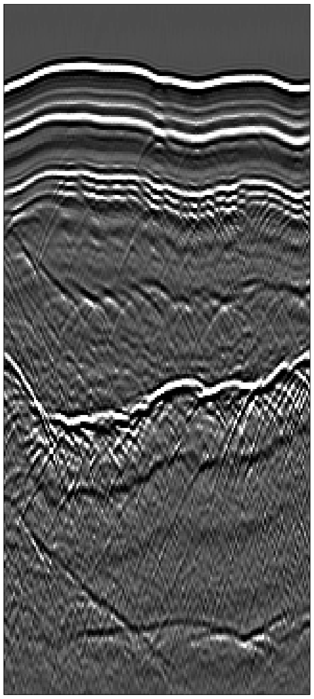
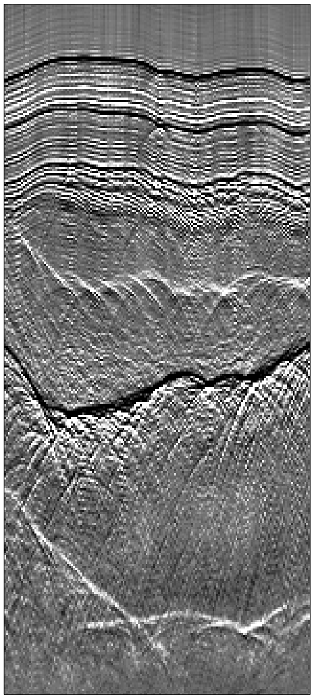
STANFORD EXPLORATION PROJECT

Ali Almomin, Gboyega Ayeni, Ohad Barak, Biondo Biondi, Jon Claerbout, Robert Clapp,
Qiang Fu, Adam Halpert, Yi Shen, and Yaxun Tang

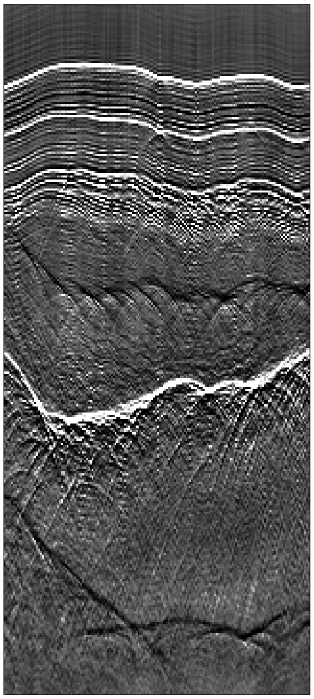
Report Number 145, September 2011



Input



PEF + Bidecon



GALI−PEF + Bidecon

Preface

The electronic version of this report[1] makes the included programs and applications available to the reader. The markings [ER], [CR], and [NR] are promises by the author about the reproducibility of each figure result. Reproducibility is a way of organizing computational research that allows both the author and the reader of a publication to verify the reported results. Reproducibility facilitates the transfer of knowledge within SEP and between SEP and its sponsors.

ER denotes Easily Reproducible and are the results of processing described in the paper. The author claims that you can reproduce such a figure from the programs, parameters, and makefiles included in the electronic document. The data must either be included in the electronic distribution, be easily available to all researchers (e.g., SEG-EAGE data sets), or be available in the SEP data library[2]. We assume you have a UNIX workstation with Fortran, Fortran90, C, X-Windows system and the software downloadable from our website (SEP makerules, SEPlib, and the SEP latex package), or other free software such as SU. Before the publication of the electronic document, someone other than the author tests the author's claim by destroying and rebuilding all ER figures. Some ER figures may not be reproducible by outsiders because they depend on data sets that are too large to distribute, or data that we do not have permission to redistribute but are in the SEP data library.

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

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

Our testing is currently limited to LINUX 2.6 (using the Intel Fortran90 compiler), but the code should be portable to other architectures. Reader's suggestions are welcome. More information on reproducing SEP's electronic documents is available online[3].

