without compromising the shallow part of the target reflector.

**SUBSURFACE MODEL**

To illustrate the method, I created a simplified subsurface model corresponding to a target horizon whose depth changes from about 0.3 km to about 2 km. The model simulates a land prospect, has high local dips and mild topography. Figures 1 and 2 show views of the model from the inline and the cross-line direction respectively. The model has significant dips in both directions. Similarly, Figures 3 and 4 show views of the target horizon from the inline and the cross-line direction. There are two possible reservoirs in this horizon, one at the top and one in the small structure to the right (Figure 4). The model is 10 km x 10 km with a maximum depth of about 3 km. The velocity field consists of blocks delimited by the reflectors. Within each block the velocity changes laterally as well as vertically in a gradient-based fashion as summarized on Table 1. Velocity at each point of each block is computed as 

\[ v(x, y, z) = v_0 + x \Delta v_x + y \Delta v_y + z \Delta v_z. \]
Figure 1: Subsurface model. View from the strike direction. The horizontal dimensions are 10 km x 10 km. The depth of the deepest reflector is 3 km.

Figure 2: Subsurface model. View from the dip direction. The horizontal dimensions are 10 km x 10 km. The depth of the deepest reflector is 3 km.
STANDARD 3-D SURVEY DESIGN

Table 2 shows the input data for the standard design. The two most critical parameters, since they control the cost of the survey, are the maximum dip and the minimum target depth. The maximum dip controls the spatial sampling whereas the minimum target depth controls the separation between the receiver and shot lines as I will show below. In Appendix A a give a fairly detailed description of the computation of the geometry parameters summarized on Table 2.

THE PROBLEM WITH THE STANDARD APPROACH

The survey parameters in Table 3 seem reasonable and well within the usual range, but there is a problem:

With an orthogonal geometry, there is a clear limit to the minimum offset that we can sample in all the bins as shown in Figure 5. The shortest offset in the central bin is approximately 700 m, which is too large compared to the depth of the shallowest part of the target reflector. With the basic assumption of flat layers and constant velocity, this means that the central bins will not have offsets short enough to image the shallow part of the target reflector. Since this is the main target, I can’t afford to compromise its image. From Figure 5 it should be clear that in order to decrease the maximum minimum offset we need to decrease the source line interval, the receiver line interval or both. In this particular case, we can simply halve them so that the maximum minimum offset is now just over 300 m. Obviously, halving the 

\[ \text{d} \]

doubles the number of required shots which in turn may double the cost of the survey. Furthermore, if the number of receiver lines is kept constant (which may be necessary if not enough equipment is available) the aspect ratio will also double, making the survey highly azimuthal, which may not be what we want. A possible solution is to use rectangular bins with the source interval
Table 1: Velocity information. Each block is delimited by two reflectors, numbered as indicated on Figure 1. Units are m/s.

<table>
<thead>
<tr>
<th>Block</th>
<th>Reflectors</th>
<th>$v_0$</th>
<th>$\Delta v_x$</th>
<th>$\Delta v_y$</th>
<th>$\Delta v_z$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 and 2</td>
<td>2000</td>
<td>0.01</td>
<td>0.05</td>
<td>0.5</td>
</tr>
<tr>
<td>2</td>
<td>2 and 3</td>
<td>3000</td>
<td>0.05</td>
<td>0.01</td>
<td>0.2</td>
</tr>
<tr>
<td>3</td>
<td>3 and 4</td>
<td>2400</td>
<td>0.01</td>
<td>0.01</td>
<td>0.5</td>
</tr>
<tr>
<td>4</td>
<td>4 and 5</td>
<td>3600</td>
<td>0.05</td>
<td>0.01</td>
<td>0.1</td>
</tr>
<tr>
<td>5</td>
<td>5 and 6</td>
<td>5000</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Table 2: Input to the standard design.

<table>
<thead>
<tr>
<th>$V_{\min}$</th>
<th>$\theta_{\max}$</th>
<th>$f_{\max}$</th>
<th>$z_{\min}$</th>
<th>$z_{\max}$</th>
<th>$f_{old}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>60</td>
<td>60</td>
<td>300</td>
<td>3000</td>
<td>24</td>
</tr>
</tbody>
</table>

equal to twice the receiver interval. This keeps both the number of shots and the aspect ratio constant, but may be an undesirable solution if significant dips are present in the strike direction (I am assuming that the source lines are in the strike direction).

Figure 5: Recording box. Receivers are along the horizontal lines and sources along the vertical lines. The minimum possible offset for the bins in the middle of the box is of the order of 700 m, much too large to image the target reflector at 300 m depth.

THE PROPOSED APPROACH

The message in the previous section was that the standard design approach incurs compromises between the requirements of image quality and the cost of the survey (and the amount of collected data). My proposed approach (Alvarez, 2002b) avoids those compromises by posing the design as an optimization problem in which the requirements of image quality and the survey cost are balanced against each other within the constraint of sound acquisition logistics.
Table 3: Parameters of the standard design: $dr$ is the receiver interval, $ds$ is the source interval, $drl$ is the receiver line interval, $dsl$ is the source line interval, $nchl$ is the number of channels per line and $nrl$ is the number of receiver lines in the recording patch. Units are meters.

<table>
<thead>
<tr>
<th>$dr$</th>
<th>$ds$</th>
<th>$drl$</th>
<th>$dsl$</th>
<th>$nchl$</th>
<th>$nrl$</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>20</td>
<td>400</td>
<td>500</td>
<td>150</td>
<td>8</td>
</tr>
</tbody>
</table>

Note that the compromises in the design stem from insufficient input data, or what amounts to the same thing, an inappropriate subsurface model. The implicit model of flat layers with constant velocities forces us to use the same acquisition parameters for the entire survey area. In the example described above, I “punished” the design because of the need to image the shallow part of the target horizon, although Figures 3 and 4 show that the target reflector is only shallow in about 10 or 20% of the survey area.

The upshot, of course, is that if we can accurately establish the correspondence between the subsurface area of the shallow reflector and the part of the surface area whose sources and receivers contribute to its image, then we could use the dense acquisition parameters in that part of the survey area and use the more standard parameters in the rest of the survey area. This is the key idea that allows us to get optimum image quality with the least acquisition effort. A similar approach can be used to locally increase the offsets for high-dipping reflectors, or to increase the azimuth coverage of a locally fractured reservoir.

I will illustrate in some detail each step of my approach with the model shown in Figures 1 and 2.

**Building of the Model**

I created the model from 5 cross-sections in the inline direction and 4 cross-sections in the cross-line direction using the Integra Software. In this system the reflector surfaces are represented by B-splines which allows the accurate computation of the normal to the reflector at every point (Pereyra, 2000). In general, the structural and velocity model is constructed by integrating all available information from geology, well logs and previous 2D and 3D seismic data. The model does not need to be particularly detailed, although the main features are expected to be reasonably well represented. In the present case the model is not derived from any data and is purely conceptual.

**Exploding Reflector Modeling**

The subsurface model is used to trace rays up to the surface from points on the reflectors of interest at finely-sampled opening (polar) and azimuth angles. The number and location of

---

2Developed by Weidlinger and Associates
the reflecting points on the reflectors of interest, as well as the number of rays shot from each such point, is a matter of design. The important point is that they represent our ideal subsurface illumination response. Departures from this ideal will be penalized in the optimization process.

For this paper I used the Integra software to do the ray tracing. I used Gocad to compute a stratigraphic grid draped over the target reflector (Figure 4) and used the grid centers as the starting points for the rays. The grid is 25 x 25 m and 20 rays were shot from each point. The range of azimuths was from 0 to 135 degrees in intervals of 45 and the range of opening angles was from 0 to 60 degrees in intervals of 15 degrees. Figure 6 shows ray cones emanating from a few selected points on the target horizon’s surface.

Figure 6: Exploding reflector rays. Rays from a few of the reflecting points on the target horizon. There are several thousand such reflecting points, so the resulting density of rays is very large.

Integra ray tracing program computes, for each ray, its travel time and the coordinates of the emerging point at the surface. The program also records the coordinates of the normal to the reflector at the starting point, as well as the opening and azimuth angle of the corresponding ray. In the present example I used this information to separate the rays originating at different depths in the target reflector.

Geometry Optimization

The most important part of my approach is using the information computed from the rays to optimize the layout of the sources and receivers. There are several ways in which this can be accomplished according to how much departure from the standard geometry we are prepared to accept.

1. Choose an orthogonal, parallel, or slanted geometry with parallel receiver lines as well as parallel source lines. In this case the optimization consists of choosing receiver and shot sampling interval, receiver and source line separations, number of channels per active receiver lines and number of active receiver lines per shot. These parameters can be changed spatially to conform with the requirements of the image, but the basic
geometry is unchanged. In other words, the optimum geometry is the composite of regular geometry patches each of which is optimum for a given part of the survey area.

2. Choose a dense receiver patch with similar receiver sampling in the inline and the cross-line direction and optimize the location of the shots which will not form continuous source lines. The source positions will be computed to optimize uniformity of the subsurface illumination. Optimizing the source positions is particularly attractive in environments in which the layout of the receivers is easy but the drilling of shot holes is expensive. The use of vibrators may also be optimized, since the road paths can be used to constrain the source positions at the design stage.

3. Allow shots and receivers to be placed in geometries other than continuous source and receiver lines. The constraints of offset and azimuth distribution as well as inline and cross-line offset sampling migration requirements can then be explicitly posed as optimization constraints.

For the sake of this example, I used the first approach, which is the least ambitious.

**Preprocessing**

For each reflecting point and azimuth, rays corresponding to the same opening angle on opposite sides of the normal are linked together as dual rays. Rays whose emergence point (or that of its dual) fall outside the permit area for sources and receivers are discarded. Also discarded are rays for which the total traveltime is longer than the trace length. The remaining rays are assigned source or receiver positions according to the shortest distance between their emergence point and the closest source or receiver position for each geometry. Once a ray is classified as, say, source, its dual will be classified as receiver. Note that this classification is geometry-dependent. Figure 7 shows an example of the classification of a ray and its dual for two candidate geometries. The classification is done for all valid rays and the total distance that the emergence points have to be moved to conform with each geometry is recorded and saved. Minimizing this distance is equivalent to maximizing uniformity of illumination. For each geometry, I compute and save all the relevant parameters such as fold of coverage, maximum and minimum maximum offsets, aspect ratio, number of receiver per patch, etc. This information is saved and will be used as geophysical constraints for the optimization as described below. I also compute all relevant statistics of each geometry, such as number of sources, number of receivers, and receiver- and source-line cuts. This information is used to apply logistic constraints to the optimization. The cost of the survey, in particular, may largely depend on those statistics.

**Geophysical Constraints**

In this case, by far the most important constraint was that the maximum minimum offset not be greater than the depth of the target reflector. The rays were split into three groups according to the depth of their reflecting point in the target reflector: from 300 to 400 m, from 400 to
Figure 7: Classification of ray emergence points as sources or receivers. Vertical lines are source lines and horizontal lines are receiver lines. Two geometries are represented. The left panel illustrates a sparse geometry for which the ray in the top left would be classified as a source. The right panel shows the same ray which will now be classified as a receiver for this less sparse geometry.

700 m and deeper than 700 m. These ranges were chosen such that the densest acquisition is only used in the part of the surface that contributes the most to the image of the shallow part of the target horizon, and the coarser geometry be used in the part of the surface area that corresponds to the deeper parts of the target horizon with a transition zone in between. The different zones themselves were chosen according to the emergence coordinates of the normal rays. Figure 8 shows the three zones in map view superimposed on the topography. The dark shaded zone in the middle corresponds to the shallow part of the target horizon, the annular zone to the 400-700 meter depths and the darkest zone to the deeper parts of the target horizon. The important point is that this area is a small part of the total survey area. In fact, if we circumscribe a rectangle on this zone we find that its area is only about 3.5 km$^2$. Similarly, the area of zone 2 is about 7.5 km$^2$ which leaves zone 3 with an area of 89 km$^2$.

Another constraint was the aspect ratio of the recording patch. This is intended to maintain a range of short offsets where the reflector is shallow and longer offsets where the reflector is deeper. Also, this has the effect of avoiding abnormally high folds due to the contribution of long offsets that are not required where the reflector is shallow. Total fold was also considered a constraint to make sure that the minimum fold requirement is honored and to penalize geometries with much higher folds.

**Logistic Constraints**

Since in this case my trial geometries are completely regular, the only logistical constraint I applied was to ensure that the number of required channels satisfies the number of available channels in the candidate recording equipments for the acquisition of the survey. In particular, no penalty is given to a geometry that exactly matches the number of channels in any of the
candidate recording equipments. Very high penalty is given to geometries that leave many channels unused and of course the highest penalty goes to any geometry that requires more channels than are available in any of the recording equipments. For example, if two recording equipments with 1000 and 2000 channels are available, a geometry that requires 900 channels is given a low penalty, one that requires 1100 channels is given a high penalty and one that requires more than 2000 channels is given the highest penalty.

The Optimization

Recall that in this case I chose to design the survey as a collection of three orthogonal surveys, the definition of each requiring only 5 parameters: source and receiver interval, source and receiver line interval and number of receiver lines per recording patch. Other considerations such as number of shots per salvo, number of stations to roll-along in the inline direction and number of receiver lines to roll-along in the x-line direction are also design parameters but I fixed them to be the number of shots between two adjacent receiver lines, the number of receivers between two adjacent source lines and one, respectively. The upshot is that the model space has only 5 parameters for each zone and an exhaustive search can be employed among all candidate geometries. In more irregular geometries, we may wish to invert for the parameters of each salvo and a micro-genetic algorithm (Alvarez, 2002a) will be a better choice.

The fitness function has two components: one to minimize the objectives and one to guarantee that the constraints are honored.

$$f_i = (1 - \lambda) \sum_{j=1}^{m} \delta_j a_{ij} + \lambda \sum_{j=1}^{n} \epsilon_j c_{ij}$$

(1)

where $i$ is the index that represents every trial geometry, $\lambda$ is the factor balancing the two
contributions to the fitness function, \( m \) is the number of objectives, \( a_{ij} \) is the figure of merit of the \( j \)th objective for \( i \)th geometry, \( \delta_j \) is the relative weight of the \( j \)th objective, \( n \) is the number of constraints, \( \epsilon_j \) is the relative weight of the \( j \)th constraint and \( c_{ij} \) is the figure of merit of the \( j \)th constraint for the \( i \)th geometry. In this case I chose \( \lambda = 0.5 \) which means that I am giving equal weight to the minimization of the objectives and to the satisfaction of the constraints.

The main objective, as mentioned before, is uniformity of target illumination, which requires minimization of the total distance that the emergence ray positions had to be moved to conform with each geometry. Also, since this is a land survey, the main factor in the cost of the survey is the number of shots. Therefore, I used the minimization of the number of shots as the second objective of the optimization. Finally, I used the total receiver- and source-line cut as additional, though less important, objective.

Notice that the constraints are not linear and that they may be partially fulfilled with partial penalties applied. The figures of merit assigned to the objectives and the constraints are normalized between 0 and 1, except when a constraint is completely violated, for example if the required number of channels is larger than the maximum number of available channels, as mentioned before. I made no attempt to differentiate cost between the available recording equipments although in practice this is likely to be an important issue.

The relative weights on each objective and on each constraint for the three zones are summarized on Table 4.

Table 4: Weights for the objectives and constraints applied in each zone: \( \delta_1 \) is for illumination, \( \delta_2 \) is for the number of shots, \( \delta_3 \) is for receiver- and source-line cut, \( \epsilon_1 \) is for maximum-minimum offset, \( \epsilon_2 \) is for number of available channels, \( \epsilon_3 \) is for aspect ratio and \( \epsilon_4 \) is for fold of coverage.

<table>
<thead>
<tr>
<th>Zone</th>
<th>( \delta_1 )</th>
<th>( \delta_2 )</th>
<th>( \delta_3 )</th>
<th>( \epsilon_1 )</th>
<th>( \epsilon_2 )</th>
<th>( \epsilon_3 )</th>
<th>( \epsilon_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.7</td>
<td>0.25</td>
<td>0.05</td>
<td>0.4</td>
<td>0.2</td>
<td>0.3</td>
<td>0.1</td>
</tr>
<tr>
<td>1</td>
<td>0.6</td>
<td>0.3</td>
<td>0.1</td>
<td>0.3</td>
<td>0.2</td>
<td>0.3</td>
<td>0.2</td>
</tr>
<tr>
<td>1</td>
<td>0.6</td>
<td>0.3</td>
<td>0.1</td>
<td>0.1</td>
<td>0.4</td>
<td>0.2</td>
<td>0.3</td>
</tr>
</tbody>
</table>

GEOMETRY WITH THE PROPOSED METHODOLOGY

As mentioned above, I subdivided the rays (and the survey area) into three groups. For each group I computed the optimum geometry using the corresponding subset of rays and the corresponding constraints as summarized in Table 5. The maximum minimum offset, the total fold of coverage, the maximum offset and the aspect ratio were chosen differently for each zone. The source and receiver interval, however, were kept constant, meaning that the surface bins are the same, and were not included in the table. I did not apply any constraint to the offset or azimuth distribution in those bins, although I will implement that in the future. I tried a total of 4608 geometries for each zone with parameter values as summarized on Table 6. Note
Table 5: Constraints applied in each zone: $c_1$ is for maximum minimum offset, $c_2$ is for available channels, $c_3$ is for aspect ratio, and $c_4$ is for fold.

<table>
<thead>
<tr>
<th>Zone</th>
<th>$c_1$</th>
<th>$c_2$</th>
<th>$c_3$</th>
<th>$c_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>300-400</td>
<td>2000,3000,5000</td>
<td>1-3</td>
<td>24-36</td>
</tr>
<tr>
<td>2</td>
<td>500-600</td>
<td>2000,3000,5000</td>
<td>1-3</td>
<td>24-36</td>
</tr>
<tr>
<td>3</td>
<td>800-900</td>
<td>2000,3000,5000</td>
<td>1-2</td>
<td>24-32</td>
</tr>
</tbody>
</table>

That except for the receiver and group interval, the parameters are different in each zone. Also different are the constraints as shown in Table 5. The parameters of the resulting geometry

Table 6: Parameters for trial geometries in each zone. Units are in meters.

<table>
<thead>
<tr>
<th>Zone</th>
<th>$drl$</th>
<th>$dsl$</th>
<th>$nrl$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>300,320,340,360,380,400,420,440</td>
<td>300,320,340,360,380,400,420,440</td>
<td>6,8,10,12</td>
</tr>
<tr>
<td>2</td>
<td>380,400,420,440,460,500,540,580</td>
<td>380,400,420,440,460,500,540,580</td>
<td>4,6,8,10</td>
</tr>
<tr>
<td>3</td>
<td>540,560,580,600,640,680,720,760</td>
<td>540,560,580,600,640,680,720,760</td>
<td>4,6,8,10</td>
</tr>
</tbody>
</table>

Table 7: Parameters for the optimum geometry in each zone. Units are meters.

<table>
<thead>
<tr>
<th>Zone</th>
<th>$dr$</th>
<th>$ds$</th>
<th>$drl$</th>
<th>$dsl$</th>
<th>$nrl$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td>20</td>
<td>380</td>
<td>320</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>20</td>
<td>380</td>
<td>440</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>20</td>
<td>760</td>
<td>720</td>
<td>10</td>
</tr>
</tbody>
</table>

for the three zones are summarized in Table 7. The parameters are significantly different, especially between zone 1, corresponding to the shallow part of the target horizon and zone 3 corresponding to the deep part of the same horizon. Notice that having these different parameters does not in itself compromise the logistics, since the distance between the receiver lines in zones 1 and 2 is half that in zone 3. In fact, for zone 3 the inversion gave me a handful of geometries that satisfied the objectives and the constraints equally and I chose the one that had this property. Logistically, all that would be required is to add an additional receiver line between two adjacent receiver lines in zones 1 and 2 assuming that we have enough equipment (and we do, since that was a constraint to the inversion). The different separation of the source lines is even less of a problem since we can in principle drill the shot-holes along any line we want.
The bottom line

The bottom line in terms of cost of the survey, is that the standard dense geometry requires 200 shots/km$^2$ (50 shots/km and 4 receiver lines/km). For this model (100 km$^2$), this means 20000 shots which is extremely high. The optimum design uses the dense parameters 156 km$^2$ (50 shots/km and 3.125 receiver lines/km) only in zone 1, whose area is about 3.5 km$^2$; uses the intermediate design of 148 km$^2$ (50 shots/km and 2.95 receiver lines/km) only in zone 2 whose area is about 7.5 km$^2$; and uses the sparse geometry of 70 km$^2$ (50 shots/km and 1.38 receiver lines/km) in the remaining 89 km$^2$. This gives a total of about 8000 shots, less than half those of the standard geometry. This saving in the number of shots is obtained without compromising the image of the target reflector at any depth and without significantly upsetting the logistics of the acquisition.

DISCUSSION

It is worth repeating that it is the simplicity of the flat layer model which makes the standard design approach so inflexible. This in turn forces compromises between the geophysical requirements of the survey and its cost. Moreover, since the geometry is fixed, subsurface variations would imply variations in the illumination as well. Exploiting all the available subsurface information has two main benefits: first, it allows target illumination to be the main goal of the design, and second it allows the geometry parameters to be locally optimum to satisfy the target local characteristics, such as depth, dip, curvature or presence of fractures.

Although not illustrated here, the resulting geometry should be used to compute illumination maps of the subsurface targets to make sure that the resulting illumination is indeed appropriate. If it is not, the density of rays in the problem areas may be increased and the process repeated perhaps with more relaxed constraints. Also, comparison with illumination maps computed with standard geometries should allow a better appreciation of the quality of the final design.

Different source-receiver geometries can be tested against each other by the optimization process, and, if desired, the optimum geometry can incorporate elements of each. Notice also that sources do not need to be along continuous lines (in land acquisition at least) and this fact can be exploited to improve the fit of the geometry to the emergence position of the rays, that is, to the illumination. The receivers are forced to be stringed together, but even so, we can use additional strings and stagger them to allow some flexibility without upsetting the logistics of the acquisition.