---

[1] http://sepwww.stanford.edu/private/docs/sep145

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

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

SEP145 — TABLE OF CONTENTS

Wave-equation migration velocity analysis

Simultaneous sources

Linearized wave-equation modeling/inversion

Object-oriented inversion

Bidirectional deconvolution

Two-parameters residual-moveout analysis for wave-equation migration velocity analysis

Biondo Biondi

ABSTRACT

The use of two-parameter RMO functions has the potential of improving the flatness of RMO-corrected gathers. The two RMO functions I propose add a second term to the conventional angle-domain RMO function. The proposed RMO functions achieve improved flatness when applied to two test CIGs that are representative of situations when either strong lateral velocity variations or anisotropy occur.

The use of two-parameter RMO functions could also improve the velocity gradients when applied within automatic MVA methods. My numerical experiments indicate that the RMO function that I defined by adding a term proportional to the fourth power of the tangent of the aperture angle should yield more accurate gradients than the one-parameter RMO function. This choice is also more robust with respect to the setting of processing parameters than the other two-parameter RMO function I introduce in the paper, which adds a term proportional to the absolute value of the sine of of the aperture angle

INTRODUCTION

The measurement of the residual moveout (RMO) in migrated common image gathers (CIG) is an important component of any migration velocity analysis (MVA) method. The choice of a robust method for measuring RMO is particularly important if the MVA process is automatized to avoid explicit picking of RMO parameters from coherency spectra (Biondi, 2008, 2010; Zhang and Biondi, 2011). In Biondi (2011), I illustrate with a simple example some of the challenges that these methods may encounter when a one-parameter RMO analysis is employed in presence of strong lateral velocity variations. To address these concerns, in this report I introduce two possible choices of RMO functions defined by two parameters instead of one. Both choices add a term to the RMO function, in addition to the usual term that is proportional to the square of the tangent of the aperture angle.

The first choice of RMO function adds a term proportional to the fourth power of the tangent of the aperture angle, and thus I will dub it the “Taylor” RMO function. The second choice adds a term proportional to the absolute value of the sine of the normalized aperture angle. The angle is normalized to enable the sine to complete a full cycle between zero and the maximum aperture angle used for the analysis. This choice is motivated by the fact that it is theoretically desirable to have the terms of the RMO function to be mutually orthogonal (Siliqi, 2009). The sine function is only approximately orthogonal to the square of the tangent, but it has the advantage of being extremely simple. I will dub this RMO function “Orthogonal”.

I test the efficacy of the proposed RMO functions using two CIGs that were obtained by migrating two different synthetic data sets. The first CIG represents the challenges presented by strong lateral velocity variations. The data were modeled assuming a strong negative velocity anomaly above a flat reflector (Biondi, 2011). The second CIG represents the effects of anisotropy on RMO analysis. The data were modeled assuming a strongly anisotropic VTI medium ($\epsilon = 0.0975$ and $\delta = -0.11$) above a flat reflector, and migrated assuming an isotropic velocity (Biondi, 2005).

In the following section I introduce the new RMO functions and apply them to compute two-dimensional spectra measuring the stack power as a function of the moveout parameters. In the subsequent section I analyze the accuracy of the potential search direction that would be computed by evaluating the gradient of the stack power as a function of the RMO parameters, and compare the results with the results of a similar analysis when the conventional one-parameter RMO analysis is applied.

TWO-PARAMETER RMO FUNCTIONS

Biondi and Symes (2004) introduced the following one-parameter RMO function for angle-domain CIG:

$$\Delta z = (1 - \rho) \tan^2 \gamma, \tag{1}$$

where γ is the aperture angle and Δz is the difference between the imaged depth at normal incidence ($\gamma = 0$) and the imaged depth at a given angle γ. For constant velocity errors in the half space above the reflector, the parameter ρ has a direct physical interpretation. It is related to the ratio between the current migration slowness s_{mig} and the true slowness s; that is, $\rho \approx s_{\text{mig}}/s$. However, in the following discussion this physical interpretation of ρ is irrelevant, and it can be simply considered as a free parameter describing a family of RMO functions.

What I call the Taylor RMO function adds the next higher-order even term to equation 1 as follows

$$\Delta z_T = (1 - \rho) \tan^2 \gamma + (1 - \lambda_T) \tan^4 \gamma, \tag{2}$$

where λ_T is the additional free parameter. As in equation 1, the RMO function is equally flat when $\rho = 1$ and $\lambda_T = 1$.

The second two-parameter RMO function that I introduce adds a sine function to equation 1 as follows

$$\Delta z_O = (1 - \rho) \tan^2 \gamma + (1 - \lambda_O) \, |\sin \bar{\gamma}|, \tag{3}$$

where λ_O is the additional free parameter, $\bar{\gamma} = 2\pi\gamma/\gamma_{\text{max}}$ is the normalized aperture angle, and γ_{max} is the maximum aperture angle used for the analysis.

Figure 1a shows the first CIG that I use for my study. It was obtained by migrating a synthetic data set that was modeled assuming a strong negative velocity anomaly above a flat reflector and migrated assuming a constant velocity (Biondi, 2011). This CIG is located under the center of the anomaly. Its moveout is not well described by the conventional RMO function expressed in equation 1 because the image at near angles is more affected by the anomaly than the image at far angles. Figure 1b shows the result of correcting this CIG using equation 1 with $\rho = 1.075$ that is the ρ value corresponding to the maximum of the stack power picked from a stack-power spectrum. This corrected CIG is far from being flat.

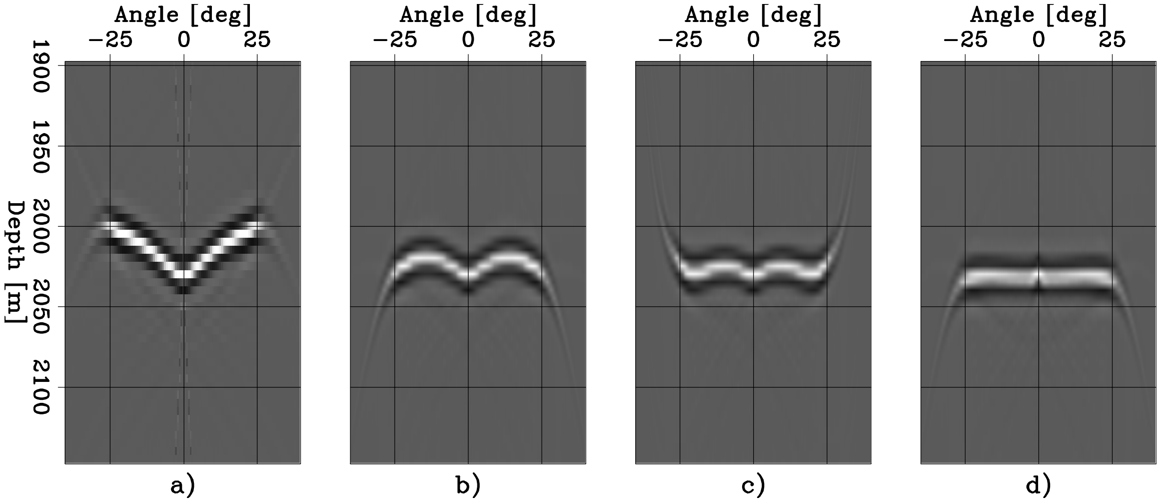


Figure 1: CIGs after constant velocity migration and: a) no correction, b) correction with a one-parameter RMO (equation 1), c) correction with the "Taylor" RMO (equation 2), and d) correction with the "Orthogonal" RMO (equation 3). [**CR**] biondo1/. Cig