Another advantage of the methodology proposed here is the information we can gather about the relative contribution of each source (or group of sources), to the target illumination. This is important when we try to overcome obstacles that prevent us from placing sources or receiver in their design positions. A river, for example, may force the displacement of sources or receivers several hundred meters. If the path of the river is known, we can use it as a constraint in the placement of sources and receivers in the optimization stage to compute the best geometry compatible with it. Even if we can’t anticipate the presence of the obstacle at design
time, we can still benefit from our knowledge of how much contribution the affected sources or receivers have on the target illumination. This information is of great help in choosing alternative positions to make up for those sources or receivers.

Finally, I would like to emphasize that the most important contribution of my methodology is flexibility. I have illustrated only one issue here, that of shallow targets, but you can easily think of other issues that would benefit from the more flexible approach. What constraints and what relative weights to assign to each one is clearly a problem-dependent decision and should reflect all the subsurface knowledge as well as the particulars of the survey.

CONCLUSION AND FUTURE WORK

I have illustrated my methodology for flexible survey design with an example in which a shallow reflector required a particularly dense acquisition geometry and have shown that it is possible to reduce considerably the number of sources by concentrating the acquisition effort where it is really required and relaxing it where it is safe to do so.

There are many issues that remain to be investigated, among them using the bin distribution of offsets and azimuths as constraints, considering multiple targets and non-standard geometries. The basic optimization method is already in place through the use of micro-genetic algorithms.

A critical issue to investigate is the influence of errors in the initial subsurface model in the computed geometry. Ultimately, I would like to quantify independently the influence of errors in reflector depths, reflector dips and velocities.

ACKNOWLEDGMENTS

I would like to express my deepest appreciation to Dr. Victor Pereyra for letting me use the Integra software for all the ray tracing in this paper. I would also like to thank Laura Carcione and Dr. Pereyra for teaching me the basics of the system and for providing me with advise and encouragement.

APPENDIX A

STANDARD GEOMETRY

Recording Geometry

To simplify the analysis I will assume an orthogonal split-spread geometry with the shot in the middle of the patch as is usual in land acquisition. I will not attempt to compare offset and azimuth distribution with other competing geometries such as parallel, slanted, or zig-zag
but instead will assume that orthogonal geometry was already selected and that I just need to compute its parameters.

Bin size

The inline bin size is chosen from the sampling requirement to avoid aliasing of the steepest dip of interest. This requirement is represented by the equation

\[ \Delta x \leq \frac{V_{\text{min}}}{2 f_{\text{max}} \theta_{\text{xmax}}} \]  

(A-1)

where \( V_{\text{min}} \) is the minimum velocity of interest, \( f_{\text{max}} \) the maximum frequency expected in the data and \( \theta_{\text{xmax}} \) the maximum inline dip. With the values in Table 2, I got \( \Delta x = 19 \) which I will round to a standard value of 20 m. With this, the inline bin size is 10 m. I could use a similar computation to find the minimum cross-line sampling, but I will simply assume that the bin is square since the model is complex enough to have significant dips in directions other than the inline direction (Figures 1 and 2).

The bin size is thus 10x10 m and the source and receiver intervals are therefore both equal to 20 m.

Maximum offset inline and cross-line

The maximum offset inline essentially controls the maximum image depth and so can be chosen simply as the maximum depth of interest which in this case is 3000 m. The maximum cross-line offset controls the azimuth distribution of the traces in the subsurface bins. We want the acquisition effort to be concentrated on those azimuths more relevant to the image, but since in this case the model shows high dips in both directions, a wide azimuth distribution is desirable and therefore I choose the maximum cross-line offset also equal to 3000 m, that is, aspect ratio of one.

Source line and receiver line interval

In this case the desired fold for a 10 m x 10 m bin is 24. This fold can be decomposed as, say, 6 inline and 4 cross-line. Since the fold cross-line is half the number of receiver lines in the recording patch, this means the recording patch will have 8 receiver lines. The distance between the receiver lines \( drl \) can be computed as

\[ drl = \frac{2 * h_{\text{xmax}}}{nrl - 1} \]  

(A-2)

where \( h_{\text{xmax}} \) is the maximum offset cross-line, 3000 m in this case, and \( nrl \) is the number of receiver lines per patch, 8 in this case. This gives \( drl = 400 m \).
The source line interval \((dsl)\) can be similarly computed from the expectation that the inline fold be 6.

\[
Fold_x = \frac{nchl\Delta x}{2dsl}
\]  
(A-3)

Here \(nchl\) is the number of channels per receiver line, 300 in this case (maximum inline offset of 3000 m and receiver interval 20 m), \(\Delta x\) is the receiver interval, 20 m, and \(Fold_x\) is 6. This gives \(dsl = 500\) m. If we wanted to apply the theory of symmetric wavefield sampling we might set \(dsl = 400\) m. Here I will not use the symmetric wavefield sampling theory.

**Salvo**

The salvo is the number of shots that are fired without moving the receiver template, so in this case it will be 20 shots since \(drl = 400\) and \(\Delta s = 20\).

**REFERENCES**


Vermeer, G., 1998, 3-D symmetric sampling: Geophysics, 63, no. 05, 1629–1647.
Ocean-bottom seismometers in Japan

Ioan Vlad

ABSTRACT
Ocean-bottom seismometers are well-tested, functional tools commonly used in crustal seismology. They can be deployed much deeper and are more robust than ocean-bottom cables, being the only type of instrument for 4-C surveys at depths greater than 1500m. I present the state-of-the-art of the Japanese OBS technology and the logistics associated with it to the seismic industry reader.

INTRODUCTION
Unlike Ocean-Bottom Cables (OBCs), in which receivers are strung on a cable, Ocean-Bottom Seismometers (OBSs) package the receivers as individual units. While OBCs do not go deeper than 1500m, OBSs are routinely operated at depths down to 6000m, which makes them the only currently available option for recording 4-C (three-component geophone plus hydrophone) data above deepwater reservoirs. They have been continuously developed and are currently employed by the crustal seismology research community, but only occasionally by the seismic industry. I became acquainted with the Japanese OBS technology during a crustal seismology cruise and I present its state-of-the-art to the seismic industry readers.

I describe the short-period OBS model developed at the University of Tokyo in the early 90’s, owned by Japan Marine Science & Technology Center (JAMSTEC) and currently operated and improved by Nippon Marine Enterprises Ltd. (NME). I refer to it as the “Japanese OBS”. I selected it among the 14 OBS models known to me because of several factors that make it interesting to a seismic industry audience: like an OBC, it has 4-C recording capability in the frequency band of active seismic surveys, but unlike an OBC, it can operate at depths down to several thousand meters. Additionally, I considered for my selection the criteria of logistical robustness and of extremely high reliability proven in hundreds of deployments of a uniform pool of instruments.

1email: nick@sep.stanford.edu
2On R/V Kaiyo, cruise KY-03-01, conducted as part of the U.S. - Japan Collaborative Research Project "Multi-Scale Seismic Imaging of the Mariana Subduction Factory". “R/V” denotes “Research Vessel”.
3No moving parts to hamper recovery, small size of the instrument even with the sinker weights on, hydrodynamic shape to minimize: 1. drifting while sinking and 2. water current induced movement while recording.
4Greater than 98% recovery rate from the ocean bottom and greater than 95% data recovery rate.
Japanese short-period OBSs consist of a glass sphere inside a plastic shell, on which a base frame and a retrieval system are fixed.

Figure 1: The glass sphere alone, unopened. Only the metal bands have been removed, their outline highlighted with a marker prior to removal. The manometer dial is very visible. The square recording unit sitting on top of the round battery pack is distinguishable through the glass. The sphere is seated in a special stand with a rubber collar on bearings and with two openings for the two halves of the sphere to be dismantled.

The glass sphere (Fig. 1) is manufactured by Benthos Technologies, Inc., USA, has a diameter of 17 inches and houses: 1. a self-gimbaling, three-component, 4.5 Hz Mark Products L-28 geophone, 2. a recording unit with a capacity of 2Gb on DAT or 8 Gb on hard disk and with circuitry for dynamic lossless compression, and 3. sealed batteries that do not generate hydrogen (Fig. 2). Because the OBSs record the data locally, they use accurate clocks such as Seascan Precision Timebase, with drift rates of $1 : 3 - 5 \times 10^{-8}$ ($< 0.5$ ms/day before correction and $< 10$ ms/yr after correction). The sphere is actually an ensemble of two glass hemispheres, held together by a low-degree vacuum. The vacuum approximately corresponds to the atmospheric pressure at the altitude of 3700 m and allows the ensemble to be checked for leaks before deployment by a manometer readable from outside the glass sphere. The hemispheres are sealed at the joint with latex and electrical tape and engirdled by two metal bands placed along great circles on the sphere. The metal bands are rigorously checked to be orthogonal to one another so that they do not slip off the circle or exert a shearing force that would separate the hemispheres enough to damage the latex. The sphere is traversed by an electrical contact point for the hydrophone cable, one for reading and synching the clock, and by a valve for modifying the pressure inside. The sphere is designed to withstand depths to 6000 m and is responsible both for the buoyancy of the entire OBS ensemble (see Table 1 for exact figures) and for much of the extra cost over the price of a land three-component seismometer.5

The price of a short-period OBS manufactured in Japan is of the order of magnitude of $40k. However, the sphere may be cheaper to manufacture if it were to withstand only to drillable depths rather than 6000 m. Manufacturing more instruments would also drive down costs. At present only slightly more than 100 Japanese short period instruments have been produced, and the number of OBSs of any kind whose existence has been made public does not surpass 500 for the entire world.
The glass sphere is placed inside a plastic shell, on which the base frame and the retrieval system are attached. The base frame is metallic; two heavy iron cylinders are welded to it. It acts as a sinker and it also ensures seafloor mechanical coupling. The shell is attached to the frame in only two points, which will be electrolytically corroded by the retrieval system, leaving the frame assembly on the ocean bottom and the OBS floating towards the surface. Leaving the frame on the seafloor has led to environmental complaints about rusting iron as a pollutant and to fishing industry complaints about nets being torn by frames, so in the future they may be replaced by concrete plates. The current horizontal size of the base frame is 1.2m by 1.0m, and the height of the shell-frame ensemble is 0.6m (Fig. 3).

The retrieval system consists of several components. The first is a sonar transceiver which listens for a certain signal from the retrieving vessel and can transmit back information. The sonar batteries are housed in a steel cylinder that protects the transponder as well. The transducer is fixed separately on the plastic shell of the OBS. The second retrieval element is the electrolytic corrosion system, which is also fed from the sonar batteries. The system is designed in such a way that should a short-circuit occur, the corrosion would begin, sending the OBS to the surface rather than leaving it stranded on the seafloor. The third element of the retrieval system is a strobe with its own C-cell batteries (Fig. 4). The fourth element is a radio transmitter with its own C-cell batteries. Both the strobe and the radio transmitter have an automatic, pressure-triggered switch and an overriding manual one. Just before launching the...
OBS into the water, the switch is set on automatic. The strobe will flash and the antenna will send signals at atmospheric pressure, but will not function when the OBS is at depth. Also, the strobe has a light sensor so that it will only flash in low-light conditions.

Figure 4: Japanese OBSs being prepared for deployment, shown from the opposite side than in Fig. 3, to get a better view of the strobe.

![nick1-view_opp](NR)

Table 1: Weights of Japanese OBS components, after Ito et al. (2002b)

<table>
<thead>
<tr>
<th>Item</th>
<th>Weight in air (kg)</th>
<th>Weight in water (kg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sinker iron cylinders</td>
<td>52.28</td>
<td>45.62</td>
</tr>
<tr>
<td>Base frame without sinker</td>
<td>3.22</td>
<td>2.58</td>
</tr>
<tr>
<td>Sphere with sensors, recording unit and batteries</td>
<td>31.00</td>
<td>-13.90</td>
</tr>
<tr>
<td>Sonar transponder</td>
<td>4.25</td>
<td>2.30</td>
</tr>
<tr>
<td>Sonar transducer</td>
<td>2.60</td>
<td>1.20</td>
</tr>
<tr>
<td>Strobe</td>
<td>1.80</td>
<td>1.00</td>
</tr>
<tr>
<td>Radio beacon</td>
<td>1.90</td>
<td>1.05</td>
</tr>
<tr>
<td>Hydrophone</td>
<td>0.40</td>
<td>0.20</td>
</tr>
<tr>
<td>Total at deployment</td>
<td>97.45</td>
<td>40.05</td>
</tr>
<tr>
<td>Total at retrieval</td>
<td>41.95</td>
<td>-8.15</td>
</tr>
</tbody>
</table>

OBS logistics aboard R/V Kaiyo

OBS deployment is quite simple: the OBS clock drift is measured, the whole assembly goes through a last quality check, the retrieval system switches are set on automatic, and the OBS is lowered into the water (Fig. 5). The clock drift is measured along a period of several hours before the launch of the OBS, and just minutes before the launch the clock is re-synched with the GPS time signal. The instrument is lowered into the water with a small crane. This operation is not likely to be hampered by bad weather because of its simplicity. Sometimes sea currents may cause the OBS to land at a different position than planned. The average drift was
Figure 5: The OBS is gently lowered into the water using a small crane. The items closest to the viewer are the sonar transceiver and transducer. The strobe is also visible as a horizontal appendage on the upper part of the OBS. Without the sinker weights, the center of gravity of the OBS shifts so that the radio beacon and the strobe are standing upright.

6.5% of the water depth for the U.S. 2-C OBSs (Gunther et al., 2002). The position is found with a precision of 2% of sea depth by watching the OBS at descent and at retrieval with the sonar array of the ship, or with a precision of 5 to 25 m by triangulating water wave arrivals (Gunther et al., 2002). Currently there are no provisions for guiding the OBS during descent. Japanese OBS descent rates are between 79 and 85 m/min. These rates are not constant (Ito et al., 2002b), which may indicate turbulence caused by the lack of hydrodynamicity of the base frame, especially since the ascent rates, when the base frame is absent, are constant. Brainstorming for potential future measures to improve the accuracy of the drop reveals two main methods: guiding fins remotely controlled from the surface by acoustic modem and increasing the descent speed. Descent can be accelerated by increasing sinker weight and improving the hydrodynamic properties of the OBS ensemble, by including a simple compressed-air propulsion system or even by reducing the water resistance through super cavitation. Care should be exercised so as not to damage the instrument by a hard impact with the seafloor.

After the OBSs are deployed, the survey is shot. The ship may tow streamers as well. For retrieval, the ship sends to each individual OBS a particular signal by sonar. The electrolytic corrosion process starts, lasting on average 13 minutes. The OBS then rotates 90 degrees to adjust to its new gravity center, which will bring the strobe and the radio beacon into an upright position. It then lifts towards the surface at a constant rate of 66.5 m/min. As at descent, the ship uses the sonar array to monitor the position of the OBS, so that it will be very close to the OBS when it surfaces. In most instances of OBS retrieval during the cruise, the
ship was able to maneuver so that the OBS would surface within 50 m in front of the ship.\textsuperscript{6} The OBS is then simply fished out with a net (Fig. 6), and samples of sediment on it are collected if there is any such scientific interest. The OBS is then washed, the clock is checked again with respect to the GPS time reference, then it is dismantled. The current data collection procedure involves taking apart the glass sphere, but that may be made unnecessary with future improvements. Even before improving the instruments, the recovery rate of Japanese OBSs was 97.2\%, computed on 472 total individual OBS deployments. Instruments that did not surface were recovered by a Remotely Operated Vessel (ROV) in order to analyze the reasons of failure (Ito et al., 2002b). The three main causes of failure were: 1. Water leaks in the glass sphere due to improper sealing and/or sphere shaping defects. This was addressed by placing a pressure gauge inside the sphere to check the integrity of the seal before launch. 2. Leaking into the transponder pressure housing due to a structural defect of the O-ring seal at the pressure housing end cap. Mandatory checks of that component before launch were instituted. 3. Release devices failing to operate. Other improvements are currently under study (Ito et al., 2002a). They may include increasing transponder battery life above the current level of three months, raising transmission rate above the current 180 dB, and retrieving data without opening the glass sphere. After improvements, the recovery rate is expected to be above 99\%.

The retrieval of about 100 OBSs (Fig. 7) placed along a 2-D line at a distance of 5 km and 10 km apart from each other took nine days, which is three times longer than their deployment and two times longer than shooting the 2-D line with the airguns. Much of the time was spent by the ship waiting for the electrolytic corrosion to finish and for the OBS to surface from

\textsuperscript{6}R/V Kaiyo is a very maneuvrable catamaran with thrusters at all four corners.
depths sometimes greater than 4000 m (approx. 1 hour ascent time). The preparation time of an OBS for deployment was about 10 minutes, using a team of four technicians.

Figure 7: Stacks of OBSs on the deck of the ship. After the base frame is left on the seafloor, they will take only 1/4 of the previous space. The ergonomic arrangements on the ship permit operations with hundreds of instruments.

OTHER OBS MODELS AND COMPARISON WITH OBC TECHNOLOGY

The first OBS, built by Maurice Ewing in 1937, used rock salt for the release mechanism, containers with gasoline (incompressible and more lightweight than water!) for flotation and a pocket Hamilton watch for timing. It went as deep as 4500 m and is on display at the SEG Virtual Museum. Today, NME operates a catamaran with a sonar array that can deploy 106 short-period OBSs for three months and retrieves them in 99% of cases. The technology has changed in the meanwhile, spinning off along the way broadband submersible instruments in the seventies and OBCs in the mid-nineties.

Currently the U.S. National OBS Instrument Pool (OBSIP) administers 135 short-period, 2-C (hydrophone and vertical geophone) OBSs and 109 4-C large broadband ones, designed for extended deployments longer than one year at a time. These instruments have been built and are maintained by Scripps Institute of Oceanography, Lamont-Doherty Earth Observatory and Woods Hole Oceanographic Institute, and are available for US academic and industrial use. Broadband (0.00277 Hz- 50 Hz) OBSs are also produced by Guralp Systems. The U.S. instruments that the Japanese ones resemble most are the 19 OBSs maintained by the University of Texas Institute of Geophysics (UTIG), developed there in 1976 by scientists formerly involved in the lunar seismograph program. Another significant pool of instruments exists in Germany. It consists of approximately 40 long-period OBSs built by Geomar GmbH by adding an external moving arm that drops the 3-C geophone on the seafloor to their Ocean-Bottom Hydrophone (OBH). The OBH was originally designed as a ocean-bottom buoy, tall enough to be moved by water currents, which did not matter for the hydrophone, but does for the geophones. The University of Cambridge, UK, has developed a 4-C short-period, short-deployment (20 days) OBS of which 25 or more have been built. It resembles the Geomar design—taller than it is wide, and with a external arm that drops the geophones so that they are not shaken by water currents together with the rest of the tall instrument. Dalhousie University

---

7 http://www.seg.org/museum/VM/pict0432.html
8 http://www.obsip.org
from Canada has a small number of short-period 4-C OBSs developed in-house. University of Durham, UK, has six in-house built short-period OBSs and the Monterey Bay Aquarium Research Institute has a few ROV-operated OBSs developed in-house too.

With the exception of the Japanese OBS, among the seismometers described above only the broadbands (Guralps and the long-deployment OBSIP ones) have both 4-C recording capabilities as well as reliability and recoverability demonstrated on a large pool of instruments. However, they are very expensive. The OBSIP short-period instruments are cheap (approx. $20k) and reliable (> 99%), but are only 2-C (vertical geophone and hydrophone). The UTIG ones seem less rugged and reliable than the Japanese OBSs, with instrument loss rates such as 3 instruments out of 33 (1997 Iberia experiment). The German short-period OBSs exhibit a lower recovery rate than the Japanese ones and, like the Cambridge ones, have external moving parts in order to correct for a design-induced problem. Other instruments (Dalhousie, Durham) look promising but they are larger and more complex than the Japanese OBSs, and very few of them have been built, so that extensive field testing (hundreds of individual deployments) has not occurred. Should the seismic industry need an existing deep water 4-C receiver technology solution, the Japanese OBS seems to be the most appropriate among the instrument designs known to the author.

A comparison between OBCs and Japanese OBSs yields interesting results. They share many advantages. Both can be used for data acquisition in areas with obstacles such as platforms, reefs, and transition zones that do not allow for streamer operation. Both record 4-C data, which can be used to image through gas clouds and in anisotropic media, to map fractures, and to predict overpressure. Neither is sensitive to noise: the seafloor is quiet and as a bonus the upward going signal and downward going noise (including the receiver ghost) can be separated by processing in the manner developed by Barr and Sanders (1989).

It is in the disadvantages areas that the comparison shows dissimilarities. OBCs are currently limited to water depths of 500 m, with the most rugged going down to 1500 m. OBSs in contrast typically have a depth limitation of 6000 m. Whereas the cables sometimes have problems with withstanding deepwater pressures for extended periods of time, the OBSs have no problem. While OBCs suffer damage by strain from repeated retrievals due to repairs and moving surveys, the small and rigid OBSs are not harmed at retrieval. A localized event that destroys part of an OBC (submarine landslide, shark bite) can affect the entire cable, but OBSs are totally independent from each other. OBCs sometimes need burial by ROV for good coupling, which increases the costs and troubles at retrieval, while the weight of an OBS is sufficient to couple it well. But OBSs have disadvantages too, the first being a higher cost per deployed receiver. This may be alleviated by economy of scale if more units are produced, and by the longer life of a OBS as compared to a OBC. Another issue is that of drift during sinking, especially in deep water. The position can be determined satisfactorily with sonar arrays, but for industry purposes the deployment precision needs to be increased.
CONCLUSIONS

OBSs are well-tested, functional tools commonly used in crustal seismology. The cost per deployed receiver is higher than that of OBCs and there are still issues to be solved regarding the accuracy of landings on the sea bottom, but they can be deployed much deeper and are more robust than OBCs. They have potential for improvement, and can be the tool of choice for 4-C surveys at depths greater than 1500m.

ACKNOWLEDGMENTS

I would like to thank Bryan Kerr for useful information, for feedback, and for his comradeship during the cruise; to Simon Klemperer for logistical support and for reviewing this paper; to Shuichi Kodaira and Narumi Takahashi for arranging the practical details of our cruise, before and during it; to JAMSTEC’s technician team, especially to Kazumi Baba (Fig 4), chief technician on R/V Kaiyo, for the great cooperation during our work, and for their patience and understanding of people of a completely different cultural background, such as Bryan and me; and to the National Science Foundation and the Stanford Graduate Fellowships program for funding.

REFERENCES


R/V Kaiyo
Dynamic permeability in poroelasticity

James G. Berryman

ABSTRACT
The concept of dynamic permeability is reviewed. Modeling of seismic wave propagation using dynamic permeability is important for analyzing data as a function of frequency. In those systems where the intrinsic attenuation of the wave is caused in large part by viscous losses due to the presence of fluids, the dynamic permeability provides a very convenient and surprisingly universal model of this behavior.

INTRODUCTION
This article will review some of what is known about the concept of “dynamic permeability,” as it is now commonly called, within the subject of poroelasticity. There are competing terms that are used sometimes. In particular, Biot (1956b) introduced the concept of dynamic viscosity, and this concept is incorporated into the dynamic permeability concept as we shall see. Dynamic tortuosity is closely related to dynamic permeability since the electric tortuosity is related (inversely) to the high frequency limit of the dynamic permeability, whereas the usual Darcy permeability is exactly the low frequency limit of the dynamic permeability. The tortuosity is also related to the effective mass of the fluid in the presence of the solid frame or to the effective mass of the solid in the presence of the pore fluid. We will discuss all of these concepts at greater length in the following sections.

EQUATIONS OF POROELASTICITY
For long-wavelength disturbances (\( \lambda >> h \), where \( h \) is a typical pore size) propagating through such a porous medium, we define average values of the (local) displacements in the solid and also in the saturating fluid. The average displacement vector for the solid frame is \( \mathbf{u} \) while that for the pore fluid is \( \mathbf{u}_f \). The average displacement of the fluid relative to the frame is \( \mathbf{w} = \phi(\mathbf{u} - \mathbf{u}_f) \), where \( \phi \) is the porosity. For small strains, the frame dilatation is
\[
e = e_x + e_y + e_z = \nabla \cdot \mathbf{u},
\]
where \( e_x, e_y, e_z \) are the Cartesian strain components. Similarly, the average fluid dilatation is
\[
e_f = \nabla \cdot \mathbf{u}_f
\]
(\epsilon_f \text{ also includes flow terms as well as dilatation}) and the increment of fluid content is defined by

\[ \zeta = -\nabla \cdot \mathbf{w} = \phi (\epsilon - \epsilon_f). \]  

With these definitions, Biot (1962) obtains the stress-strain relations in the form

\[ \delta \tau_{xx} = He - 2\mu (\epsilon_z + e_z) - C \zeta, \]  

and similarly for \( \delta \tau_{yy}, \delta \tau_{zz}, \)

\[ \delta \tau_{zx} = \mu \left( \frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x} \right), \]  

and again similarly for \( \delta \tau_{yz}, \delta \tau_{xy}, \) and finally

\[ \delta p_f = M \zeta - Ce. \]  

The \( \delta \tau_{ij} \) are deviations from equilibrium of average Cartesian stresses in the saturated porous material and \( \delta p_f \) is similarly the isotropic pressure deviation in the pore fluid.

With time dependence of the form \( \exp(-i\omega t) \), the coupled wave equations that incorporate (4)-(6) are of the form

\[ -\omega^2 (\rho \mathbf{u} + \rho_f \mathbf{w}) = (H - \mu) \nabla \epsilon + \mu \nabla^2 \mathbf{u} - C \nabla \zeta, \]  

\[ -\omega^2 (\rho_f \mathbf{u} + q \mathbf{w}) = -M \nabla \zeta + C \nabla \epsilon, \]  

where \( \rho = \phi \rho_f + (1 - \phi) \rho_s \), with \( \rho_f \) being the fluid density, \( \rho_s \) being the solid density, and \( \rho \) being the overall bulk-density of the material, while

\[ q(\omega) = \rho_f \left[ \alpha/\phi + i F(\xi) \eta/\kappa_0 \omega \right] \]  

is the frequency dependent effective density of the fluid in relative motion. This expression is also the one that we study later in order to introduce the concepts of dynamic permeability and tortuosity (Johnson et al., 1987). So we want to emphasize now how important this equation for \( q(\omega) \) will be to our later analyses. The kinematic viscosity of the liquid is \( \eta \); the low frequency permeability of the porous frame is \( \kappa_0 \); the 3D dynamic viscosity factor [as it was first called by Biot (1956b)] is given, for our choice of sign for the frequency dependence, by \( F(\xi) = \frac{1}{4} \left\{ \xi T(\xi)/[1 + 2T(\xi)/i\xi] \right\} \) where \( T(\xi) = \text{ber}'(\xi) - i\text{bei}'(\xi) \) and \( \xi = (\omega h^2/\eta) \frac{1}{2} \). The functions \( \text{ber}(\xi) \) and \( \text{bei}(\xi) \) are the real and imaginary parts of the Kelvin function. The dynamic parameter \( h \) is a characteristic length generally associated with and comparable in magnitude to the steady-flow hydraulic radius. (For a model calculation we discuss later, \( h = a \), the radius of a cylindrical pore.) The tortuosity \( \alpha \geq 1 \) is a pure number related to the frame inertia which has been measured by Johnson et al. (1982) and has also been estimated theoretically by Berryman (1980). We discuss \( \alpha \) and its various interpretations in greater detail later in this review.
The coefficients $H$, $C$, and $M$ are given (Gassmann, 1951; Biot and Willis, 1957; Biot, 1962) by

$$H = K + \frac{4}{3} \mu + \frac{(K_s - K)^2}{(D - K)}, \quad (10)$$

$$C = K_s(K_s - K)/(D - K), \quad (11)$$

and

$$M = K_s^2/(D - K), \quad (12)$$

where

$$D = K_s[1 + \phi(K_s/K_f - 1)], \quad (13)$$

with $K_f$ being the fluid bulk modulus and $K_s$ being the solid bulk modulus. The frame (porous solid without liquid in the pores) constants are $K$ for bulk and $\mu$ for shear. Equations (10)-(13) are correct as long as the porous material may be considered homogeneous on the microscopic scale as well as the macroscopic scale.

Eq. (7) is essentially the equation of elastodynamics of the solid frame with coupling terms (involving $w$ and $\zeta$) to the fluid motion. Eq. (8) reduces exactly to Darcy’s equation when the solid displacement $u$ and frame strain $e$ are zero, since the right hand side of the equation is just $-\nabla p_f$.

To decouple (and subsequently solve) the wave equations in (7) and (8) into Helmholtz equations for the three modes of propagation, note that the displacements $u$ and $w$ can be decomposed as

$$u = \nabla \Upsilon + \nabla \times \beta, \quad w = \nabla \psi + \nabla \times \chi, \quad (14)$$

where $\Upsilon$, $\psi$ are scalar potentials and $\beta$, $\chi$ are vector potentials. Substituting (14) into (7) and (8), the two equations are solved if two pairs of equations are satisfied:

$$(\nabla^2 + k_s^2)\beta = 0, \quad \chi = -\rho_f \beta/q \quad (15)$$

and

$$(\nabla^2 + k_{\pm}^2)A_{\pm} = 0. \quad (16)$$

The wavenumbers in (15) and (16) are defined by

$$k_s^2 = \omega^2(\rho - \rho_f^2/q)/\mu \quad (17)$$

and

$$k_{\pm}^2 = \frac{1}{2} \left[ b + f \mp \sqrt{(b - f)^2 + 4cd} \right]^2, \quad (18)$$
\[ b = \omega^2 (\rho M - \rho_f C) / \Delta, \quad c = \omega^2 (\rho_f M - q C) / \Delta, \]
\[ d = \omega^2 (\rho_f H - \rho C) / \Delta, \quad f = \omega^2 (q H - \rho_f C) / \Delta, \]

(19)

with \( \Delta = MH - C^2 \). The linear combination of scalar potentials has been chosen to be \( A_\pm = \Gamma_\pm Y + \psi \), where

\[ \Gamma_\pm = d / (k^2_\pm - b) = (k^2_\pm - f) / c. \]

(20)

With the identification (20), the decoupling is complete.

Equations (15) and (16) are valid for any choice of coordinate system, not just Cartesian coordinates, and they are therefore very useful in all applications of the theory.

**INDUCED MASS EFFECT**

Biot (1956a) defines the kinetic energy \( T \) of the porous medium by

\[ 2T = \rho_{11} \dot{u} \cdot \dot{u} + 2\rho_{12} \dot{u} \cdot \dot{u}_f + \rho_{22} \dot{u}_f \cdot \dot{u}_f, \]

(21)

where \( \dot{u} \) and \( \dot{u}_f \) are the solid and fluid velocities at a point in the medium. In the standard way, the overdots indicate a time derivative. Then he shows that the inertial coefficients satisfy two sum rules:

\[ \rho_{11} + \rho_{12} = (1 - \phi) - \rho_s \]

(22)

and

\[ \rho_{22} + \rho_{12} = \phi \rho_f. \]

(23)

Furthermore, as a matter of definition for the “structure factor” \( \alpha \), we also have

\[ \rho_{12} = -(\alpha - 1) \phi \rho_f. \]

(24)

And Biot’s discussion makes it clear, furthermore, that \( -\rho_{12} \) should be thought of as the added or induced mass of the solid when it oscillates in the presence of the fluid. So we have

\[ \rho_{11} = (1 - \phi) - \rho_s - \rho_{12} \equiv (1 - \phi)(\rho_s + r \rho_f), \]

(25)

where \( r \) is a measure of the geometry of the solid. In particular, it is well-known [see Lamb (1936)] that the factor \( r = \frac{1}{2} \) for well-separated spheres and that it can vary from zero to unity, depending on the shape for simple objects such as spheroids. Combining these results, we find in general that

\[ \alpha = 1 + r(\phi^{-1} - 1), \]

(26)

or, if we take a conservative value for \( r \) and set it to \( r = \frac{1}{2} \), then we have

\[ \alpha = \frac{1}{2} (1 + \phi^{-1}). \]

(27)

The formula (27) was derived by Berryman (1980, 1981), and it is frequently used with success when fitting real data.
ANOTHER INTERPRETATION OF $\alpha$ AS TORTUOSITY

It has been shown by Berryman (1980, 1981) that the speed of the slow wave \textit{i.e.}, the second compressional wave predicted by Biot (1956a) can be written as

$$v_2^2 = \frac{K_f}{\alpha \rho_f},$$

where $K_f$ is the bulk modulus of the pore fluid. Or, first noting that the sound speed in the fluid is given by $v_f = \sqrt{K_f/\rho_f}$, we also have

$$v_2 = \frac{v_f}{\sqrt{\alpha}}.$$ (29)

Thus, since the slow wave is a compressional wave through the pore space which presumably travels at approximately the speed $v_f$ locally, the formula (29) can be interpreted as showing that the path length between two points in the pore space a distance $L$ apart in space are a distance $\sqrt{\alpha}L$ apart when the travel path is restricted to the pore space. Thus, $\sqrt{\alpha}$ is a measure of the tortuosity of the pore space.

That $\alpha$ is a measure of the tortuosity has also been shown using a very different argument by Brown (1980). He shows that $\alpha$ is related to the electric formation factor for a porous system. In particular $\alpha = \phi F$, where $F$ is the electrical formation factor \textit{[which should not be confused with the dynamic viscosity factor $F(\xi)$ to be introduced later]} defined by $F = g_f/g \geq 1$, with $g_f$ being the electrical conductivity of the pore fluid and $g$ being the effective overall conductivity of the system when the solid is nonconducting. The formation factor and the tortuosity are both equal to unity when $\phi = 1$. For some systems, it is known both from Archie’s law and from theory too that $F \simeq \phi^{-3/2}$ is often a reasonable estimate for pore systems. Thus, $\alpha \simeq \phi^{-1/2}$. Both this expression and (27) have the same representation as $\alpha \simeq (3-\phi)/2$ when $\phi$ is large, showing that the two approaches using electrical tortuosity and the induced mass do in fact give similar results in this limit, even though it might seem hard to understand physically just exactly why this should be true.

DYNAMIC PERMEABILITY AND TORTUOSITY

Now we will return to the expression (9) and make the following two definitions: First, the dynamic tortuosity is given by

$$\alpha(\omega) \equiv \frac{\phi q(\omega)}{\rho_f} = \alpha + i F(\xi) \eta \phi / \kappa_0 \omega$$

and, second, the inverse of the dynamic permeability is given by

$$\frac{1}{\kappa(\omega)} \equiv \frac{\omega q(\omega)}{i \eta \rho_f}.$$ (31)

This second expression can also be rewritten as

$$\kappa(\omega) = \frac{\kappa_0}{F(\xi) - i \kappa_0 \alpha \omega / \eta \phi}.$$ (32)
The motivations for both definitions can be seen in their limiting values:

\[ \alpha(\infty) = \alpha \quad \text{and} \quad \kappa(0) = \kappa_0. \]  (33)

These results follow once it is recognized that the dynamic viscosity factor \( F(\xi) \) has unity as its the limit when \( \omega \to 0 \) and, although it can have differing numerical factors, it always goes like \((-i \omega/\omega_0)^{1/2}\) for large \( \omega \to \infty \). We will give some examples of this behavior in the next section.

**ANALYTICAL EXAMPLES OF THE DYNAMIC VISCOSITY FUNCTION**

**Solid spheres**

Chase (1979), using some results from Landau and Lifshitz (1959), shows that, for well-separated spheres of radius \( R \), the factor

\[ q(\omega) = \frac{\rho_f}{\phi} [1 + i \Delta(\omega)], \]  (34)

where

\[ \Delta(\omega) = \frac{9}{4}(\phi^{-1} - 1) \left[ 1 + z - i z(1 + 2z/9) \right] z^{-2} \]  (35)

with \( z = 2^{-1/2} \xi \). Here \( \xi = (\omega R^2/\eta)^{1/2} \), and \( \eta \) is the viscosity of the fluid, as usual. Comparing the main terms, we find that this formula shows

\[ \alpha = 1 + \frac{1}{2}(\phi^{-1} - 1), \]  (36)

which is in complete agreement with (27), and

\[ F(\xi) = 1 + (-i)^{1/2} \xi. \]  (37)

The result (37) is the first of several results showing that \( F \) behaves like \( \omega^{1/2} \) for large \( \omega \).

**2D duct**

Biot (1956b) shows that for a 2D duct, \( i.e., \) fluid between two planes separated by a distance \( 2a_1 \), \( F(\xi) \to 1 \) as \( \omega \to 0 \) and

\[ F(\xi) \to (-i)^{1/2} \frac{\xi}{3} \]  (38)

as \( \omega \to \infty \). Here \( \xi = (\omega a_1^2/\eta)^{1/2} \) with \( a_1 \) being half the separation distance between the planes.
Dynamic permeability

3D duct

Biot (1956b) also shows that for a 3D circular cylinder duct, i.e., fluid inside a circular cylindrical pore of radius \( a \), \( F(\xi) \to 1 \) as \( \omega \to 0 \) and

\[
F(\xi) \to (-i)^{1/2} \frac{\xi}{4}
\]

as \( \omega \to \infty \). Here \( \xi = (\omega a^2/\eta)^{1/2} \), with \( a \) being the cylinder radius.

Discussion of analytical examples

All three of these examples, and indeed any other example as well, show that \( F(\xi) \to 1 \) as \( \omega \to 0 \). This result is universal. The other limit for \( \omega \to \infty \) gives somewhat different results for the three cases considered, but they can all be approximated by using the form

\[
F(\xi) \simeq (1 - i P \xi^2)^{1/2},
\]

where \( P \) is a number that depends on the duct model. Assuming for the moment that \( R = a_1 = a \): For spherical particles, \( P = 1 \). For the 2D duct, \( P = 1/9 \). For the 3D duct, \( P = 1/16 \).

Since one of the implicit goals of the dynamic permeability analysis is to determine a universal form for the function \( F(\xi) \) and thereby determine a universal form for the dynamic permeability, it is important to consider how these problems differ from each other. The radius \( R \) is a measure of the particle size, but this size is not easy to relate to the radius of the cylindrical pore in the third case, or to the duct height in the second case. It would make more sense to relate these quantities in some more general way since the typical rock sample will not have any of these geometries. Perhaps the obvious choice is to use \( \kappa_0 \) itself, since we need a pertinent measure of length squared, and that is exactly what the low frequency permeability is.

In fact, if we first consider the form of (37) and take the point of view that the factor \( \kappa_0 \alpha \omega/\eta \phi \) in the denominator determines a "natural" characteristic frequency for the problem given by

\[
\omega_0 = \eta \phi / \alpha \kappa_0,
\]

then, we can choose to approximate the dynamic viscosity factor by

\[
F(\xi) \simeq (1 - i P \xi^2)^{1/2}
\]

where now \( \xi \equiv (\omega/\omega_0)^{1/2} \) and \( P \) is a real numerical factor that is at least approximately problem independent. Experimental and computational results show (Sheng and Zhou, 1988; Johnson, 1989; Sheng et al., 1989) that many rocks and other porous systems can be successfully modeled this way using \( P \)'s such that \( 0.4 \leq P \leq 0.5 \). So the range of values for \( P \) is really quite small in many cases.

There are some exceptions to these rules but they are beyond our present scope, so the reader is encouraged to see the paper by Pride et al. (1993) for an extended discussion.
NUMERICAL EXAMPLES OF THE DYNAMIC PERMEABILITY FUNCTION

From (32) and the preceding discussion, we conclude that a reasonable choice of the functional form for dynamic permeability is

\[
\frac{\kappa(\omega)}{\kappa_0} = \frac{1}{(1 - i P \omega/\omega_0)^{1/2} - i \omega/\omega_0},
\]

(43)

where \(0 \leq P \leq 1\) and \(\omega_0 = \eta \phi / \alpha \kappa_0\). We plot this function in the complex plane for five choices of \(P\) in Figure 1. The polar angle \(\theta\) is displayed as a function of \(\omega/\omega_0\) in Figure 2. Then, the real and imaginary parts are plotted in Figures 3 and 4 as a function of the quantity \(\omega/\omega_0\).

Figure 1 shows that all these choices give very similar behavior in the complex plane. The function looks much like a semi-circle centered at the point \((1/2, 0)\) and having radius \(1/2\) in all cases. But this observation is only exactly true for the case \(P = 0\). For that case it is also true that the relationship between \(\omega\) and the polar angle \(\theta\) in the complex plane is given exactly by \(\omega/\omega_0 = \tan(\theta/2)\). For the other values of \(P\), this relationship holds approximately true for \(\omega/\omega_0 \leq 1\) and \(P \leq 1/2\), but deviations become substantial in all cases for higher values of \(\omega\), as is observed in Figure 2.

Figure 3 shows that the real part of the dynamic permeability is essentially flat for a wide range of the smaller frequencies, but then the function falls off rapidly as the frequency gets close to the resonance frequency \(\omega_0\). Figure 4 shows that all the action occurs over six decades of frequency and the main region of the deviation (half maximum and above in the imaginary part) lies approximately in the range \(0.2 \leq \omega/\omega_0 \leq 3.0\), which is just slightly over one decade in width.
Figure 2: Polar angle $\theta$ in degrees in the complex plane of points of five model functions of the form found in Eq. (43) for the different values of $P = 0, 0.25, 0.4, 0.5, 1$ (see Figure 1). Note that the case $P = 0$ is exactly a semi-circle of radius $1/2$ centered at the point $(1/2, 0)$ in the complex plane, and for this case $\omega/\omega_0 = \tan(\theta/2)$. All other cases are observed to deviate from this behavior.

Figure 3: Illustration of the behavior of the real part of five model functions for different values of $P = 0, 0.25, 0.4, 0.5, 1$ as a function of the argument $\omega/\omega_0$. Note that the real part of the dynamic permeability is essentially flat for a wide range of the smaller frequencies, but then falls off rapidly as the frequency approaches the resonance frequency $\omega_0$ from below.

Figure 4: Illustration of the behavior of the imaginary part of five model functions for different values of $P = 0, 0.25, 0.4, 0.5, 1$ as a function of the argument $\omega/\omega_0$. Note the main region of the deviation (half maximum and above) lies approximately in the range $0.2 \leq \omega/\omega_0 \leq 3.0$, which is just slightly over one decade in width.
CONCLUSIONS

The dynamic permeability is a useful concept that is important for analyzing seismic data as a function of frequency. In those systems where the intrinsic attenuation of the wave is caused in large part by viscous losses due to the presence of fluids, the dynamic permeability provides a convenient model of this behavior. Perhaps surprising is the fact that most systems can be modeled with a nearly universal function, depending only weakly on a scalar parameter $P$ whose range of variation has been shown to be quite small ($0.4 \leq P \leq 0.5$) for a very wide range of systems relevant to seismic exploration.

ACKNOWLEDGMENTS

These notes were generated to supplement a lecture given at University of California – Berkeley. Questions from Professor Chi-Yuen Wang and other members of the audience played a significant role in the choice of topics presented here.