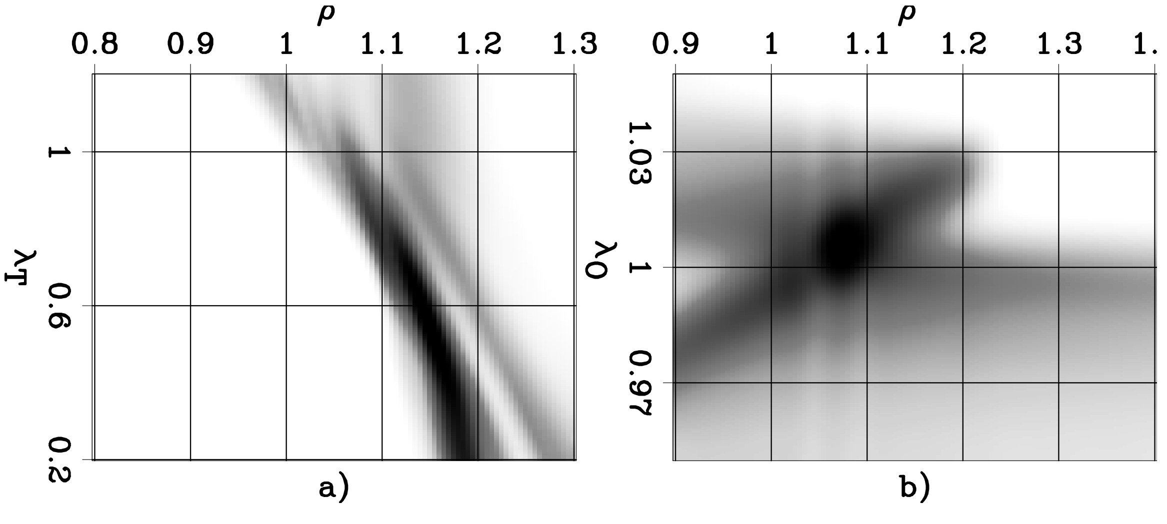


Figure 2: Two-parameter stack-power spectra resulting from RMO analysis of the migrated CIG shown in Figure 1a obtained applying: a) the "Taylor" RMO function (equation 2), and b) the "Orthogonal" RMO function (equation 3). [**CR**] biondo1/. Smooth

Figure 2 shows stack-power spectra as a function of two parameters. The panel on the left (Figure 2a) was computed using the "Taylor" RMO function described by equation 2, whereas the panel on the right (Figure 2b) was computed using the "Orthogonal" RMO function described by equation 3. In both cases the stack power was computed over the range of $-25^o \leq \gamma \leq 25^o$. Consequently, I set $\gamma_{\rm max} = 25^o$ to compute the normalized angle $\bar{\gamma}$ in equation 3. The power spectra were averaged over a thick (200 m) depth interval and slightly smoothed along the RMO parameters ρ and λ.

The maxima of both of these two-parameter spectra are not along the $\lambda = 1$ line, indicating that the two-parameter RMO improves the flatness with respect to the one-parameter RMO. Indeed, when the values corresponding to the maxima of the power spectra shown in Figure 2 are used to correct the original CIG I obtain flatter gathers than when using a one-parameter RMO. Figure 1c shows the result of correcting the CIG shown Figure 1a using equation 2 with $\rho = 1.15$ and $\lambda_T = .55$. Figure 1c shows the result of correcting the CIG shown Figure 1a using equation 3 with $\rho = 1.075$ and $\lambda_O = 1.0055$. The CIG corrected using the "Orthogonal" RMO is almost perfectly flat within the $-25^o \leq \gamma \leq 25^o$ range.

Notice that the shape of the spectra around their respective maxima is substantially different between the two plots. The function corresponding to the "Orthogonal" RMO is more isotropic around the maximum than the one corresponding to the "Taylor" RMO. This behavior is expected because the two terms of the "Orthogonal" RMO function are close to be orthogonal with respect to each other. Theoretically, this more isotropic shape could lead to better gradients. However, we can also notice diagonal artifacts in Figure 2b. As we discuss below, the effects of these artifacts tend to outweigh any advantage provided by the more isotropic shape of the spectrum.