REFERENCES


Poroelastic shear modulus dependence on pore-fluid properties arising in a model of thin isotropic layers

James G. Berryman

ABSTRACT

Gassmann’s fluid substitution formulas for bulk and shear moduli were originally derived for the quasi-static mechanical behavior of fluid-saturated rocks. It has been shown recently that it is possible to understand deviations from Gassmann’s results at higher frequencies when the rock is heterogeneous, and in particular when the rock heterogeneity anywhere is locally anisotropic. On the other hand, a well-known way of generating anisotropy in the earth is through fine (compared to wavelength) layering. Then, Backus’ averaging of the mechanical behavior of the layered isotropic media at the microscopic level produces anisotropic mechanical and seismic behavior at the macroscopic level. For our present purposes, the Backus averaging concept can also be applied to fluid-saturated porous media, and thereby permits us to study how and what deviations from Gassmann’s predictions could arise in an elementary fashion. We consider both closed-pore and open-pore boundary conditions between layers within this model in order to study in detail how violations of Gassmann’s predictions can arise. After evaluating a number of possibilities, we determine that energy estimates show unambiguously that one of our possible choices – namely, $G_{\text{eff}}^{(2)} = \frac{(C_{11} + C_{33} - 2C_{13} - C_{66})}{3}$ – is the correct one for our purposes. This choice also possesses the very interesting property that it is one of two sets of choices satisfying a product formula $6k_{V}G_{\text{eff}}^{(1)} = 6k_{R}G_{\text{eff}}^{(2)} = \omega_{+}\omega_{-}$, where $\omega_{\pm}$ are eigenvalues of the stiffness matrix for the pertinent quasi-compressional and quasi-shear modes. $k_{R}$ is the Reuss average for the bulk modulus, which is also the true bulk modulus $K$ for the simple layered system. $k_{V}$ is the Voigt average. For a polycrystalline system composed at the microscale of simple layered systems randomly oriented in space, $k_{V}$ and $k_{R}$ are the upper and lower bounds respectively on the bulk modulus, and $G_{\text{eff}}^{(2)}$ and $G_{\text{eff}}^{(1)}$ are the upper and lower bounds respectively on the $G_{\text{eff}}$ of interest here. We find that $G_{\text{eff}}^{(2)}$ exhibits the expected/desired behavior, being dependent on the fluctuations in the layer shear moduli and also being a monotonically increasing function of Skempton’s coefficient $B$ of pore-pressure buildup, which is itself a measure of the pore fluid’s ability to stiffen the porous material in compression.
INTRODUCTION

It has been shown recently (Berryman and Wang, 2001) that local anisotropy in heterogeneous porous media must play a significant role in deviations from the well-known fluid substitution formulas of Gassmann (Gassmann, 1951; Berryman, 1999). In particular, it is not easy to see in an explicit way how fluid dependence of the effective shear modulus $G_{\text{eff}}$ of such systems can arise within a Gassmann-like derivation. Even though such effects have been observed in experimental data (Berryman et al., 2002a,b), locally isotropic materials have been shown to be incapable of producing such results. So a simple theoretical means of introducing anisotropy into such a poroelastic system is desirable in order to aid our physical intuition about these problems.

When viewed from a point close to the surface of the Earth, the structure of the Earth is often idealized as being that of a layered (or laminated) medium with essentially homogeneous physical properties within each layer. Such an idealization has a long history and is well represented by famous textbooks such as Ewing et al. (1957), Brekhovskikh (1980), and White (1983). The importance of anisotropy due to fine layering (i.e., layer thicknesses small compared to the wavelength of the seismic or other waves used to probe the Earth) has been realized more recently, but efforts in this area are also well represented in the literature by the work of Postma (1955), Backus (1962), Berryman (1979), Schoenberg and Muir (1987), Anderson (1989), Katsube and Wu (1998), and many others.

In a completely different context, because of the relative ease with which their effective properties may be computed, finely layered composite laminates have been used for theoretical purposes to construct idealized but, in principle, realizable materials to test the optimality of various rigorous bounds on the effective properties of general composites. This line of research includes the work of Tartar (1976), Schulgasser (1977), Tartar (1985), Francfort and Murat (1986), Kohn and Milton (1986), Lurie and Cherkaev (1986), Milton (1986), Avellaneda (1987), Milton (1990), deBotton and Castañeda (1992), and Zhikov et al. (1994), among others. Recent books on composites by Cherkaev (2000), Milton (2002), and Torquato (2002) also make frequent use of these ideas.

In this work, we will study some simple means of estimating the effects of fluids on elastic and poroelastic constants and, in particular, we will derive formulas for anisotropic poroelastic media using a straightforward generalization of the method of Backus (1962) originally formulated for determining the effective constants of a laminated elastic material. There has been some prior work in this area for poroelastic problems by Norris (1993), Gurevich and Lopatnikov (1995), Gelinsky et al. (1998), and others. However, our focus is specific to the issue of shear modulus dependence on pore fluids [see Mavko and Jizba (1991) and Berryman et al. (2002b)]. We initially review facts about elastic layered systems and then show that, of all the possible candidates for an effective shear modulus exhibiting mechanical dependence on pore fluids, the evidence shows that one choice is unambiguously preferred. We then use Backus averaging to obtain an explicit formula for this shear modulus in terms of layer elastic parameters. The results agree with prior physical arguments indicating that fluid presence stiffens the medium in shear, but the layered material needs substantial inhomogeneity in its shear properties for the effect to be observed. An Appendix provides a simple derivation of
NOTATION FOR ELASTIC ANALYSIS

In tensor notation, the relationship between components of stress $\sigma_{ij}$ and strain $u_{k,l}$ is given by

$$\sigma_{ij} = C_{ijkl} u_{k,l},$$

where $C_{ijkl}$ is the stiffness tensor, and repeated indices on the right hand side of (3) are summed. In (1), $u_k$ is the $k$th Cartesian component of the displacement vector $\mathbf{u}$, and $u_{k,l} = \partial u_k / \partial x_l$. Whereas for an isotropic elastic medium the stiffness tensor has the form

$$C_{ijkl} = \lambda \delta_{ij} \delta_{kl} + \mu (\delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk}),$$

depending on only two parameters (the Lamé constants, $\lambda$ and $\mu$), this tensor can have up to 21 independent constants for general anisotropic elastic media. The stiffness tensor has pairwise symmetry in its indices such that $C_{ijkl} = C_{jikl} = C_{ijlk}$, which will be used later to simplify the resulting equations.

The general equation of motion for elastic wave propagation through an anisotropic medium is given by

$$\rho \ddot{u}_i = \sigma_{ij,j} = C_{ijkl} u_{k,l},$$

where $\ddot{u}_i$ is the second time derivative of the $i$th Cartesian component of the displacement vector $\mathbf{u}$ and $\rho$ is the density (assumed constant). Equation (3) is a statement that the product of mass times acceleration of a particle is determined by the internal stress force $\sigma_{ij,j}$. For the present purposes, we are more interested in the quasistatic limit of this equation, in which case the left-hand side of (3) vanishes and the equation to be satisfied is just the force equilibrium equation

$$\sigma_{ij,j} = 0.$$  \hfill (4)

A commonly used simplification of the notation for elastic analysis is given by introducing the strain tensor, where

$$e_{ij} = \frac{1}{2} (u_{i,j} + u_{j,i}) = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right).$$  \hfill (5)

Then, using one version of the Voigt convention, in which the pairwise symmetries of the stiffness tensor indices are used to reduce the number of indices from 4 to 2 using the rules $11 \rightarrow 1, 22 \rightarrow 2, 33 \rightarrow 3, 23 \rightarrow 4, 13 \rightarrow 5, 12 \rightarrow 6$, we have

$$\begin{pmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{31} \\ \sigma_{12} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & C_{13} \\ C_{12} & C_{22} & C_{23} \\ C_{13} & C_{23} & C_{33} \end{pmatrix} \begin{pmatrix} e_{11} \\ e_{22} \\ e_{33} \end{pmatrix} = \begin{pmatrix} 2C_{44} \\ 2C_{55} \\ 2C_{66} \end{pmatrix}.$$  \hfill (6)
Although the Voigt convention introduces no restrictions on the stiffness tensor, we have chosen to limit discussion to the form in (6), which is not completely general. Of the 36 coefficients (of which 21 are generally independent), we choose to treat only those cases for which the 12 coefficients shown (of which nine are generally independent) are nonzero. This form includes all orthorhombic, cubic, hexagonal, and isotropic systems, while excluding triclinic, monoclinic, trigonal, and some tetragonal systems, since each of the latter contains additional off-diagonal constants that may be nonzero. Nevertheless, we will restrict our discussion to (6) or to the still simpler case of transversely isotropic (TI) materials.

For TI materials whose symmetry axis is in the \( x_3 \) direction, another common choice of notation is \( C_{11} = C_{22} \equiv a, C_{12} \equiv b, C_{13} = C_{23} \equiv f, C_{33} \equiv c, C_{44} = C_{55} \equiv l, \) and \( C_{66} \equiv m \). There is also one further constraint on the constants that \( a = b + 2m \), following from rotational symmetry in the \( x_1 x_2 \)-plane. In such materials, (6) may be replaced by

\[
\begin{pmatrix}
\sigma_{11} \\
\sigma_{22} \\
\sigma_{33} \\
\sigma_{23} \\
\sigma_{31} \\
\sigma_{12}
\end{pmatrix} = \begin{pmatrix}
a & b & f \\
b & a & f \\
f & f & c
\end{pmatrix} \begin{pmatrix}
e_{11} \\
e_{22} \\
e_{33} \\
e_{23} \\
e_{31} \\
e_{12}
\end{pmatrix},
\]

in which the matrix has the same symmetry as hexagonal systems and of which isotropic symmetry is a special case (having \( a = c = \lambda + 2\mu, b = f = \lambda, \) and \( l = m = \mu \)).

**BACKUS AVERAGING OF FINE ELASTIC LAYERS**

Backus (1962) presents an elegant method of producing the effective constants for a finely layered medium composed of either isotropic or anisotropic elastic layers. For simplicity, we will assume that the layers are isotropic, in which case the equation relating elastic stresses \( \sigma_{ij} \) to elastic strains \( e_{ij} \) is given by

\[
\begin{pmatrix}
\sigma_{11} \\
\sigma_{22} \\
\sigma_{33} \\
\sigma_{23} \\
\sigma_{31} \\
\sigma_{12}
\end{pmatrix} = \begin{pmatrix}
\lambda + 2\mu & \lambda & \lambda \\
\lambda & \lambda + 2\mu & \lambda \\
\lambda & \lambda & \lambda + 2\mu
\end{pmatrix} \begin{pmatrix}
e_{11} \\
e_{22} \\
e_{33} \\
e_{23} \\
e_{31} \\
e_{12}
\end{pmatrix}.
\]

The key idea presented by Backus is that these equations can be rearranged into a form where rapidly varying coefficients multiply slowly varying stresses or strains. For simple layering, we know physically (and can easily prove mathematically) that the normal stress and the tangential stresses must be continuous at the boundaries between layers. If the layering direction is the \( z \) or \( x_3 \) direction as is the normal choice in the acoustics and geophysics literature, then \( \sigma_{33}, \sigma_{23}, \sigma_{31}, e_{11}, e_{22}, \) and \( e_{12} \) are continuous and in fact constant throughout such a laminated material. If the constancy of \( e_{11}, e_{22}, \) and \( e_{12} \) were not so, the layers would necessarily experience relative slip; while if the constancy of \( \sigma_{33}, \sigma_{23}, \) and \( \sigma_{31} \) were not so, then there would...
be force gradients across boundaries necessarily resulting in nonstatic material response to the lack of force equilibrium.

By making use of this elegant idea, we arrive at the following equation

\[
\begin{pmatrix}
\sigma_{11} \\
\sigma_{22} \\
-e_{33} \\
e_{23} \\
e_{31} \\
\sigma_{12}
\end{pmatrix} = \begin{pmatrix}
\frac{4\mu(\lambda+\mu)}{\lambda+2\mu} & \frac{2\mu}{\lambda+2\mu} & \frac{\lambda}{\lambda+2\mu} \\
\frac{2\mu}{\lambda+2\mu} & \frac{4\lambda+2\mu}{\lambda+2\mu} & \frac{\lambda}{\lambda+2\mu} \\
\frac{\lambda}{\lambda+2\mu} & \frac{\lambda}{\lambda+2\mu} & \frac{1}{\lambda+2\mu} \\
\frac{1}{2\mu} & \frac{1}{2\mu} & 2\mu
\end{pmatrix} \begin{pmatrix}
e_{11} \\
e_{22} \\
e_{33} \\
e_{23} \\
e_{31} \\
e_{12}
\end{pmatrix},
\]

(9)

which can be averaged essentially by inspection. Equation (9) can be viewed as a Legendre transform of the original equation, to a different set of dependent/independent variables in which both vectors have components with mixed physical significance, some being stresses and some being strains. Otherwise these equations are completely equivalent to the original ones in (8).

Performing the layer average using the symbol \(\langle \cdot \rangle\), assuming as mentioned previously that the variation is along the \(z\) or \(x_3\) direction, we find, using the notation of (7),

\[
\begin{pmatrix}
\langle \sigma_{11} \rangle \\
\langle \sigma_{22} \rangle \\
-\langle e_{33} \rangle \\
\langle e_{23} \rangle \\
\langle e_{31} \rangle \\
\langle \sigma_{12} \rangle
\end{pmatrix} = \begin{pmatrix}
\frac{4\mu(\lambda+\mu)}{\lambda+2\mu} & \frac{2\mu}{\lambda+2\mu} & \frac{\lambda}{\lambda+2\mu} \\
\frac{2\mu}{\lambda+2\mu} & \frac{4\lambda+2\mu}{\lambda+2\mu} & \frac{\lambda}{\lambda+2\mu} \\
\frac{\lambda}{\lambda+2\mu} & \frac{\lambda}{\lambda+2\mu} & \frac{1}{\lambda+2\mu} \\
\frac{1}{2\mu} & \frac{1}{2\mu} & 2\mu
\end{pmatrix} \begin{pmatrix}
e_{11} \\
e_{22} \\
e_{33} \\
e_{23} \\
e_{31} \\
e_{12}
\end{pmatrix},
\]

(10)

which can then be solved to yield the expressions

\[
a = \left(\frac{\lambda}{\lambda+2\mu}\right)^2 \left(\frac{1}{\lambda+2\mu}\right)^{-1} + 4 \left(\frac{\mu(\lambda+\mu)}{\lambda+2\mu}\right),
\]

(11)

\[
b = \left(\frac{\lambda}{\lambda+2\mu}\right)^2 \left(\frac{1}{\lambda+2\mu}\right)^{-1} + 2 \left(\frac{\lambda\mu}{\lambda+2\mu}\right),
\]

(12)

\[
c = \left(\frac{1}{\lambda+2\mu}\right)^{-1}
\]

(13)
Equations (11)–(16) are the well-known results of Backus (1962) for layering of isotropic elastic materials. One very important fact that is known about these equations is that they reduce to isotropic results, having \( a = c, b = f, \) and \( l = m, \) if the shear modulus \( \mu \) is a constant, regardless of the behavior of \( \lambda. \) Another fact that can easily be checked is that \( a = b + 2m, \) which is a general condition (mentioned earlier) that must be satisfied for all transversely isotropic materials and shows that there are only five independent constants.

**BULK MODULUS \( K \) AND ESTIMATES OF \( K_{EFF} \) FOR POLYCRYSTALS**

**Bulk modulus \( K \)**

The bulk modulus \( K \) for the layered system is well-defined. Assuming that the external compressional/tensional stresses are hydrostatic so that \( \sigma_{11} = \sigma_{22} = \sigma_{33} = \sigma, \) and the total volume strain is \( e = e_{11} + e_{22} + e_{33}, \) we find directly that

\[
e = \frac{\sigma}{K},
\]

where, after some rearrangement of the resulting expressions, we have

\[
K = f + \left( \frac{1}{\mu_1^+} + \frac{1}{2\mu_3^±} \right)^{-1}.
\]

The new terms in (18) are defined by

\[
2\mu_1^+ \equiv a + b - 2f \quad \text{and} \quad 2\mu_3^± \equiv c - f.
\]

and are measures of shear behavior in the simple layered system. We can write the formula (18) this way, or in another suggestive form

\[
\frac{1}{K - f} = \frac{1}{a - m - f} + \frac{1}{c - f},
\]

in anticipation of results concerning various shear modulus measures that will be discussed at length in the next section.
Effective bulk modulus estimates for polycrystals

Assuming that the simple layered material we have been studying is present locally at some small (micro-)scale in a heterogeneous (macro-)medium, and also assuming that the axis of symmetry of these local constituents is randomly distributed so the whole composite medium is isotropic, then we have a polycrystalline material. The well-known results of Reuss and Voigt provide simple and useful estimates for moduli of polycrystals [actually lower and upper bounds on the moduli as shown by Hill (1952)].

Reuss average

The Reuss average is obtained by assuming constant stress, which is the same condition we applied already to estimate the bulk modulus $K$ for the simple layered material. The well-known result in terms of compliances ($S = C^{-1}$) is

$$
\frac{1}{K_R} = 2S_{11} + 2S_{12} + S_{33} + 4S_{13}.
$$

(21)

It is straightforward to show that this produces exactly the same result as either (18) or (20). So $K_R = K$, which should be interpreted as meaning that the lower bound on the bulk modulus in the polycrystalline system is equal to the bulk modulus of the simple layered system.

Voigt average

The Voigt average is obtained with a constant strain assumption, and leads directly to the estimate in terms of stiffnesses

$$
K_V = \frac{1}{9} (2C_{11} + 2C_{12} + C_{33} + 4C_{13}) = \left[9f + 4(a - m - f) + (c - f)\right]/9 = \left[9f + 4\mu_1^* + 2\mu_3^*\right]/9,
$$

(22)

where the final equality makes use of the definitions from (19). It is well-known that $K_V \geq K_{\text{eff}} \geq K_R$ [see Hill (1952)].

For an isotropic system, the bulk modulus $K = \lambda + 2\mu/3$. The results (18) and (22) obtained for $K_{\text{eff}}$ suggest that $f$ plays the role of $\lambda$ and that some combination or combinations of the constants $\mu_1^*$ and $\mu_3^*$ may play the role of the one nontrivial effective shear modulus $G_{\text{eff}}$ for both the simple layered system and for the polycrystalline system. The combinations arising here are

$$
G_{KR} \equiv \left[\frac{2}{3} \left(\frac{1}{\mu_1^*} + \frac{1}{2\mu_3^*}\right)\right]^{-1} \quad \text{and} \quad G_{KV} \equiv \left(2\mu_1^* + \mu_3^*\right)/3,
$$

(23)

the harmonic mean and mean, respectively, of $\mu_1^*$ and $\mu_3^*$ after having accounted for the duplication of $\mu_1^*$ in the system. We might anticipate (incorrectly!) that these two estimates of the magnitude the remaining shear response will be, respectively, the lowest and the highest that we will find. However, in fact both these estimates usually take lower values than the ones we study more carefully in the next section.
EFFECTIVE SHEAR MODULUS ESTIMATES FOR SIMPLE LAYERS AND FOR POLYCRYSTALS

To understand the effective shear modulus \( G_{\text{eff}} \) and how to estimate it, we need first to introduce some facts about the eigenvalue structure of the elasticity matrices.

**Singular value decomposition**

The singular value decomposition (SVD), or equivalently the eigenvalue decomposition in the case of a real symmetric matrix, for (6) is relatively easy to perform. We can immediately write down four eigenvectors:

\[
\begin{bmatrix}
0 \\
0 \\
0 \\
1 \\
0 \\
0
\end{bmatrix}, \quad
\begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
1 \\
0
\end{bmatrix}, \quad
\begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
1
\end{bmatrix}, \quad
\begin{bmatrix}
1 \\
-1 \\
0 \\
0 \\
0 \\
0
\end{bmatrix}.
\]

(24)

and their corresponding eigenvalues, respectively \( 2l, 2l, 2m, \) and \( a - b = 2m \). All four correspond to shear modes of the system. The two remaining eigenvectors must be orthogonal to all four of these and therefore both must have the general form

\[
\begin{bmatrix}
1 \\
1 \\
\Omega \\
0 \\
0 \\
0
\end{bmatrix}^T
\]

(25)

with corresponding eigenvalue

\[
\omega = a + b + f\Omega.
\]

(26)

The remaining condition that determines both \( \Omega \) and \( \omega \) is

\[
\omega\Omega = 2f + c\Omega,
\]

(27)

which, after substitution for \( \omega \), yields a quadratic equation having the solutions

\[
\Omega_{\pm} = \frac{1}{2} \left( \frac{c - a - b}{f} \pm \sqrt{8 + \left( \frac{c - a - b}{f} \right)^2} \right).
\]

(28)

Then (26) and (28) imply that

\[
\omega_+ \omega_- = (a + b)c - 2f^2
\]

(29)

and

\[
\omega_+ + \omega_- = a + b + c = 2(a - m) + c,
\]

(30)
which are two identities that will be used repeatedly later.

The ranges of values for $\Omega_{\pm}$ are $0 \leq \Omega_{\pm} \leq \infty$ and (since $\Omega_{-} = -2/\Omega_{+}$) $-\infty \leq \Omega_{-} \leq 0$. The interpretation of the solutions $\Omega_{\pm}$ is simple for the isotropic limit where $\Omega_{+} = 1$ and $\Omega_{-} = -2$, corresponding respectively to pure compression and pure shear modes. For all other cases, these two modes have mixed character, indicating that pure compression cannot be excited in the system, and must always be coupled to shear. Some types of pure shear modes can still be excited even in the nonisotropic cases, because the other four eigenvectors in (24) are not affected by this coupling, and they are all pure shear modes. Pure shear (in strain) and pure compressional (in stress) modes are obtained as linear combinations of these two mixed modes according to

$$
\alpha \begin{pmatrix}
1 \\
1 \\
\Omega_+
\end{pmatrix} + 
\begin{pmatrix}
1 \\
1 \\
\Omega_-
\end{pmatrix} = (1 + \alpha) \begin{pmatrix}
1 \\
1 \\
-2
\end{pmatrix},
$$

with $\alpha = -2(\Omega_{+} - 1)/[\Omega_{+}(\Omega_{+} + 2)]$ for pure shear, and

$$
\begin{pmatrix}
1 \\
\Omega_+
\end{pmatrix} + \beta 
\begin{pmatrix}
1 \\
\Omega_-
\end{pmatrix} = (1 + \beta) \begin{pmatrix}
1 \\
1 \\
0
\end{pmatrix},
$$

and with $\beta = \Omega_{+}(\Omega_{+} - 1)/(\Omega_{+} + 2)$ for pure compression.

To understand the behavior of $\Omega_{+}$ in terms of the layer property fluctuations, it is first helpful to note that the pertinent functional $f(x) = \frac{1}{2} \left[-x + \sqrt{8 + x^2}\right]$ is easily shown to be a monotonic function of its argument $x$. So it is sufficient to study the behavior of the argument $x = (a + b - c)/f$.

**Exact results in terms of layer elasticity parameters**

Combining results from Eqs. (11)–(14), we find after some work on rearranging the terms that

$$
\frac{a + b - c}{f} = \left(\frac{\lambda}{\lambda + 2\mu}\right)^{-1} \left[\frac{\lambda}{\lambda + 2\mu} - 6 \left(\frac{\Delta\mu}{\lambda + 2\mu}\right) - 8 \left(\frac{\mu^2}{\lambda + 2\mu} - \left(\frac{\mu}{\lambda + 2\mu}\right)^2\right)\right],
$$

where $\Delta\mu \equiv \mu - \langle \mu \rangle$ is the deviation of the shear modulus from the layer-averaged shear modulus $\mu$. Note that the term in curly brackets in (33) is in Cauchy-Schwartz form (i.e., $\langle a^2 \rangle\langle b^2 \rangle - \langle ab \rangle^2 \geq 0$) and therefore is always non-negative. This term is also effectively quadratic in the
deviations of $\mu$ from its layer average, and thus is of higher order than the term explicitly involving $\Delta \mu$. This fact, together with the fact that $\langle \Delta \mu / \mu \rangle = 1 - \langle \mu \rangle \langle 1 / \mu \rangle \leq 0$, suggests that the dominant corrections to unity (since the leading term is exactly unity) for this expression will be positive if $\lambda$ and $\mu$ are positively correlated throughout all the layers, but the correction could be negative in cases where there is a strong negative correlation between $\lambda$ and $\mu$. If the fluctuations in shear modulus are very large throughout the layered medium, then the quadratic terms can dominate, in which case the result could be less than unity. Numerical examples developed by applying a code of V. Grechka [used previously in a similar context by Berryman et al. (1999)] confirm (and, in fact, motivated) these analytical results.

Our main conclusion is that the shear modulus fluctuations giving rise to the anisotropy due to layering are (as expected) the main source of deviations of (33) from unity. But, there are some other more subtle effects present having to do with the interplay between $\lambda$, $\mu$ correlations as well as the strength of the $\mu$ fluctuations that ultimately determine the magnitude of the deviations of (33) from unity.

A fifth effective shear modulus?

From what has gone before, we know that there are four eigenvalues of the system that are easily identified with effective shear moduli (two are $l$ and two are $m$). The bulk modulus $K$ of the simple system is well-defined, and the bounds on $K_{\text{eff}}$ for polycrystals are quite simple to apply and interpret. But we are still missing an important element of the overall picture of this system, and that is how the remaining degree of freedom is to be interpreted. It seems clear that it should be interpreted as an effective (quasi-)shear mode, since we have already accounted for the bulk mode. It is also clear that analysis of this remaining degree of freedom is not so easy because it is never an eigenfunction of the elasticity/poroelasticity tensor except in the cases that are trivial and therefore of interest to us here only as the isotropic baseline for comparisons.

Although it seems problematic to define a new shear modulus arbitrarily, we will now proceed to enumerate a number of possibilities and then, by a process of elimination, arrive at what appears to be a useful result.

**Bounds for polycrystals**

**Reuss average for shear**

The well-known Reuss result for shear modulus is

$$G_R^{-1} = \frac{1}{15} \left( 8S_{11} - 4S_{12} + 4S_{33} - 8S_{13} + 6S_{44} + 3S_{66} \right)$$

$$= \frac{1}{15} \left( \frac{6}{l} + \frac{6}{m} + \frac{9f + 4(a - m - f) + (c - f)}{(a - m)c - f^2} \right), \tag{34}$$

but

$$(a - m)c - f^2 = \left[ (a - m - f) + (c - f) \right] K_R = [\mu^*_1 + 2\mu^*_3] K_R, \tag{35}$$
again using the definitions from (19). Combining these results, we have

\[ G_R^{-1} = \frac{1}{5} \left( \frac{2}{l} + \frac{2}{m} + \left[ \mu_1^* + 2\mu_3^* \right]^{-1} \frac{3K_V}{K_R} \right) \] (36)

Since the multiplicity of the shear modulus eigenvalues \( l \) and \( m \) is properly accounted for (2 and 2, respectively, out of 5), this result strongly suggests that one reasonable estimate of the fifth shear modulus for the system is

\[ G_{eff}^{(1)} \equiv (\mu_1^* + 2\mu_3^*) \frac{K_R}{3K_V}. \] (37)

**Voigt average for shear**

The Voigt average for shear modulus is

\[ G_V = \frac{1}{15} \left[ 2C_{11} - C_{12} + C_{33} - 2C_{13} + 6C_{44} + 3C_{66} \right] = \frac{1}{15} \left[ (a - m - f) + (c - f) + 6l + 6m \right]. \] (38)

[Note that (38) corrects an error in equation (69) of Berryman and Wang (2001).] This result shows that the combinations \( 2(a - m - f) \) and \( (c - f) \) again play the roles of twice a shear modulus contribution. By analogy to (36) and (37), we can define another effective constant

\[ G_{eff}^{(2)} = \frac{[(a - m - f) + (c - f)]}{3} = \left( \mu_1^* + 2\mu_3^* \right) / 3. \] (39)

This constant has a sensible dependence on these parameters and is consistent with the rest of our analysis (see Discussion of \( G_{eff} \) below), and also is a somewhat simpler form than (37). Both this constant and \( G_{eff}^{(1)} \) reappear in the later analyses. Because these two estimates are related to rigorous bounds, it seems that estimates not lying in the range from \( G_{eff}^{(1)} \) to \( G_{eff}^{(2)} \) can surely be excluded from further consideration.

**Estimates based on matrix invariants 1. Trace estimates**

The trace of the stiffness matrix \( C \) is an invariant, and equals the sum of its eigenvalues. Similarly, the trace of the compliance matrix \( S \) is also invariant, and equals the sum of its eigenvalues, which are the inverses of the eigenvalues of \( C \). These facts provide two more ways of obtaining estimates of the \( G_{eff} \) we seek.

**Trace estimate from \( S \)**

After eliminating the four eigenvalues associated with simple shear, the remainder of the trace of \( S \) is just the sum of the inverses of \( \omega_+ \) and \( \omega_- \). To obtain an estimate of \( G_{eff} \), we again make use of the known bulk modulus \( K \) and set

\[ \frac{1}{3K} + \frac{1}{2G_{eff}^{(3)}} = \frac{1}{\omega_+} + \frac{1}{\omega_-}, \] (40)
which can easily be shown to imply that
\[ G^{(3)}_{\text{eff}} = (\mu^*_1 + 2\mu^*_3) \frac{K_R}{3K_V} = G^{(1)}_{\text{eff}}. \]  \hfill (41)

So this compliance estimate again produces the same result found earlier in (37).

\textit{Trace estimate from }C\textit{
}

Again eliminating the four eigenvalues associated with simple shear, the remainder of the trace of \( C \) is just the sum of \( \omega_+ \) and \( \omega_- \). To obtain another estimate of \( G_{\text{eff}} \), we make use of the known bulk modulus \( K \) as before and set
\[ 3K + 2G^{(4)}_{\text{eff}} = \omega_+ + \omega_. \]  \hfill (42)

After some manipulation, we find
\[ 2G^{(4)}_{\text{eff}} = \frac{2(a - m - f)^2 + (c - f)^2}{(a - m - f) + (c - f)} = 2\frac{(\mu^*_1)^2 + 2(\mu^*_3)^2}{\mu^*_1 + 2\mu^*_3}. \]  \hfill (43)

So this estimate does not agree with any of the others, but it is nevertheless an interesting new combination of the shear modulus measures \( (a - m - f) \) and \( (c - f) \).

\textbf{Estimates based on matrix invariants 2. Determinant estimates}

Another invariant of an elasticity matrix is its determinant, which is given by the product of its eigenvalues. Thus,
\[ \det C = \omega_+ \omega_-(2l)^2(2m)^2 \quad \text{and} \quad \det S = [\det C]^{-1}, \]  \hfill (44)

and so there is only one new estimate available based on this fact. In particular, if we assume that a reasonable estimate of \( G_{\text{eff}} \) might be obtained from this condition, then we would again make use of the four known eigenvalues for shear, and the bulk modulus for the simple layered medium. Setting
\[ \det C = (3K)(2G_{\text{eff}})(2l)^2(2m)^2, \]  \hfill (45)

comparing to (44), and recalling that \( K = K_R \), we immediately find
\[ 2G^{(5)}_{\text{eff}} = \frac{\omega_+ + \omega_-}{3K} = 2G^{(2)}_{\text{eff}}, \]  \hfill (46)

which is a repeat of an earlier estimate.

It is easy to show that \( G^{(5)}_{\text{eff}} \leq G^{(4)}_{\text{eff}} \), and, since \( G^{(2)}_{\text{eff}} \) was derived from the Voigt average for shear, now we should be able to exclude \( G^{(4)}_{\text{eff}} \) safely from any further consideration.
Relationship among estimates

These three shear modulus estimates, $G_{\text{eff}}^{(3)}$, $G_{\text{eff}}^{(4)}$, and $G_{\text{eff}}^{(5)}$, can be related, by making use of the fact that $\omega_+ + \omega_- = \omega_+ \omega_- (1/\omega_+ + 1/\omega_-)$. The main result is

$$\frac{G_{\text{eff}}^{(5)} - G_{\text{eff}}^{(3)}}{2G_{\text{eff}}^{(3)}} = \frac{G_{\text{eff}}^{(4)} - G_{\text{eff}}^{(5)}}{3K}, \quad (47)$$

which shows first of all (since $K > 0$ and $G_{\text{eff}}^{(3)} > 0$) that if any two of these constants are equal, then they are all equal. Eq. (47) also shows that the value of $G_{\text{eff}}^{(5)}$ always lies between the other two estimates, so in general $G_{\text{eff}}^{(4)} \geq G_{\text{eff}}^{(5)} \geq G_{\text{eff}}^{(3)}$, with equality holding only in the case of isotropic composites.

Energy estimates

The energy $W$ of the elastic system can be written conveniently in either of two ways, based on strain and stress respectively:

$$2W = a(e_{11}^2 + e_{22}^2) + ce_{33}^2 + 2be_{11}e_{22} + 2fe_{33}(e_{11} + e_{22})$$

$$= S_{11}(\sigma_{11}^2 + \sigma_{22}^2) + S_{33}\sigma_{33}^2 + 2S_{12}\sigma_{11}\sigma_{22} + 2S_{13}\sigma_{33}(\sigma_{11} + \sigma_{22}). \quad (48)$$

We can use these energy formulas to help decide whether any of the estimates obtained so far are actually fundamental quantities, by which we mean that they actually provide useful measures of the energy stored in the system. For example, it is well-known (as we have already discussed) that, if we set $\sigma_{11} = \sigma_{22} = \sigma_{33} = \sigma$, then

$$W = \sigma^2/2K, \quad (49)$$

where $K$ is given by (20). Thus, even though $K$ is not simply related to the eigenvalues of the system in general, it is still the fundamental measure of compressional energy in the simple layered system.

To check to see if any of the shear constants studied so far might play a similar role for shear, we can set $e_{11} = e_{22} = -e_{33}/2 = e/\sqrt{6}$. Then, we find that

$$W = 2[(a - m - f) + (c - f)]e_{11}^2 = G_{\text{eff}}^{(2)} e^2. \quad (50)$$

So there is no ambiguity in the result for the shear energy. Clearly, $G_{\text{eff}}^{(2)}$ plays the same role for shear energy that $K$ plays for bulk energy in this system, again regardless of the fact that it is not simply related to the eigenvalues.

Discussion of $G_{\text{eff}}$

Our goal is to obtain some new insight into the effective shear modulus of a poroelastic system in order to understand how the shear and bulk modes become coupled in such systems.
and thereby violate Gassmann’s (1951) results for quasi-static systems having nonzero fluid permeability. The purpose of such an analysis will ultimately be to understand why some laboratory ultrasonics data show that the effective shear modulus of porous saturated and partially saturated rocks/systems has a substantial dependence on saturation when Gassmann’s result would appear to deny the possibility of such behavior. It was demonstrated by Berryman and Wang (2001) that such deviations from Gassmann’s predictions are expected, but they are surely not universal. For example, local elastic isotropy and spherical pores are two cases in which the shear modulus should remain independent of pore-fluid saturation.

For the transversely isotropic system arising from finely layered isotropic layers, we know that four out of five of the shear modes of the system are always independent of fluid saturation. They depend only on the Reuss and Voigt averages of the shear moduli present in the layered system. The remaining two modes are both of mixed character, not being pure shear or pure compression. So at some intellectual level it is clear that this coupling imposed through the eigenvalues is the reason for the shear wave dependence on fluid saturation. But this mathematical statement, although surely correct, is not really helpful in achieving our goal of understanding how the mixing of these effects happens down at the microscale. The best possible way to elucidate this behavior would be to make use of a formula for shear modulus (if one were known), containing the desired effects in it explicitly. But, a rigorous formula of this type is probably not going to be found. So, the next best option is to have in hand a formula that, although it is known to be approximate, still has the right structure and thereby permits analysis to proceed.

What is the right structure? The appropriate shear strain that contains all the effects of interest clearly is of the form \((1, 1, -2, 0, 0, 0)^T\). If we apply this shear strain to the stiffness matrix, the two distinct stresses generated are proportional to \(2(a - m - f)\) [twice] and \((c - f)\) [once]. So the effective shear modulus we seek should depend on these two quantities, each of which acts like \(2\mu\) in the limit of an isotropic system. But in the nonisotropic cases of most interest, these combinations both include coupling between \(\lambda\) and \(\mu\) of the layers through the Backus formulas — coupling that is good for our present purposes. Furthermore, since there are two nontrivial constants, it is not obvious what combinations to choose for study. But, the preceding analysis shows that one likely candidate for the effective shear modulus is \(G^{(1)}\), since it appears naturally in two out of five of the main cases considered above. \(G^{(2)}\) also appears in two similar cases, as well as in the energy estimate.

The shear moduli \(G^{(1)}\) and \(G^{(2)}\) are clearly not eigenvalues; but the most likely candidate eigenvalue \(\omega_\perp\) relevant for our study is even more difficult to analyze and interpret (both because of its eigenvector’s mixed character and because of the complicated formula relating it to the elastic constants) than either \(G^{(1)}\) or \(G^{(2)}\). There are other choices that could be made, but we will give preference to \(G^{(1)}\) and \(G^{(2)}\) in the following discussion — both for definiteness and because they do seem to be useful constants to quantify and to help focus our attention. Of these two, \(G^{(2)}\) is especially appealing because it is the one obtained using energy estimates.
POROUS ELASTIC MATERIALS CONTAINING FLUIDS

Now we want to broaden our outlook and suppose that the materials composing the layers are not homogeneous isotropic elastic materials, but rather poroelastic materials containing voids or pores. The pores may be air-filled, or alternatively they may be partially or fully saturated with a liquid, some gas, or a fluid mixture. For simplicity, we will suppose here that the pores are either air-filled or they are fully saturated with some other homogeneous fluid. When the porous layers are air-filled, it is generally adequate to assume that the analysis of the preceding section holds, but with the new interpretation that the Lamé parameters are those for the porous elastic medium in the absence of saturating liquids. The resulting effective constants \( \lambda_{dr} \) and \( \mu_{dr} \) are then said to be those for “dry” — or somewhat more accurately “drained” — conditions. These constants are also sometimes called the “frame” constants, to distinguish them from the constants associated with the solid materials composing the frame, which are often called the “grain” or “mineral” constants.

One simplification that arises immediately is due to the fact, according to a quasi-static analysis of Gassmann (1951), that the presence of pore fluids has no mechanical effect on the layer shear moduli, so \( \mu_{dr} = \mu \). There may be other effects on the layer shear moduli due to the presence of pore fluids, such as softening of cementing materials or expansion of interstitial clays, which we will term “chemical” effects to distinguish them from the purely mechanical effects to be considered here. We neglect all such chemical effects in the following analysis. This means that the lamination analysis for the four effective shear moduli \( (l, l, m, m) \) associated with eigenvectors (since they are uncoupled from the analysis involving \( \lambda \)) does not change in the presence of fluids. Thus, equations (15) and (16) continue to apply for the poroelastic problem, and we can therefore simplify our system of equations in order to focus on the parts of the analysis that do change in the presence of fluids.

The presence of a saturating pore fluid introduces the possibility of an additional control field and an additional type of strain variable. The pressure \( p_f \) in the fluid is the new field parameter that can be controlled. Allowing sufficient time for global pressure equilibration will permit us to consider \( p_f \) to be a constant throughout the percolating (connected) pore fluid, while restricting the analysis to quasistatic processes. The change \( \zeta \) in the amount of fluid mass contained in the pores [see Berryman and Thigpen (1985)] is the new type of strain variable, measuring how much of the original fluid in the pores is squeezed out during the compression of the pore volume while including the effects of compression or expansion of the pore fluid itself due to changes in \( p_f \). It is most convenient to write the resulting equations in terms of compliances rather than stiffnesses, so the basic equation for an individual layer in the stack of layers to be considered takes the form:

\[
\begin{pmatrix}
e_{11} \\
e_{22} \\
e_{33} \\
-\zeta
\end{pmatrix} = \begin{pmatrix}S_{11} & S_{12} & S_{12} & -\beta \\
S_{12} & S_{11} & S_{12} & -\beta \\
S_{12} & S_{12} & S_{11} & -\beta \\
-\beta & -\beta & -\beta & \gamma
\end{pmatrix} \begin{pmatrix}\sigma_{11} \\
\sigma_{22} \\
\sigma_{33} \\
p_f
\end{pmatrix}.
\]

The constants appearing in the matrix on the right hand side will be defined in the following two paragraphs. It is important to write the equations this way rather than using the inverse
relation in terms of the stiffnesses, because the compliances $S_{ij}$ appearing in (51) are simply related to the drained constants $\lambda_{dr}$ and $\mu_{dr}$ in the same way they are related in normal elasticity, whereas the individual stiffnesses obtained by inverting the equation in (51) must contain coupling terms through the parameters $\beta$ and $\gamma$ that depend on the pore and fluid compliances. Thus, we find easily that

$$S_{11} = \frac{1}{E_{dr}} = \frac{\lambda_{dr} + \mu}{\mu(3\lambda_{dr} + 2\mu)} \quad \text{and} \quad S_{12} = -\frac{\nu_{dr}}{E_{dr}}, \quad (52)$$

where the drained Young’s modulus $E_{dr}$ is defined by the second equality of (52) and the drained Poisson’s ratio is determined by

$$\nu_{dr} = \frac{\lambda_{dr}}{2(\lambda_{dr} + \mu)}. \quad (53)$$

When the external stress is hydrostatic so $\sigma = \sigma_{11} = \sigma_{22} = \sigma_{33}$, equation (51) telescopes down to

$$\begin{pmatrix} e \\ -\zeta \end{pmatrix} = \frac{1}{K_{dr}} \begin{pmatrix} 1 & -\alpha \\ -\alpha & \alpha/B \end{pmatrix} \begin{pmatrix} \sigma \\ -p_f \end{pmatrix}, \quad (54)$$

where $e = e_{11} + e_{22} + e_{33}$, $K_{dr} = \lambda_{dr} + \frac{2}{3}\mu$ is the drained bulk modulus, $\alpha = 1 - K_{dr}/K_m$ is the Biot-Willis parameter (Biot and Willis, 1957) with $K_m$ being the bulk modulus of the solid minerals present, and Skempton’s pore-pressure buildup parameter $B$ (Skempton, 1954) is given by

$$B = \frac{1}{1 + K_p(1/K_f - 1/K_m)}. \quad (55)$$

New parameters appearing in (55) are the bulk modulus of the pore fluid $K_f$ and the pore modulus $K_p^{-1} = \alpha/\phi K_{dr}$ where $\phi$ is the porosity. The expressions for $\alpha$ and $B$ can be generalized slightly by supposing that the solid frame is composed of more than one constituent, in which case the $K_m$ appearing in the definition of $\alpha$ is replaced by $K$, and the $K_m$ appearing explicitly in (55) is replaced by $K_\phi$ [see Brown and Korrinda (1975), Rice and Cleary (1976), Berryman and Milton (1991), Berryman and Wang (1995)]. This is an important additional complication (Berge and Berryman, 1995), but one that we choose not to pursue here.

Comparing (51) and (54), we find easily that

$$\beta = \frac{\alpha}{3K_{dr}} \quad \text{and} \quad \gamma = \frac{\alpha}{BK_{dr}}. \quad (56)$$

All the constants are defined now in terms of “easily” measurable quantities.

When the mechanical changes (i.e., applied stress or strain increments) in the system happen much more rapidly than the fluid motion can respond, the system is “undrained” — a situation commonly modeled by taking $\zeta = 0$ in the layer, $p_f = B\sigma$, and $e = (1 - \alpha B)\sigma / K_{dr}$. So the undrained bulk modulus of the layer is

$$K_u \equiv \frac{K_{dr}}{(1 - \alpha B)}. \quad (57)$$
Then, it is not hard to show that

$$
\begin{pmatrix}
\sigma_{11} \\
\sigma_{22} \\
\sigma_{33}
\end{pmatrix} =
\begin{pmatrix}
\lambda_u + 2\mu & \lambda_u & \lambda_u \\
\lambda_u & \lambda_u + 2\mu & \lambda_u \\
\lambda_u & \lambda_u & \lambda_u + 2\mu
\end{pmatrix}
\begin{pmatrix}
\epsilon_{11} \\
\epsilon_{22} \\
\epsilon_{33}
\end{pmatrix},
$$

(58)

where the undrained Lamé constant is given by

$$
\lambda_u = K_u - \frac{2\mu}{3}.
$$

For our purposes, the language used here to describe drained/undrained layers is intended to convey the same meaning as opened/closed pores at the boundaries separating the layers. If pores are open, fluid can move between layers, so pressure can equilibrate over long periods of time. If the pores are closed at these interfaces, pressure equilibration can occur within each layer, but not between layers — no matter how long the observation time.