Figure 3a shows the second CIG that I use for my analysis. It was obtained by migrating a synthetic data set that was modeled assuming a strongly anisotropic VTI medium ($\epsilon = 0.0975$ and $\delta = -0.11$) above a flat reflector, and migrated assuming an isotropic velocity (Biondi, 2005). Because the anisotropy in the medium is not taken into account by the isotropic migration, the CIG moveout is not well described by the conventional one-parameter RMO function expressed in equation 1. Figure 1b shows the result of correcting this CIG using equation 1 with $\rho = .9375$ that is the ρ value corresponding to the maximum of the stack power picked from a stack-power spectrum. This corrected CIG is far from being flat.

Figure 4 shows stack-power spectra as a function of two parameters. The panel on the left (Figure 4a) was computed using the "Taylor" RMO function described by equation 2, whereas the panel on the right (Figure 4b) was computed using the "Orthogonal" RMO function described by equation 3. In both cases the stack power was computed over the range of $-50^o \leq \gamma \leq 50^o$. Consequently, I set $\gamma_{\rm max} = 50^o$ to compute the normalized angle $\bar{\gamma}$ in equation 3. The power spectra were averaged over a thick (200 m) depth interval and slightly smoothed along the RMO parameters ρ and λ.

As for the spectra computed from the first CIG (Figure 2), the function corresponding to the "Orthogonal" RMO is more isotropic around the maximum than the one corresponding to the "Taylor" RMO. This difference in shape is less pronounced for this example than for the previous one.

Because the maxima of both of these two-parameter spectra are not along the $\lambda = 1$ line, we have indication that the two-parameter RMO improves the flatness with respect to the

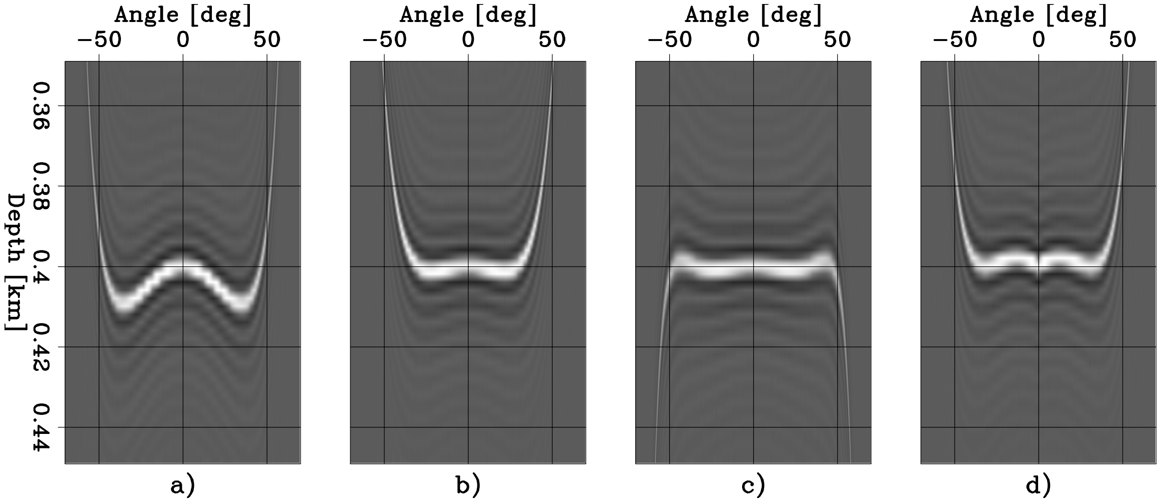


Figure 3: CIGs after isotropic velocity migration and: a) no correction, b) correction with a one-parameter RMO (equation 1), c) correction with the "Taylor" RMO (equation 2), and d) correction with the "Orthogonal" RMO shows the second CIG that I use (equation 3). [**CR**] biondo1/. Cig-aniso