With (58) and (59), we can now repeat the Backus analysis for a layered system in which each layer is undrained. The only difference is that everywhere \( \lambda \) appeared explicitly before, now \( \lambda_u \) is substituted. Furthermore, the constants resulting from the Backus lamination analysis can now be distinguished as \( a_u, b_u, c_u, \) and \( f_u \) for the undrained system. The constants without \( u \) subscripts are assumed to be drained, i.e., \( a = a_{dr} \), etc. The constants \( l \) and \( m \) are the same in both drained and undrained systems, since they do not depend on either \( \lambda \) or \( \lambda_u \).

Carrying through this analysis for our main estimate \( G_{eff}^{(2)} \), we find, after some rearrangement of terms, that

$$
G_{eff}^{(2)} = m - \frac{4c_u}{3} \left[ \frac{\mu^2}{\lambda_u + 2\mu} \left( \frac{1}{\lambda_u + 2\mu} \right) - \left( \frac{\mu}{\lambda_u + 2\mu} \right)^2 \right].
$$

(60)

Equation (60) is the main result of this paper. It incorporates all the earlier work by making use of the effective shear modulus \( G_{eff}^{(2)} \) selected in the end by using energy estimates. Then it is evaluated explicitly here for the simple layered system. The term in square brackets is in Cauchy-Schwartz form (i.e., \( \langle \alpha^2 \rangle \langle \beta^2 \rangle \geq \langle \alpha \beta \rangle^2 \)), and thus this term is always nonnegative. The contribution of this term is therefore a nonpositive correction to the leading term \( m \). Further, it is clear that the correction is second order in the fluctuations in the shear modulus throughout the layered material. This fact means that fluctuations must be fairly large before any effect can be observed, since there are second order subtractions but no first order corrections at all. It is also easy to show from this formula that the range of possible values for this shear modulus is \( l \leq G_{eff}^{(2)} \leq m \), because as \( K_u \) in the layers ranges from zero to infinity the corrections to \( m \) range from \( l - m \) to 0. Finally, we note that the effect of an increase in the undrained constant \( \lambda_u \) is to reduce the magnitude of these correction terms. Thus, since a reduction in a negative contribution leads to a positive contribution, it is also clear that an increase in Skempton’s coefficient \( B \) always leads to an increase in the effective shear modulus \( G_{eff}^{(2)} \), as we expected.

Since each layer in the stack is isotropic, the result (60) may appear to contradict the earlier results of Berryman and Wang (2001), showing that shear modulus dependence on fluid properties does not and cannot occur in microhomogeneous isotropic media. But these layered
media are not isotropic everywhere. In particular, if we consider a point right on the boundary between any two layers, the elastic properties look very anisotropic at these points. So it is exactly at, and near the vicinity of, these boundary points where deviations from Gassmann’s results can and do arise, leading to the result (60).

We will check these ideas by showing some numerical examples in the next section.

**NUMERICAL EXAMPLES**

As we learned in the previous section, large fluctuations in the layer shear moduli are required before the predicted deviations from Gassmann’s quasi-static constant shear modulus result will become noticeable. To generate a model that demonstrates these results, I made use of a code of V. Grechka [used previously in a joint publication (Berryman et al., 1999)] and then I arbitrarily picked one of the models that seemed to be most interesting for the present purposes. The parameters of this model are displayed in Table 1. The results of the calculations are shown in Figures 1 and 2.

The model calculations were simplified in one way, which is that the value of the Biot-Willis parameter was chosen to be a uniform value of $\alpha = 0.8$ in all layers. We could have actually computed a value of $\alpha$ from the other layer parameters, but to do so would require another assumption about the porosity values in each layer. Doing this seemed of little use because we are just trying to show in a simple way that the formulas given here really do produce the results predicted. Furthermore, if $\alpha$ is a constant, then it is only the product $\alpha B$ that matters. Since we are using $B$ as the plotting parameter, whatever choice of constant $\alpha \leq 1$ is made, it mainly determines the maximum value of the product $\alpha B$ for $B$ in the range $[0, 1]$. So, for a parameter study, it is only important not to choose too a small value of $\alpha$, which is why the choice $\alpha = 0.8$ was made. This means that the maximum amplification of the bulk modulus due to fluid effects can be as high as a factor of 5 for the present example.

**TABLE 1.** Layer parameters for the three materials in the simple layered medium used to produce the examples in Figures 1 and 2.

<table>
<thead>
<tr>
<th>Constituent</th>
<th>$K$ (GPa)</th>
<th>$\mu$ (GPa)</th>
<th>$z$ (m/m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9.4541</td>
<td>0.0965</td>
<td>0.477</td>
</tr>
<tr>
<td>2</td>
<td>14.7926</td>
<td>4.0290</td>
<td>0.276</td>
</tr>
<tr>
<td>3</td>
<td>43.5854</td>
<td>8.7785</td>
<td>0.247</td>
</tr>
</tbody>
</table>

The results for bulk modulus in Figure 1 show that $K = K_R$ and $\omega_+ / 3$ are always quite close in value. The Voigt upper bound does bound both $K_R$ and $\omega_+$ from above as expected. Furthermore, the simple Gassmann estimate based on the value of the drained constant ($B = 0$) gives a remarkably good fit to these estimates with no free fitting parameters. This good agreement with Gassmann may depend in part on our choosing $\alpha$ to be a constant. (Without this choice, a direct comparison in fact could not have been made here.)

The results for the effective shear moduli $G_{eff}^{(2)}, G_{eff}^{(1)}$, and the quasi-shear mode eigenvalue
Figure 1: Bulk modulus as a function of Skempton’s coefficient $B$. The Biot-Willis parameter was chosen to be $\alpha = 0.8$, constant in all layers.

Figure 2: Shear modulus as a function of Skempton’s coefficient $B$. The Biot-Willis parameter was chosen to be $\alpha = 0.8$, constant in all layers.

contribution $\omega_-/2$, show that $G_{\text{eff}}^{(2)} \simeq \omega_-/2$, although it actually does bound this parameter from above. The third shear modulus estimate $G_{\text{eff}}^{(1)}$ is far from the other two. We also display a constant value of shear modulus based on the Gassmann-like prediction of constant shear modulus equaling the drained modulus, but it is clear that this labeling is not fair to Gassmann as his result was for microhomogeneous materials, and therefore does not strictly speaking apply to layered materials at all. Nevertheless, we see that Gassmann’s result for bulk modulus did a very creditable job of matching the bulk modulus values, even though that too was not a fair comparison for exactly the same reasons. This shows that Gassmann’s results are much more sensitive to heterogeneity in shear modulus than they are to heterogeneity in the bulk modulus for these layered materials.

DISCUSSION AND CONCLUSIONS

Our main conclusion is that the two best constants to study for our present purposes are the ones derived from the Reuss and Voigt bounds, $G_{\text{eff}}^{(1)}$ and $G_{\text{eff}}^{(2)}$. Furthermore, these estimates
are naturally paired with the Voigt and Reuss bounds on bulk modulus through the product formulas (see the Appendix for a derivation)

\[(3K_V)(2G_{eff}^{(1)}) = (3K_R)(2G_{eff}^{(2)}) = \omega_+\omega_-\text{.} \quad (61)\]

The product formulas are true for both drained and undrained constants, but of course the numerical values of these constants differ in going from drained to undrained constants. Of these two estimates, \(G_{eff}^{(2)}\) has the simplest form and further is paired with the Reuss bound on bulk modulus, which is actually the true bulk modulus of the simple (not polycrystalline) layered system. So we believe, based on theoretical considerations – especially using energy estimates, that this choice is worthy of special attention. But the numerical experiments show that \(\omega_-/2\) has quantitatively very similar values while \(G_{eff}^{(1)}\) has qualitatively similar behavior in the cases studied. So the practical advantages of this one choice over the others may not always be overwhelming.

Eq. (60) shows explicitly that \(G_{eff} = m\) together with generally negative corrections whose magnitude depends strongly on fluctuations in layer shear modulus. The contribution of these correction terms decreases as Skempton’s coefficient \(B\) increases, so \(G_{eff}^{(2)}\) is a monotonically increasing function of \(B\). This is exactly the behavior we were trying to explicate in the present paper, so (60) is one example of the types of aid-to-intuition that we were seeking. The leading term in \(G_{eff}^{(2)}\) is also easy to understand, as \(G_{eff}^{(2)}\) was first obtained here using the Voigt average, which is an upper bound on the overall behavior; so it is natural that the leading term is \(m\), which is just the Voigt average of the \(\mu\)’s in the layers.

ACKNOWLEDGMENTS

I thank V. Y. Grechka, P. A. Berge, and H. F. Wang for earlier codes and research results that have been retooled here to play a new role. I thank Graeme Milton for various comments and pointers to the literature on lamination analysis.

APPENDIX: PRODUCT FORMULAS

This Appendix will clarify the derivation of the product formulas relating the bulk modulus \(K = K_R\) and \(G_{eff}^{(2)}\). The product formula for \(K_V\) and \(G_{eff}^{(1)}\) is just the dual, and is obtained in a very similar fashion. Since the relevant excitation for \(G_{eff}^{(2)}\) has been shown to be a shear strain proportional to \((1, 1, −2)^T\), consider

\[
\begin{pmatrix}
C_{11} & C_{12} & C_{13} \\
C_{12} & C_{11} & C_{13} \\
C_{13} & C_{13} & C_{33}
\end{pmatrix}
\begin{pmatrix}
1 \\
1 \\
-2
\end{pmatrix}
= 
\begin{pmatrix}
C_{11} + C_{12} - 2C_{13} \\
C_{11} + C_{12} - 2C_{13} \\
-2(C_{33} - C_{13})
\end{pmatrix} \text{.} \quad (62)
\]
Then, by applying the inverse of the matrix in (62) to the left side of the equation, we get the useful formula:

$$
\begin{pmatrix}
1 & 1 & -2 \\
1 & S_{12} & S_{13} \\
-2 & S_{13} & S_{33}
\end{pmatrix}
= 
\begin{pmatrix}
C_{11} + C_{12} - 2C_{13} \\
C_{13} \\
-2(C_{33} - C_{13})
\end{pmatrix},
$$  

(63)

which supplies two independent identities among the elastic coefficients. These are

$$1 = (S_{11} + S_{12})(C_{11} + C_{12} - 2C_{13}) - 2S_{13}(C_{33} - C_{13})$$  

(64)

and

$$-1 = S_{13}(C_{11} + C_{12} - 2C_{13}) - S_{33}(C_{33} - C_{13}).$$  

(65)

Adding these together and switching to the $a, b, c$ notation for stiffnesses, we find

$$S_{11} + S_{12} + S_{13} = (S_{33} + 2S_{13}) \frac{c - f}{a + b - 2f}. $$  

(66)

Recalling that

$$\frac{1}{K} = 2(S_{11} + S_{12} + S_{13}) + (S_{33} + 2S_{13}),$$  

(67)

and then substituting (66), we find

$$\frac{1}{K} = (S_{33} + 2S_{13}) \frac{(a - m - f) + (c - f)}{a - m - f}. $$  

(68)

Then, since

$$S_{33} + 2S_{13} = \frac{2(a - m - f)}{\omega_+ \omega_-},$$  

(69)

we find immediately that

$$6K G_{eff}^{(2)} = \omega_+ \omega_-,$$  

(70)

because $G_{eff}^{(2)} = [(a - m - f) + (c - f)]/3$.

### REFERENCES


Short Note

SEPlib programming and irregular data

Robert G. Clapp

INTRODUCTION

Effectively handling the irregularity of 3-D data poses a significant challenge in 3-D seismic processing. Biondi et al. (1996) described SEP’s attempt to incorporate 3-D geometries into SEPlib’s fundamental concept of hyper-cubes. The design allowed for great flexibility, but imposed significant coding overhead. Clapp and Crawley (1996) describes the first attempt to write a simplifying interface to the underlying library. This first attempt proved to be unsuccessful because it was inefficient, was limited to Fortran90, and limited coding flexibility.

Clapp (1999) introduced the second attempt at a simplifying interface. Since this initial paper, the library has progressed significantly and is now used by almost all of SEPlib’s 3-D programs. This paper attempts to explain the design principles, interfaces, and uses of the library.

SEP3D REVIEW

SEP3D is a powerful extension to the original SEPlib hypercube format. SEP3D attempts to build as much as possible on top of the original hypercube format. As a result, a SEP3D dataset is to some extent an amalgamation of two or three SEPlib datasets. Traditionally SEPlib files are composed of two parts, an ASCII description file and a binary data file. The ASCII file provides a description and a pointer to the binary file (by default ending in @). Below is a summary of the three ASCII/binary pairs.

**History/Data** This is exactly the same as the original SEP hypercube with the limitation that the hypercube is 2-D, simply a collection of 1-D vectors (e.g. seismic traces). The history file contains pointers to the other potential file pairs: header format file (HFF, ending in H@@) and headers (ending in H@@@), and grid format file (GFF, ending ing H@@@) and the grid (ending ing H@@@@).
**HFF/Headers** This pair contains information about the 1-D vectors in the data file. It is at least a 2-D, and possibly higher dimension, hyper-cube. The first dimension is the number of headers associated with the vectors. Each header is described in the HFF by name, type, and format. In addition the headers can contain a special key, the tdata_record_number, which provides a pointer into the data file. For example, a header with a tdata_record_number of 5 would indicate it is associated with the fifth header in the data file.

**GFF/Grid** The grid and the GFF are optional portions of a SEP3D dataset. The grid attaches the axes of a hypercube onto a fundamentally irregular dataset. For example, imagine you have an irregular dataset that is a function of the coordinate tx and ty. You could impose a 2-D grid on the data described by its origin (ox and dx), sampling (dx and dy), and length (nx and ny). At each grid cell you will then have a range of possible tx and ty locations (ox+dx*ix:ox+dx*(ix+1), oy+dy*iy:oy+dy*(iy+1)). The grid will contain a list of the headers that fall within these grid limits.

SEP3D allows a significant amount of flexibility. Any dataset can have from 1-3 of these files. There is no set standard of key names or key types. The grid can be constructed in any manner. Operations can be performed on the headers and grid without accessing the data. The coding overhead to allow this flexibility is significant.

**DESIGN**

To overcome the coding overhead associated with SEP3D, SEPlib contains the tsuperset library. The general design principal is to hide everything that is complicated or cumbersome in SEP3D behind a simple interface. The tsuperset library has an internal copy of each SEP3D dataset that is accessed through a tsep3dtag. This tag is a pointer to an inaccessible C structure (the same basic concept that the normal SEP tag uses).

To use the C library you need to t#include<superset.h> and a tsep3d variable. For Fortran90 you need to use sep3d_struct_mod and use the ttype(sep3d).

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>The sep3dtag associated with the structure.</td>
</tr>
<tr>
<td>data_type</td>
<td>The type of data (float, integer, or complex).</td>
</tr>
<tr>
<td>file_type</td>
<td>The type of SEPlib data (regular, with headers, or with grid and headers).</td>
</tr>
<tr>
<td>usage_type</td>
<td>The purpose of the data (input, output, or scratch).</td>
</tr>
<tr>
<td>n,o,d,label,unit</td>
<td>Properties describing the axes.</td>
</tr>
<tr>
<td>ntraces</td>
<td>The number of traces in the dataset.</td>
</tr>
<tr>
<td>ndim</td>
<td>The number of dimensions in the dataset.</td>
</tr>
<tr>
<td>keyname,keytype,keyfmt</td>
<td>Description for the keys associated with the data.</td>
</tr>
<tr>
<td>drn</td>
<td>The tdata_record_number associated with portion of the dataset currently in the structure.</td>
</tr>
<tr>
<td>nh</td>
<td>The number of headers currently stored in the structure.</td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>headers</td>
<td>The headers currently stored in memory.</td>
</tr>
<tr>
<td>ng</td>
<td>The number of grid values currently stored in memory.</td>
</tr>
<tr>
<td>grid</td>
<td>The grid currently stored in memory.</td>
</tr>
<tr>
<td>wrote_data,wrote_headers</td>
<td>Whether or not the headers and data have been written for this dataset.</td>
</tr>
<tr>
<td>nkeys_in</td>
<td>The number of keys in the dataset.</td>
</tr>
<tr>
<td>nextra_keys</td>
<td>The number of keys not in the original dataset that will be created by some mathematical operation.</td>
</tr>
<tr>
<td>exp</td>
<td>The mathematical expressions associated with these extra keys.</td>
</tr>
<tr>
<td>su_input</td>
<td>Whether or not this is a SU dataset.</td>
</tr>
</tbody>
</table>

The programmer does not directly access this internal structure. Instead he uses a local Fortran90 type or C struct. These structures contain a subset of the internal representation, with the items up to key description in the above list included and the addition of the total number of keys in the dataset.

The programmer begins by initializing the local dataset from a file or structure, or initializes an empty container. The structure is initialized by also describing its purpose (tINPUT, OUTPUT, or tSCRATCH). The programmer can modify their local version and synchronize it at any time with the internal version.

Reading a dataset is done in two steps. First a window (potentially the entire dataset) of the overlying hypercube is requested. The library reads in the grid (if it exists) and the headers associated with this window. The return value is the total number of traces within the hypercube. The user can then: request the header values associated with individual keys, look at the grid values associated with the hypercube currently in memory, or read the data associated with the hypercube.

Writing a dataset follows a similar pattern. The user can provide a grid (if applicable) for a dataset. They pass the individual header key values to the internal representation. They then tell the library where within the data’s hypercube these values are, and what tag to write them out to. They can also specify data associated with the hypercube portion.

Having an accessible data structure for an irregular dataset has significant advantages. One of the advantages is it now becomes much easier to pass a complex object between routines. In fact, because both the C and Fortran90 structures are just representations of an underlying data type, it is possible to smoothly go between Fortran90 and C by simply passing the tsep3dtag.

**SU**

Another benefit of representing a SEP3D dataset in this manner is that interfacing with SU is basically free. SU is simply a SEP3D dataset with specific set of header values and without a grid. By writing a simple converter to and from this specific representation (tlibsepsu) it becomes simple to transform to and from SU. For example, tSu2sep is a 15 line program. In
fact, most SU libraries can be recompiled, without editing source code, to run with SEP3D datasets.

INITIALIZERS AND DESTRUCTOR’S

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Fortran90 function</th>
<th>C Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initialize a tag from a SEP tag.</td>
<td>init_sep3d</td>
<td>init_sep3d_tag</td>
</tr>
<tr>
<td>Initialize a tag from another structure.</td>
<td>init_sep3d</td>
<td>init_sep3d_struct</td>
</tr>
<tr>
<td>Initialize a tag from parameters.</td>
<td>init_sep3d</td>
<td></td>
</tr>
<tr>
<td>Delete all memory associated with the structure</td>
<td>sep3df_delete</td>
<td>sep3dc_delete</td>
</tr>
</tbody>
</table>

INTERACTION WITH DISK REPRESENTATION

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Fortran90 function</th>
<th>C Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read in all/some portion of the headers from a file.</td>
<td>sep3d_grab_headers</td>
<td>sep3dc_grab_headers</td>
</tr>
<tr>
<td>Read in the data associated with the current headers/grid in memory.</td>
<td>sep3d_read_data</td>
<td>sep3dc_read_data</td>
</tr>
<tr>
<td>Write a structure to disk.</td>
<td>sep3d_write_description</td>
<td>sep3dc_write_description</td>
</tr>
<tr>
<td>Tell which portions of a dataset should be written to disk</td>
<td>sep3d_write_status</td>
<td>sep3dc_write_status</td>
</tr>
<tr>
<td>Write the number of traces in the dataset to disk</td>
<td>sep3d_rite_num_traces</td>
<td>sep3dc_rite_num_traces</td>
</tr>
<tr>
<td>Write data, headers, and/or grid to a tag.</td>
<td>sep3d_write_data</td>
<td>sep3dc_write_data</td>
</tr>
</tbody>
</table>

INFORMATION ABOUT STRUCTURE

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Fortran90 function</th>
<th>C Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Print a description of the structure.</td>
<td>print_sep3d</td>
<td>print_sep3dc</td>
</tr>
<tr>
<td>Return the last data axis of length &gt; 1.</td>
<td>sep3d_ndims</td>
<td>sep3dc_ndims</td>
</tr>
<tr>
<td>Check to make sure the structure meets minimum criteria.</td>
<td>valid_structure</td>
<td>valid_structure</td>
</tr>
<tr>
<td>Check to see if two spaces share the same vector space.</td>
<td>sep3d_conform</td>
<td>sep3dc_conform</td>
</tr>
<tr>
<td>Check to see if space occupied by structure A is contained within structure B.</td>
<td>sep3d_ge_space</td>
<td>sep3dc_ge_space</td>
</tr>
<tr>
<td>Check to see if structure has a tdata_record_number.</td>
<td>sep3d_with_drn</td>
<td>sep3dc_with_drn</td>
</tr>
<tr>
<td>Return the axis associated with a given label.</td>
<td>sep3d_axis_index</td>
<td>sep3dc_axis_index</td>
</tr>
<tr>
<td>Return the key associated with a given name.</td>
<td>sep3d_key_index</td>
<td>sep3dc_key_index</td>
</tr>
</tbody>
</table>
INTERACTION WITH INTERNAL REPRESENTATION

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Fortran90 function</th>
<th>C Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synchronize structure with internal representation.</td>
<td>sep3d_grab_sep3d</td>
<td>sep3dc_grab_sep3d</td>
</tr>
<tr>
<td>Grab the header values currently in memory through its key name.</td>
<td>sep3d_grab_key_vals</td>
<td>sep3dc_grab_key_vals</td>
</tr>
<tr>
<td>Set the header values currently in memory through its internal key index.</td>
<td>sep3d_set_key_vals</td>
<td>sep3dc_set_key_vals</td>
</tr>
<tr>
<td>Grab the header values currently in memory through its internal key index.</td>
<td>sep3d_set_key_vals</td>
<td>sep3dc_set_key_vals</td>
</tr>
<tr>
<td>Grab grid values currently in memory.</td>
<td>sep3d_grab_grid_values</td>
<td>sep3dc_grab_grid_values</td>
</tr>
<tr>
<td>Store grid values currently in memory.</td>
<td>sep3d_set_grid_values</td>
<td>sep3dc_set_grid_values</td>
</tr>
<tr>
<td>Copy the grid values from one structure to another.</td>
<td>sep3d_grid_copy</td>
<td>sep3dc_grid_copy</td>
</tr>
<tr>
<td>Copy the header values from one structure to another.</td>
<td>sep3d_header_copy</td>
<td>sep3dc_header_copy</td>
</tr>
</tbody>
</table>

MODIFY A STRUCTURE

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Fortran90 function</th>
<th>C Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set the number of headers</td>
<td>sep3d_set_number_headers</td>
<td>sep3dc_set_number_headers</td>
</tr>
<tr>
<td>Specify that the headers and the data are in the same order</td>
<td>sep3d_inorder</td>
<td>sep3dc_inorder</td>
</tr>
<tr>
<td>Reshape the hyper cube.</td>
<td>sep3d_reshape</td>
<td>sep3dc_reshape</td>
</tr>
<tr>
<td>Allocate space for keys.</td>
<td>key_allocate</td>
<td>key_allocate</td>
</tr>
<tr>
<td>Allocate space for axes.</td>
<td>axis_allocate</td>
<td>axis_allocate</td>
</tr>
</tbody>
</table>

COMMAND LINE OPTIONS

In addition to the explicit library calls, there are several command line parameter options to change how individual datasets are handled. These can be broken into two classes: ones to support SU-SEP3D interoperability and ones to do simple data transforms on the fly.
SU support

As mentioned earlier in many ways the SU format is just a subset of SEP3D. This fact allows us to pass between SEP3D and SU program and libraries. To make this as easy as possible the library has numerous command line options.

\texttt{suinput=1} To specify that the input dataset is in SU format.

\texttt{suoutput=1} To specify that the output data is in SU format.

\texttt{nmem=X} To increase the maximum number of SU headers to \(X\) that are checked for non-zero values when reading in a SU dataset. By default set to 100.

\texttt{nextra_keys} The number of keys in SU format that are in non-standard location.

\texttt{extra_namei=N} The name of keys \(i\) (1..nextra_key) that are in non-standard SU locations.

\texttt{extra_typei=a} The type of values \(i\) (1..nextra_key) that are in non-standard SU locations.

\texttt{extra_offseti=N} The offset of the keys \(i\) (1..nextra_key) that are in non-standard SU locations.

\texttt{keyname=sepkeyname} The SEP3D keyname associated with a given SU keyname.

\texttt{keyname-fract=fract} What to scale to SEP3D key by when converting to SU. The reciprocal used for converting from SEP3D to SU.

Command line tricks

When using SEP3D or SU dataset there are often rather simple operations that you would like to do to the dataset. Some of the most common command line supported options:

\texttt{tag.ignore_gff=0} Set to 1 to ignore the grid format file in a SEP3D dataset.

\texttt{tag.ignore_hff=0} Set to 1 to ignore the header format file in a SEP3D dataset.

\texttt{tag-extra_keys=a:b:c} A list of SEP3D keys, separated by colons, that you would like to create on the fly for a given tag.

\texttt{tag-key-type=scalar_float} For keys specified in the above. Provide their type either tscalar_float or tscalar_int.

\texttt{tag-key-eqn=A} Specify a mathematical function to create a key. The expression should be enclosed in quotes. It should not include any spaces. Functions should be specified by t@COS (a full list can be found by doing tman Headermath), they can include axes description (e.g. taxis3) or keynames ts_x.
CONCLUSIONS

Irregular data can significantly increase coding complexity. The tsuperset library attempts to reduce the complexity by providing simple functions for common complex tasks. An added benefit of this approach is much greater interoperability between SEPlib and SU.

REFERENCES


APPENDIX A

! <
! read_geometry
!
! read in some data and extract some headers using SEPlib3d
!
! USAGE:
! read_geometry < in.H > out.H
! %end of self-documentation
!
PROGRAM read_geometry
!
! access the sep module
!
USE sep3d_struct_mod
use sep
IMPLICIT NONE
!
! define the data structure (do i know what i am doing here? Yep)
!
type (sep3d) :: input,output
!
! allocatable arrays
!
REAL, ALLOCATABLE, DIMENSION(:, :) :: d ! input seismic data
REAL, ALLOCATABLE, DIMENSION(:) :: xs, ys, zs ! source coordinates
REAL, ALLOCATABLE, DIMENSION(:) :: xg, yg, zg ! group coordinates
!
! simple variables
!
INTEGER :: nt ! number of input samples
INTEGER :: ntrace ! total number of input traces
INTEGER :: nh ! number of headers per trace
INTEGER :: maxmem ! memory available for storage in words.
INTEGER :: maxtr_per_block ! maximum number of traces in a block of data.
INTEGER :: ntr_per_block ! number of traces in current block of data.
INTEGER :: nblocks ! number of blocks of needed to read all the traces.
INTEGER :: jblock ! block index
INTEGER :: fwind ! first trace of the data window to be read.
!
! control variables
!
INTEGER :: allocation_status ! return code for Fortran90 allocate command
INTEGER :: stderr=0 ! standard error unit
logical :: grid_exists ! Whether or not grid exists

! initialize the input seismic data file.
! do i need to specify any arguments?

!INITIALIZE SEP PARAMETER HANDLING/SELF DOC
call sep_init(SOURCE)

!INITIALIZE SEP3D IO HANDLING
CALL init_3d()

! read in important parameters from the history or parameter file
call from_param("maxmem",maxmem,1000000)

!READ IN THE SEP3D STRUCTURE FROM INPUT TAG
call init_sep3d("in",input,"INPUT")

!CHECK TO SEE IF THE INPUT HAS AN HFF
if(input%file_format == "REGULAR") then
   call seperr("code will only work if headers exist")
else if(input%file_format="GRID") then
   grid_exists=.true.
else
   grid_exists=.false.
end if

!INITIALIZE THE OUTPUT
call init_sep3d(input,output,"OUTPUT")
call sep3d_set_inorder(output%tag) !SPECIFY THAT THE OUTPUT HEADERS SYNCHED WITH DATA
call sep3d_write_description("out",output)

!WE DON'T CARE ABOUT ADDITIONAL STRUCTURE IN THE HEADER OR GRID FILE INPUT
!THEREFORE JUST GO n2=n2*n3*n4*n... N3=3 n4=1 n_=1

!FOR NOW ONLY AVAILABLE IN C SO USE THE C TAG CORRESPONDING TO F90
call sep3d_change_dims(input%tag,2,(/1,input%ndims/))

!AND RESYNC THE F90
call sep3d_grab_sep3d(input%tag,input)

! FOR CONVENIENCE
nt=input%n(1)
ntrace=input%n(2)

! determine how many traces can fit into maxmem words of memory.
! determine how many blocks will need to be read in to read through
! all the input data.
!
maxtr_per_block=(maxmem-1)/(ntrace*nt)+1
nblocks=(ntrace-1)/maxtr_per_block+1
!
! ALLOCATE arrays
! note that output arrays are of length nt (the same as the input).
! arrays used in Fourier transform are of length nt_fft >= nt.
!
ALLOCATE(d(0:nt-1,maxtr_per_block), &
xs(maxtr_per_block),ys(maxtr_per_block),zs(maxtr_per_block), &
xg(maxtr_per_block),yg(maxtr_per_block),zg(maxtr_per_block), &
STAT=allocation_status)
IF(allocation_status /= 0) &
call seperr'space for d,xs,ys,zs,xg,yg,zg could not be allocated!'

!
! write out the history files defining the size of the output.
!
!
! loop over the blocks of data.
!
loop_over_blocks: DO jblock=1,nblocks
!
! calculate the number of traces in this block.
!
   ntr_per_block=maxtr_per_block
   IF(jblock == nblocks) ntr_per_block=ntrace-(nblocks-1)*maxtr_per_block
!
! grab the next ntr_per_block trace headers from the trace header file.
! 'in' = standard in?
! headers = a fortran90 structure
! nh = the number of headers in the structure.
! fwind = the sequential trace position in the trace header file.
! or is the area where the next position on the file will be
! mapped?
!
   fwind=(jblock-1)*maxtr_per_block
   CALL sep3d_grab_headers('in',input,nh, &
      fwind= (/fwind/), nwind= (/ntr_per_block/))
! echo out how many headers are out there on this file.
!
! FOR DATA WITHOUT A GRID nh will always = ntr_per_block
!
WRITE(stderr,*) 'number of headers read in = ', nh
!
! grab a subset of desired headers using the header key words.
!
CALL sep3d_grab_key_vals(input,"s_x",xs(1:nh))
CALL sep3d_grab_key_vals(input,"s_y",ys(1:nh))
CALL sep3d_grab_key_vals(input,"s_elev",zs(1:nh))
CALL sep3d_grab_key_vals(input,"g_x",xg(1:nh))
CALL sep3d_grab_key_vals(input,"g_y",yg(1:nh))
CALL sep3d_grab_key_vals(input,"g_elev",zg(1:nh))
!
! now let's try to read ntr_per_block traces of seismic data.
!
!WE ALWAYS READ IN THE DATA ASSOCIATED WITH PRE READ HEADERS
!SO IT ISN'T NECESSARY TO PUT WINDOWING PARAMETERS UNLESS WE
!WANT A SUBSECTION OF THE FIRST AXIS
IF(.not. sep3d_read_data('in',input,d(0:nt-1,l:ntr_per_block))) &
   CALL seperr('error in sep3d_read_data!')
!
! now lets do something simple. like add 2000m to the group and source!
elevations
!
   zs=zs+1000.; zg=zg+1000.; d=d*.5

! LETS SET THE OUTPUT UP
!FIRST COPY ALL OF THE HEADERS
call sep3d_copy_headers(input%tag,output%tag)
!THEN OUR NEW HEADERS
   CALL sep3d_set_key_vals(output,"s_elev",zs(1:nh))
   CALL sep3d_set_key_vals(output,"g_elev",zg(1:nh))
!NOW LETS WRITE OUT
if(.not. sep3d_write_data("out",output,d,write_headers=.true.,write_grid=grid_exists)) &
call seperr("trouble writing out data")

END DO loop_over_blocks
WRITE(stderr,"'normal completetion. routine read_geometry'

!CALL sep_close()
END PROGRAM read_geometry
Short Note

MPI in SEPlib

Robert G. Clapp

In the last 5 years, a dramatic shift has taken place in the way high-performance computation is done in the oil industry (Bednar et al., 1999; Clapp and Sava, 2002) and many other fields that have heavy computation requirements. No longer is a single computer with massive shared memory and multiple processors the economical choice. Instead numerous inexpensive PC based machines are clustered together for computation. This change in hardware paradigms requires a corresponding change in programing paradigms. No longer are shared memory programing models such as Open MP is sufficient. Instead a more distributed model such as Parallel Virtual Machine (PVM) or more commonly, Message Passing Interface (MPI) must be used.

A second challenge with this change in paradigm is related to the type of problems that we deal with in the oil industry. The volume of data makes network transfer a non-trivial portion of the processing flow and is the dominant element in some (Mosher, 1991). In this paper, I describe the routines in SEPlib to facilitate MPI processing and discuss some future possible directions.

**MPI WITH SEPLIB FILES**

For many applications, a very few routines are all the MPI that is needed. The basic idea of these routines is to make a local version of global files. These files are transfered to and from the master process through some simple routines. These routines can be broken into three categories: initialization, distribution, and collection. All of the routines are written in C with a Fortran interface (e.g. floats become reals, ints become integers).

**Initializers/Destructors**

`mpi_sep_send_args(int nproc, int max_mem, int verb)` - Transfer parameters from the master node to all the slave nodes. Only issued by the master node.

- **nproc** - Number of processes in the MPI job.
- **max_mem** - Maximum memory to use when transferring data.
- **verb** - Whether or not to be verbose when transferring.

---

1email: bob@sep.stanford.edu
**mpi_sep_receive_args()** - Receive arguments from the master node. MPI standard does not require command line arguments to be passed to slave processes. This is SEPlib’s solution to the problem.

**mpi_sep_clean()** - Deallocate the memory used for transferring.

**Distribution**

All of the values are in terms of 4 byte words and assume that the file has been moved (tsseek) to the correct position. Each function returns an integer, $t0$ if successful.

**mpi_sep_tag_bcast**(int $i$ _from, char *$t$ _from, int $i$, char *$t$ _to, int $bs$, int $nb$) - This routine makes local copies of a given tag.

- **$i$ _from** - The master node containing the file.
- **$t$ _from** - The tag of the sepfile on the master node.
- **$i$** - The thread number of the calling process.
- **$t$ _to** - The tag for the local version of the file.
- **$nb$, $bs$** - The size of the file $nb * bs$. Broken into parts to allow larger than 2GB.

**mpi_sep_tag_distribute**(int $i$ _from, char *$t$ _from, int $i$, char *$t$ _to, int $bs$, int $nb$, int *$s$ _to) - Distribute a tag to the various nodes.

- **$i$ _from** - The master node containing the file.
- **$t$ _from** - The tag of the sepfile on the master node.
- **$i$** - The thread number of the calling process.
- **$t$ _to** - The tag for the local version of the file.
- **$bs$** - The blocksize of the various dataset portions.
- **$nb$** - The number of blocks in the dataset.
- **$s$ _to** - An array of size $tnb$ telling what node should have each portion of the dataset.

**Collection**

All of the values are in terms of 4 byte words and assume that the file has been positioned (tsseek) to the correct position. Each function returns an integer, value 0, if successful.

**mpi_sep_tag_sum**(int $i$ _from, char *$t$ _from, int $i$ _to, char *$t$ _to, int $bs$, int $nb$, int $add$) - Sum the results of the various local versions into one file.

- **$i$ _from** - The local thread number.
- **$t$ _from** - The local tag name.
i_to - The thread where the image will be collected.

t_to - The tag for the global file.

bs,nb - The size of the file to combine \( nb \times bs \).

add - Whether (t1) or not (t0) to add the result to its current contents of the output tag.

**mpi_sep_tag_combine**(int i_from,char *t_from, int i_to, char *t_to, int bs, int nb, int *s_from,int add)
- Combine the local files into a single file.

i_from - The local thread number.

t_from - The local tag name.

i_to - The thread where the image will be collected.

t_to - The tag for the global file.

bs - The size of the blocks to combine.

nb - The number of blocks.

s_from - An array of size tnb listing which process has component of the global file.

add - Whether (t1) or not (t0) to add the result to its current values.

Appendix A contains a simple program illustrating how to use the library.

**AUTOMATIC PARALLELIZATION**

The library routines described earlier are effective in many situations. They become difficult to use when the distribution requirements become complex and inefficient when IO transfer times dominate processing times. Another set of routines is available when facing a task for which the first set of routines does not provide an effective solution.

The idea of this set of routines is twofold. First, a tag is distributed along some axis in blocks. This distribution can take two forms: sequential (tSEQUENTIAL) and combined (tCOMBINED). Figure 1 illustrates the two distribution patterns. The second principle is that there will be three thread types. The master thread will distribute and collect data. Each slave processor will then run two threads simultaneously. The first will simply be concerned with receiving and sending IO to the master node. The second will be processing the data. IO threads will notify its processing thread whenever it has finished reading in a block. The processing thread will then read in its local input, process the data and notify the IO thread that it is safe to transfer back the information to the master node. This will generally reduce the processing time from

\[
t_{tot} = 2t_i + t_o + t_{proc}
\]  

(1)

to

\[
t_{tot} = \min(t_i,t_o,t_{proc}) + t_{edge}
\]  

(2)

where \( t_{tot} \) is the total processing time, \( t_i \) is the time to send the input, \( t_o \) is the time to send the output, \( t_{edge} \) is the time to transfer first input and last output block, and \( t_{proc} \) is the total
processing time. This methodology is most effective if the master thread can handle higher bandwidth than the slave threads.

The routines to do MPI in this manner can be broken into four portions. First, constructors and destructors, which begin and end the MPI processing and describe how the data is distributed. The second batch of functions provide information to the programmer about what a given thread will be doing. The final two groups are specific to processing and IO threads.

![Figure 1: The different distribution patterns. The top-left panel shows a dataset distributed into 4 parts along the first axis (you would normally never distribute along the first axis but the argument holds true for any axis distribution ≪ ndim). In the top-right panel note that sections A and C go to processor 1 while B and D go to processor 2. The bottom two panels show the two possible distribution patterns. In the combined distribution the blocks retain their normal relationship. In the sequential distribution the processed blocks are written sequentially.](bob4-dis)

**Constructors and destructors**

`sep_mpi_auto_io_init()` - Initialize the automatic IO routines for the program.

`sep_mpi_finish()` - Finish all of the MPI processes and clean up intermediate files.

`sep_mpi_distrib_in_tag(char *tag_in, char *io_tag, int iaxis, int nblock, char *distrib_type)`
- Distribute an input tag.
  
  **tag_in** - The SEP3D tag name for the dataset.
  
  **io_tag** - The history file associated with the dataset.
  
  **iaxis** - The axis to distribute along.
  
  **nblock** - The block size (in iaxis elements) for each distributed portion.
  
  **distrib_type** - What type of distribution pattern to use for the given tag.

`sep_mpi_distrib_out_tag(char *tag_in, char *io_tag, int iaxis, int nblock, char *distrib_type)`
- Distribute an output tag.
  
  **tag_in** - The SEP3D tag name for the dataset.
io_tag - The history file associated with the dataset.

iaxis - The axis to distribute along.

nblock - The block size (in iaxis elements) for each distributed portion.

distrib_type - What type of distribution pattern to use for the given tag.

Process information

nproc=sep_mpi_proc_type - Return the type of process for the tag.

    nproc=0 - The thread is the master for the process.
    +iproc - The thread is for processing.
    -proc - The thread is for IO (-1 * thread number is returned).

nport=sep_mpi_thread_portion(char *tag, int ithread) - Portion that the given thread must process.

    tag - The tag to get the information about.
    ithread - The thread number returned by tsep_mpi_proc_type.
    nport - The total number of elements in the distributed axis that this thread needs to process.

Processing thread routines

sep_mpi_proc_begin(char *intag, char *outtag) - Get the local tag names for the input and output.

    intag - The local tag name for the input.
    outtag - The local tag name for the output.

sep_mpi_begin_io(char *tag, int nblock) - Return when IO can be begun on a given block.

    tag - The local tag name.
    nblock - The block that I need to have transfered before beginning IO.

sep_mpi_update_block(char *tag, int nblock, int input) - Record that the current block has finished processing.

    tag - The local tag name.
    nblock - The block that has been finished.
    input - Whether this is as input [1] or output tag.
Transferring thread routines

**sep_mpi_distrib_all(char *tag)** - Distribute a tag to slave processes.

  *tag* - Tag to distribute from the master node.

**sep_mpi_collect_all(char *tag)** - Collect a tag from the slave processes.

  *tag* - Tag to collect on the master thread.

Appendix B contains an example program using this IO framework.

**FUTURE WORK**

There are several directions to extend this work. First, there is no reason to limit the sending and receiving of data to regular datasets. The library is based on the tsuperset library (Clapp, 2003) so extending it to work on irregular data is straightforward. Second, many operations are a series of parallel operations. An intelligent extension would be to introduce the concept of a dataset spread over many machines. The user would choose whether or not to perform the collection operation or leave local versions. Most of this mechanism is already available in SEPlib (Clapp, 1999). Another extension is to incorporate some of the ideas mentioned in Clapp and Sava (2002). Specifically, with numerous users and diverse needs some type of dynamic allocation and sharing ability is needed. This would require each block to be considered a task and a check would have to be performed before executing the next section.

**CONCLUSIONS**

In this paper, I summarize some of the routines available in SEPlib for MPI processing. The routines break into two categories. The first is a collection of routines that simulate MPI operations of SEPlib files. The second set allows for some degree of automatic parallelization.