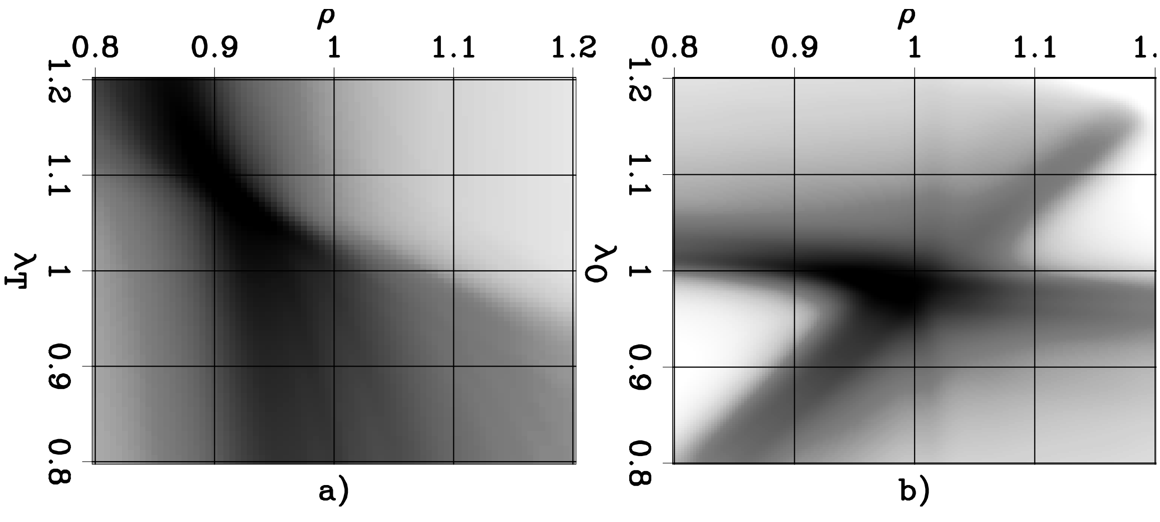


Figure 4: Two-parameter stack-power spectra resulting from RMO analysis of the migrated CIG shown in Figure 3a obtained applying: a) the "Taylor" RMO function (equation 2), and b) the "Orthogonal" RMO function (equation 3). [**CR**] biondo1/. Smooth-aniso

one-parameter RMO. Indeed, when the values corresponding to the maxima of the power spectra shown in Figure 4 are used to correct the original CIG I obtain flatter gathers than when using a one-parameter RMO. Figure 3c shows the result of correcting the CIG shown Figure 3a using equation 2 with $\rho = 0.915$ and $\lambda_T = 1.075$. Figure 3c shows the result of correcting the CIG shown Figure 3a using equation 3 with $\rho = 0.97$ and $\lambda_O = 0.988$. In particular, the CIG corrected using the "Taylor" RMO is significantly flatter, within the $-50^o \leq \gamma \leq 50^o$ range, than the one corrected using a one-parameter RMO.

CONVERGENCE ANALYSIS

In the previous section I showed that we can obtain flatter migrated CIGs by applying a two-parameter RMO correction instead of a conventional one-parameter correction. These results would be sufficient to motivate the use of a two-parameter RMO if the goal were to improve the signal-to-noise in the stacked cube, or to perform velocity analysis by picking the stack-power maxima. However, I am interested in using the new RMO functions in an MVA method that avoids picking the maxima of coherency spectra; this method computes the gradients of the objective function from the gradient of the stack-power spectra with respective to the RMO parameters (Biondi, 2008, 2010; Zhang and Biondi, 2011). It is therefore important to analyze the quality of the gradient information computed from two-parameter spectra, and compare to the corresponding gradient information computed from one-parameter spectra.

As a quality measurements of the gradient information, I compute the correlation across the angle axis between the RMO function that would be computed by picking the maxima of the coherency spectra ${}_p\Delta z$ and the RMO function ${}_g\Delta z$ computed using the gradient.