**REFERENCES**


APPENDIX A

program reduce_it{
use reshape_mod
use sep
use mpi_sep
integer :: i,ndo,mpi_status,ithread,nthread
integer :: ierr,my_status,i,local
integer, allocatable :: isend(:)
type(sep3d) :: input,output
real, pointer :: array(:,:)
character(len=256) :: intag,outtag
character(len=256) :: localin,localout
integer,external :: mpi_sep_tag_distribute,mpi_sep_tag_collect

call MPI_INIT(ierr) !INITIALIZE MPI
call set_no_putch() !DON’T WRITE TO THE HISTORY FILE
call sep_init( ) !SETUP SEP IO
!INITIALIZE AUTO IO

call MPI_COMM_SIZE(MPI_COMM_WORLD, nthread,ierr)
call MPI_COMM_RANK(MPI_COMM_WORLD, ithread,ierr)
if(ithread==0) then !IF MASTER THREAD
call from_param("verb",verb,.false.) !BE VERBOSE
call from_param("mem",mem,10) !USE 10 MB
iverb=0;if(verb) iverb=1

call MPI_SEP_SEND_ARGS(nthread,mem,iverb) !SEND ARGUMENTS
else !SLAVE THREAD

call MPI_SEP_RECEIVE_ARGS() !RECEIVE THREAD
end if
intag="intag";outtag="outtag" !GLOBAL INPUT/OUTPUT TAGS
write(localin,"(a,i2)") "intag",i !CREATE LOCAL INPUT TAG
write(localout,"(a,i2)") "outtag",i !CREATE LOCAL OUTPUT TAG

call init_sep3d(intag,input,"INPUT") !READ IN DATASET PARAMETERS
!MASTER TAG WRITES OUT TO OUTTAG
if(ithread==0){
call set_yes_putch();
call set_alternate_putch(outtag)
}
!FIGURE OUT WHERE TO SEND STUFF
allocate(isend(input%n(3)))
ito=1; nlocal=0
do i=1,size(isend){
isend(i)=ito;
if(ito==i) nlocal+=1
if(inext==n) ito=1
else ito+=1
}

!DISTRIBUTE THE INPUT
call auxout(localin)
if(0/=mpi_sep_tag_distribute(0,intag,ithread,localin,input%n(1)*input%n(2),&
   input%n(3), isend)) call seperr("trouble distributing data ");
call auxclose(localin); call auxin(localin)
allocate(array(input%n(1),input%n(2))) !ALLOCATE DATA

!COLLECT THE OUTPUT
if(0/=mpi_sep_tag_collect(0,outtag,ithread,localout,input%n(1)*input%n(2),&
   input%n(3), isend)) call seperr("trouble collecting output ");
call MPI_FINALIZE(ierr) !CLEAN UP
}
program reduce_it{
  use reshape_mod
  use sep
  use mpi_sep
  integer :: n(3), ns(3), no(4)
  integer :: i, ndo, mpi_status
  integer :: ierr, my_status
  real :: o(3), os(3), d(3), ds(3), oo(4), do(4)
  type(sep3d) :: input, output
  real, pointer :: array(:, :)
  character(len=256) :: intag, outtag
  integer, external :: sep_mpi_proc_type, sep_mpi_auto_io_init
  integer, external :: sep_mpi_distrib_all, sep_mpi_collect_all
  integer, external :: sep_mpi_distrib_in_tag, sep_mpi_distrib_out_tag
  integer, external :: sep_mpi_in_out_tags, sep_mpi_thread_portion
  call MPI_INIT(ierr) ! INITIALIZE MPI
  call set_no_putch() ! DON'T WRITE TO THE HISTORY FILE
  call sep_init("") ! SETUP SEP IO
  ! INITIALIZE AUTO IO
  if(0/=sep_mpi_auto_io_init()) call seperr("trouble initializing MPI IO ")
  my_status=sep_mpi_proc_type() ! GET THREAD NUMBER/TYP
  intag="intag"; outtag="outtag" ! GLOBAL INPUT/OUTPUT TAGS
  call init_sep3d(intag, input,"INPUT") ! READ IN DATASET PARAMETERS
  ! MASTER TAG WRITES OUT TO OUTTAG
  if(my_status==0){
    call set_yes_putch();
    call set_alternate_putch(outtag)
  }
  ! DISTRIBUTE THE INPUT ALONG THE THIRD AXIS SIZE 1
  if(0/=sep_mpi_distrib_in_tag(input%tag, intag, 3, 1, "SEQUENTIAL"))&
    call seperr("trouble distributing in tag")
  ! DISTRIBUTE THE OUTPUT ALONG THE THIRD AXIS SIZE 1
  if(0/=sep_mpi_distrib_out_tag(output%tag, outtag, 3, 1, "SEQUENTIAL"))&
    call seperr("trouble distributing out tag")
  if( my_status <=0){ ! IO/MASTER TAG
    if(0/=sep_mpi_distrib_all(input%tag)) ! DISTRIBUTE DATA
      call seperr("trouble distributing input")
    if(0/=sep_mpi_collect_all(output%tag)) & ! COLLECT DATA
      call seperr("trouble collecting output")
  }else{ ! PROCESSING TAG
    ! GET THE AMOUNT OF AXIS 3 I AM PROCESSING
    ndo=sep_mpi_thread_portion(intag, abs(my_status)-1)
if(ndo<0) call seperr("trouble grabbing my_thread_portion")
call sep_mpi_proc_begin(intag,outtag) !GET THE LOCAL TAG NAMES
allocate(array(input%n(1),input%n(2))) !ALLOCATE DATA
do i=1,ndo{
    call sreed(intag,array,size(array)*4) !READ IN DATA FROM LOCAL TAG
    array=array*4 !SCALE DATA
    call srite(outtag,array,size(array)*4) !WRITE OUT DATA TO LOCAL TAG
    call sep_mpi_update_block(outtag,i-1,0)!MARK BLOCK COMPLETED
}
call sep_mpi_finish() !CLEAN UP
SEP ARTICLES PUBLISHED OR IN PRESS


Sava, P., and Fomel, S., 2003, Angle-domain common image gathers by wavefield continuation methods: accepted for publication in Geophysics.


Vlad, I., 2002a, Enhanced attenuation of spatially uncorrelated noise: Eos Trans. AGU.

Vlad, I., 2002b, Reproducibility in computer-intensive sciences: Ad Astra, 1(2), accepted for publication.
SEP PHONE DIRECTORY

<table>
<thead>
<tr>
<th>Name</th>
<th>Phone</th>
<th>Login Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alvarez, Gabriel</td>
<td>724-0461</td>
<td>gabriel</td>
</tr>
<tr>
<td>Artman, Brad</td>
<td>723-6007</td>
<td>brad</td>
</tr>
<tr>
<td>Berryman, James</td>
<td>723-1250</td>
<td>berryman</td>
</tr>
<tr>
<td>Biondi, Biondo</td>
<td>723-1319</td>
<td>biondo</td>
</tr>
<tr>
<td>Brown, Morgan</td>
<td>723-6006</td>
<td>morgan</td>
</tr>
<tr>
<td>Chen, Weitian</td>
<td>725-1334</td>
<td>chen</td>
</tr>
<tr>
<td>Claerbout, Jon</td>
<td>723-3717</td>
<td>jon</td>
</tr>
<tr>
<td>Clapp, Marie</td>
<td>724-0461</td>
<td>marie</td>
</tr>
<tr>
<td>Clapp, Robert</td>
<td>724-0461</td>
<td>bob</td>
</tr>
<tr>
<td>Curry, William</td>
<td>723-1250</td>
<td>bill</td>
</tr>
<tr>
<td>Guitton, Antoine</td>
<td>723-6007</td>
<td>antoine</td>
</tr>
<tr>
<td>Lau, Diane</td>
<td>723-1703</td>
<td>diane</td>
</tr>
<tr>
<td>Lomask, Jesse</td>
<td>724-0461</td>
<td>lomask</td>
</tr>
<tr>
<td>Rakshe, Tejas</td>
<td>723-1250</td>
<td>rakshe</td>
</tr>
<tr>
<td>Rosales, Daniel</td>
<td>725-1334</td>
<td>daniel</td>
</tr>
<tr>
<td>Sava, Paul</td>
<td>723-0463</td>
<td>paul</td>
</tr>
<tr>
<td>Shan, Guojian</td>
<td>723-0463</td>
<td>shan</td>
</tr>
<tr>
<td>Shragge, Jeff</td>
<td>723-4417</td>
<td>jeff</td>
</tr>
<tr>
<td>Tisserant, Thomas</td>
<td>723-1250</td>
<td>thomas</td>
</tr>
<tr>
<td>Valenciano, Alejandro</td>
<td>723-6006</td>
<td>valencia</td>
</tr>
<tr>
<td>Vlad, Ioan</td>
<td>723-5911</td>
<td>nick</td>
</tr>
</tbody>
</table>

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.
<table>
<thead>
<tr>
<th>Name</th>
<th>Company</th>
<th>Telephone</th>
<th>E-Mail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raymond Abma</td>
<td>BP</td>
<td>(281) 366-4604</td>
<td><a href="mailto:abmar1@bp.com">abmar1@bp.com</a></td>
</tr>
<tr>
<td>Francois Audebert</td>
<td>CGG</td>
<td>(281) 646-2524</td>
<td><a href="mailto:faudebert@cgg.com">faudebert@cgg.com</a></td>
</tr>
<tr>
<td>Dimitri Bevc</td>
<td>3DGeo</td>
<td>(650) 969-3886</td>
<td><a href="mailto:dimitri@3dgeo.com">dimitri@3dgeo.com</a></td>
</tr>
<tr>
<td></td>
<td>(Co-chair, 1st year)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Biondo Biondi</td>
<td>SEP</td>
<td>(650) 723-1319</td>
<td><a href="mailto:biondo@sep.stanford.edu">biondo@sep.stanford.edu</a></td>
</tr>
<tr>
<td>Luis Canales</td>
<td>WesternGeco</td>
<td>(713) 689-5717</td>
<td><a href="mailto:lcanales@houston.westerngeco.slb.com">lcanales@houston.westerngeco.slb.com</a></td>
</tr>
<tr>
<td>Jon Claerbout</td>
<td>SEP</td>
<td>(650) 723-3717</td>
<td><a href="mailto:jon@sep.stanford.edu">jon@sep.stanford.edu</a></td>
</tr>
<tr>
<td>Richard Cook</td>
<td>Shell</td>
<td>(713) 245-7195</td>
<td><a href="mailto:richard.cook@shell.com">richard.cook@shell.com</a></td>
</tr>
<tr>
<td></td>
<td>(Co-chair, 2nd year)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Helmut Jakubowicz</td>
<td>Veritas</td>
<td>(44) 1293 443219</td>
<td><a href="mailto:helmut_jakubowicz@veritasdgc.com">helmut_jakubowicz@veritasdgc.com</a></td>
</tr>
<tr>
<td>Stewart Levin</td>
<td>Landmark Graphics</td>
<td>(303) 779-8080</td>
<td><a href="mailto:salevin@lgc.com">salevin@lgc.com</a></td>
</tr>
<tr>
<td>Simon Spitz</td>
<td>CGG</td>
<td>(281) 646-2502</td>
<td><a href="mailto:sspitz@cgg.com">sspitz@cgg.com</a></td>
</tr>
</tbody>
</table>
Research Personnel

Gabriel Alvarez received a B.S. degree in Physics from the National University of Colombia in 1985 and an M.Sc. in Geophysics from the Colorado School of Mines in 1995, where he was a member of the Center for Wave Phenomena. From 1989 to 2000, he worked at the Instituto Colombiano del Petroleo (ICP), the Research and Development Division of Ecopetrol, the National Oil Company of Colombia. Joined SEP in 2000 and is currently working towards a Ph.D. in geophysics at Stanford University.

Brad Artman received his B.Sc. in Geophysical Engineering from the Colorado School of Mines in December 1996. He worked at Shell Deepwater Development Company in New Orleans in petrophysical and geophysical capacities until joining SEP in the fall of 2000 to work toward a Ph.D. Brad is member of SEG and SPWLA.

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 sound waves in rocks – 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. Continuing research interests include seismic and electrical methods of geophysical imaging and waves in porous media containing fluids. He is a member of ASA, AGU, APS, 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.

**Morgan Brown** received a B.A. in Computational and Applied Mathematics from Rice University in 1997 and is currently completing a Ph.D. at SEP. Morgan worked as a research intern with Western Geophysical in 1997 and with Landmark Graphics in 2000. He is a member of SEG.

**Weitian Chen** received a B.A. in geophysics from Peking University in 1998, and obtained a M.Sc. in geophysics from Peking University in 2001. He is currently working towards a Ph.D in geophysics at Stanford, and is a member of SEG.

Marie Clapp, formerly Marie Prucha, received her B.Sc. in Geophysical Engineering from Colorado School of Mines in May 1997. She joined SEP in September 1997 and received her MS in June 1999. She married one of her fellow SEPers in 2001 and finally changed her last name in the summer of 2002. She is currently edging towards a Ph.D. in geophysics. She is a member of SEG.

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.
William Curry is a second year student with SEP, who graduated in 2000 from the University of Alberta with a B.Sc. (Hons.) in Geophysics. From 1998-1999, he worked as a seismic interpreter with Jet Energy Corp. in Calgary, Canada. Bill is a student member of the AGU, CGU, CSEG, EAGE and SEG.

Antoine Guitton received a MSc in geophysics from Universite de Strasbourg, France in 1996 and from Stanford University in 2000. Received a ”Diplome d’ingenieur de L’Ecole de Physique du Globe de Strasbourg” in 1996. He received the Best Student Paper Award from the SEG in 1999. Assistant research geophysicist at the Institut Francais du Petrole (Paris-1996/97) working on well seismic imaging. Assistant research geophysicist at CGG Houston (1997-98) working on multiples attenuation. He joined SEP in September 1998. Current research topics are nonlinear inversion and noise attenuation. He is member of the SEG.

Seth Haines received a B.A. (Hons) in Geology and Physics (double major) from Middlebury College in 1997, and began graduate work in the Stanford Geophysics Department in 1998 with a project involving seismic reflection and refraction profiling of the Tibetan crust. He received his M.Sc. in 2001, and is currently working on applying the electroseismic method to geophysical exploration problems as he continues toward a Ph.D. He is a member of AGU, GSA, and EEGS.
**Jesse Lomask** graduated in 1993 with a B.S. in Geology from Temple University in Philadelphia. He then worked as a field engineer for Anadrill-Schlumberger in the Gulf of Mexico for two years. In 1998, he completed a Masters in Exploration and Development at Stanford which included a summer internship at Mobil Exploration and Producing in Houston. He then worked as a staff geophysicist for Occidental Oil and Gas at Elk Hills, California where he interpreted some of a recently acquired 3D seismic survey. In the fall of 2001, he returned to Stanford to join SEP.

**Paul Sava** graduated in June 1995 from the University of Bucharest, with an Engineering Degree in Geophysics. Between 1995 and 1997 he was employed by Schlumberger GeoQuest. He joined SEP in 1997, received his M.Sc. in 1998, and continues his work toward a Ph.D. in Geophysics. His main research interest is in seismic imaging using wave-equation techniques. He is a member of SEG.

**Guojian Shan** received his B.Sc in Mathematics school of Peking University in July, 1998. From 1998 to 2001, he was studying in Institute of computational mathematics and scientific/Engineering Computing, Chinese Academy of Sciences (CAS), and got his M.S in Applied mathematics in July, 2001. He joined SEP in 2001 and now he is currently working toward a Ph.D. in geophysics. He is a student member of SEG.
Jeff Shragge graduated in 1998 with a BScH in Honours Physics from Queen's University in Kingston, Canada. After completing a MSc degree in 2001 in teleseismic imaging at the UBC in Vancouver, Canada, he spent 2001 and 2002 working for a small geophysical consulting company based out of Woburn, MA. He joined SEP in 2002, and is working towards a Ph.D. in Geophysics. His main research interest is migration and wavefield inversion. He is a member of SEG and AGU.

William W. Symes received degrees in Mathematics from the University of California at Berkeley and Harvard University, and has held faculty and research positions at the University of British Columbia, the University of Wisconsin, and Michigan State University. Since 1983 he has been Professor of Computational and Applied Mathematics at Rice University, where he also directs The Rice Inversion Project, an industry-university consortium project on mathematical and computational problems in seismic imaging.

Thomas Tisserant received a “Diplôme d’ingénieur de l’Ecole de Physique du Globe de Strasbourg” in 2002. He joined SEP in September 2002 and is working toward a M.S. in Geophysics. He is a member of SEG.
Alejandro A. Valenciano received a B. Sc. degree in Physics from Havana University (Cuba) in 1994, and a MSc 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 the SEP to work toward a Ph.D in Geophysics in the fall of 2001.

Ioan Vlad graduated in June 2000 with a Engineer Diploma (5-year degree) in Geophysics from the University of Bucharest, with a thesis on gravity and geodynamical modeling of the lithosphere. He joined SEP in 2000 and is currently working towards a Ph.D. in geophysics at Stanford. He is a member of SEG.
SPONSORS OF THE STANFORD EXPLORATION PROJECT, 2002-2003

Amerada Hess Corporation
One Allen Center
500 Dallas St.
Houston, TX 77002
USA
tel: (713) 609-5829
fax: (713) 609-5666
contact: Scott A. Morton
email: morton@hess.com

ChevronTexaco Corporation
6001 Bollinger Canyon Rd.
D-1058
San Ramon, CA 94583
USA
tel: (925) 842-6266
fax: (925) 842-2061
contact: Lin Zhang
email: lizh@chevronexaco.com

Aramco Services Company
Saudi Aramco
Geophysical R&D Division
P.O. Box 750
Dhahran 31311
Saudi Arabia
tel: 966 (3) 874 7262
fax: 966 (3) 873 1020
contact: Mohammed N. Alfaraj
email: farajmn@aramco.com.sa

Chinese Petroleum Corporation
Geophysical Data Processing Center
Taiwan Petroleum Exploration Division
14th Fl., No. 3, Sungren Road
Shinyi Chiu
Taipei, Taiwan 11010
tel: 886-2-87259910
fax: 886-2-87899060
contact: Jenyang Lin
email: jenyang@cpc.com.tw

BP America Inc.
200 Westlake Park Blvd.
WL 1018
Houston, TX 77079
USA
tel: (281) 366-3611
fax: (281) 366-5856
contact: John T. Etgen
email: etgenjt@bp.com

ConocoPhillips Inc.
Seismic Processing Development
540G Plaza Office Bldg.
Bartlesville, OK 74004
USA
tel: (918) 661-1514
fax: (918) 662-1587
contact: Robert M. Habiger
email: rmhabig@ppco.com

CGG Americas, Inc.
16430 Park Ten Place
Houston, TX 77084
USA
tel: (281) 646-2400
fax: (281) 646-2640
contact: Simon Spitz
email:sspitz@cgg.com

Ecopetrol-ICP
Laboratorio de Geofisica
A.A. 4185 Bucaramanga
Colombia
tel: (57) 76 445420
fax: (57) 76 445444
contact: Alfredo Tada
email: atada@ecopetrol.com.co
SPONSORS OF THE STANFORD EXPLORATION PROJECT, 2002-2003

ENI SPA - AGIP Division
Dept. RIGE
via Unione Europea 3
20097 S. Donato Milanese
Italy
tel: 39 (02) 520 55308
tax: 39 (02) 520 45694
contact: Vittorio De Tomasi
demail: vittorio.detomasi@agip.it
JGI, Inc.
Meikei Building
1-5-21 Otsuka, Bunkyo-ku
Tokyo, 112-0012
Japan
tel: 81 (3) 5978 8043
tax: 81 (3) 5978 8060
contact: Yoichi Ohta
demail: yohta@jgi.co.jp

ExxonMobil Upstream Research Co.
Seismic Processing Research
3319 Mercer St., ST-401
P.O. Box 2189
Houston, TX 77027
USA
tel: (713) 431-6011
tax: (713) 431-6326
contact: Thomas A. Dickens
demail: tom.a.dickens@exxonmobil.com
Landmark Graphics Corporation
1805 Shea Center Dr.
Suite 400
Highlands Ranch, CO 80129-2258
USA
tel: (303) 779-8080
tax: (303) 796-0807
contact: Stewart A. Levin
demail: salevin@lgc.com

4th Wave Imaging Corporation
16A Journey
Suite 200
Aliso Viejo, CA 92656
USA
tel: (949) 916-9787
tax: (949) 916-9786
contact: David E. Lumley
demail: david.lumley@4thwaveimaging.com
Norsk Hydro ASA
PB 7190
Sandsliveien 90
N-5001 Bergen
Norway
tel: 47 (55) 99 6861
tax: 47 (55) 99 6970
contact: Per Riste
demail: per.riste@hydro.com

GX Technology Corporation
5847 San Felipe
Suite 3500
Houston, TX 77057-3010
USA
tel: (713) 789-7250
tax: (713) 789-7201
contact: Nick Bernitsas
demail: nxb@gxt.com
Paradigm Geophysical Corporation
1200 Smith St.
Suite 2100
Houston, TX 77002
USA
tel: (713) 393-4979
tax: (713) 393-4901
contact: Orhan Yilmaz
demail: yilmaz@paradigmgeo.com
SPONSORS OF THE STANFORD EXPLORATION PROJECT, 2002-2003

Petrobras S.A.
Av. Chile 65, sala 1302
Rio de Janeiro
20035-900 RJ
Brazil
tel: 55 (21) 2534 2706
fax: 55 (21) 2534 1076
contact: Carlos A. Cunha Filho
eemail: s002@ep.petrobras.com.br

3DGeo Development Inc.
465 Fairchild Dr.
Suite 227
Mountain View, CA 94043
USA
tel: (650) 969-3886
fax: (650) 969-6422
contact: Dimitri Bevc
eemail: dimitri@3dgeo.com

PGS Research
10550 Richmond Ave.
Suite 250
Houston, TX 77042
USA
tel: (713) 735-6322
fax: (713) 532-6774
contact: Ruben D. Martinez
eemail: ruben.martinez@pgs.com

TotalFinaElf Exploration
800 Gestner
Suite 700
Houston, TX 77024
USA
tel: 
fax: 
contact: Henri Calandra
eemail: henri.calandra@totalfinaelf.com

Shell Oil Company
P.O. Box 481
Houston, TX 77001-0481
USA
tel: (713) 245-7195
fax: (713) 245-7339
contact: Richard Cook
eemail: richardcook@shell.com

Unocal Corporation
14141 Southwest Frwy
Sugar Land, TX 77478
USA
tel: (281) 287-7481
fax: (281) 287-5360
contact: Philip S. Schultz
eemail: phil.schultz@unocal.com

Statoil Research Centre
Arkitekt Ebbellsvei 10
7005, Trondheim
Norway
tel: (47) 73584917
fax: (47) 73584665
contact: Constantin Gerea
eemail: geco@statoil.com

Veritas DGC Ltd.
Crompton Way
Manor Royal Estate
Crawley, West Sussex RH10 2QR
England
tel: 44 (1293) 443219
fax: 44 (1293) 443010
contact: Helmut Jakubowicz
eemail: helmut_jakubowicz@veritasdgc.com
SPONSORS OF THE STANFORD EXPLORATION PROJECT, 2002-2003

WesternGeco
10001 Richmond Ave.
Houston, TX 77042-4299
USA
tel: (713) 689-5717
fax: (713) 689-5757
contact: Luis L. Canales
email: lcanales@houston.westerngeco.slb.com