For the "Taylor" RMO the reference RMO function ${}_p\Delta z_T$ is computed as follows:

$$ {}_p\Delta z_T\left(\gamma,\rho,\lambda_T\right) = \left(\rho - \widehat{\rho}\right)\tan^2\gamma + \left(\lambda_T - \widehat{\lambda_T}\right)\tan^4\gamma. \tag{4} $$

where $\left(\widehat{\rho},\widehat{\lambda_T}\right)$ are the coordinates of the power-spectrum maximum. The RMO function ${}_g\Delta z_T$ computed from the gradient of the power spectrum P_T is,

$$ {}_g\Delta z_T\left(\gamma,\rho,\lambda_T\right) = -\frac{\partial P_T}{\partial \rho}\left(\rho,\lambda_T\right)\tan^2\gamma - \frac{\partial P_T}{\partial \lambda_T}\left(\rho,\lambda_T\right)\tan^4\gamma, \tag{5} $$

and the correlation is computed as

$$ C_T\left(\rho,\lambda_T\right) = \sum_{\gamma} {}_p\Delta z_T\left(\gamma,\rho,\lambda_T\right) {}_g\Delta z_T\left(\gamma,\rho,\lambda_T\right). \tag{6} $$

I compare this correlation function over a range of (ρ,λ_T) with the correlation function computed by

$$ C_{T1}\left(\rho,\lambda_T\right) = \sum_{\gamma} {}_p\Delta z_T\left(\gamma,\rho,\lambda_T\right) {}_g\Delta z_{T1}\left(\gamma,\rho,\lambda_T\right), \tag{7} $$

where the one-parameter RMO ${}_g\Delta z_{T1}$ is computed as follows

$$ {}_g\Delta z_{T1}\left(\gamma,\rho,\lambda_T\right) = -\frac{\partial P_T}{\partial \rho}\left(\rho,\lambda_T\right)\tan^2\gamma. \tag{8} $$

Figure 5 compares the correlation functions C_T (panel a) and C_{T1} (panel b) for the first CIG analyzed (Figure 1a). The asterisk superimposed onto the plots of the correlation functions is located at the maximum of the power spectrum displayed in Figure 2a. The coordinates $\left(\widehat{\rho}, \widehat{\lambda_T}\right)$ of this maximum are used to evaluate the moveout ${}_p\Delta z_T$ according to equation 4. Accurate gradient directions correspond to positive correlation (plotted in white in the figure), whereas potentially misleading gradient directions correspond to negative correlation (plotted in black in the figure).

The correlation functions are mostly positive over a wide range of parameters (ρ, λ_T), indicating that a velocity estimation method based on these RMO functions is likely to have good global convergence properties. In particular, the positive correlation functions at $(\rho = 1, \lambda_T = 1)$ indicates that the gradient computed starting from the migrated CIG shown in Figure 2a would be accurate, even if this CIG is far from being flat.

The correlation functions shown in Figure 5 are very similar. Therefore, the global convergence of the velocity estimation would be robust independently of whether the one-parameter or the two-parameter RMO function is used.

Similar correlation analysis of the RMO function can be performed when applying the "Orthogonal" RMO instead of the "Taylor" RMO. In this case the reference RMO function ${}_p\Delta z_O$ is computed as follows:

$$ {}_p\Delta z_O\left(\gamma, \rho, \lambda_O\right) = \left(\rho - \widehat{\rho}\right) \tan^2\gamma + \left(\lambda_O - \widehat{\lambda_O}\right) |\sin\bar{\gamma}|. \tag{9} $$

where $\left(\widehat{\rho}, \widehat{\lambda_O}\right)$ are the coordinates of the corresponding power-spectrum maximum. The RMO function ${}_g\Delta z_O$ computed from the gradient of the power spectrum P_O is,

$$ {}_g\Delta z_O\left(\gamma, \rho, \lambda_O\right) = -\frac{\partial P_O}{\partial \rho}\left(\rho, \lambda_O\right) \tan^2\gamma - \frac{\partial P_O}{\partial \lambda_T}\left(\rho, \lambda_O\right) |\sin\bar{\gamma}|, \tag{10} $$

and the correlation is computed as

$$ C_O\left(\rho, \lambda_O\right) = \sum_\gamma {}_p\Delta z_O\left(\gamma, \rho, \lambda_O\right) {}_g\Delta z_O\left(\gamma, \rho, \lambda_O\right). \tag{11} $$

This correlation is compared with the correlation

$$ C_{O1}\left(\rho, \lambda_O\right) = \sum_\gamma {}_p\Delta z_O\left(\gamma, \rho, \lambda_O\right) {}_g\Delta z_{O1}\left(\gamma, \rho, \lambda_O\right), \tag{12} $$

where the one-parameter RMO ${}_g\Delta z_{O1}$ is computed as follows

$$ {}_g\Delta z_{O1}\left(\gamma, \rho, \lambda_O\right) = -\frac{\partial P_O}{\partial \rho}\left(\rho, \lambda_O\right) \tan^2\gamma. \tag{13} $$

Figure 6 compares the correlation functions C_O (panel a) and C_{O1} (panel b) for the first CIG analyzed (Figure 1a). The asterisk superimposed onto the plots of the correlation functions is located at the maximum of the power spectrum displayed in Figure 2b. The coordinates $\left(\widehat{\rho}, \widehat{\lambda_O}\right)$ of this maximum are used to evaluate the moveout ${}_p\Delta z_O$ according to equation 9. As for the previous figure, accurate gradient directions correspond to positive correlation (plotted in white in the figure), whereas potentially misleading gradient directions correspond to negative correlation (plotted in black in the figure).

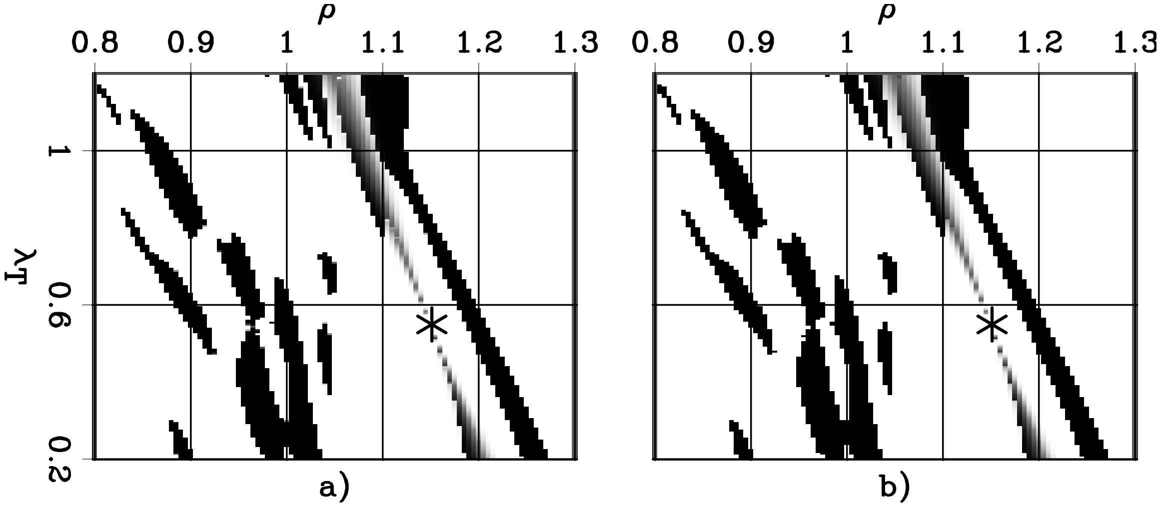


Figure 5: Correlation functions corresponding to the CIG shown in Figure 1a for: a) the "Taylor" two-parameter RMO function (equation 6), and b) the one-parameter RMO function (equation 7). [**CR**] biondo1/. CorrShift-TP

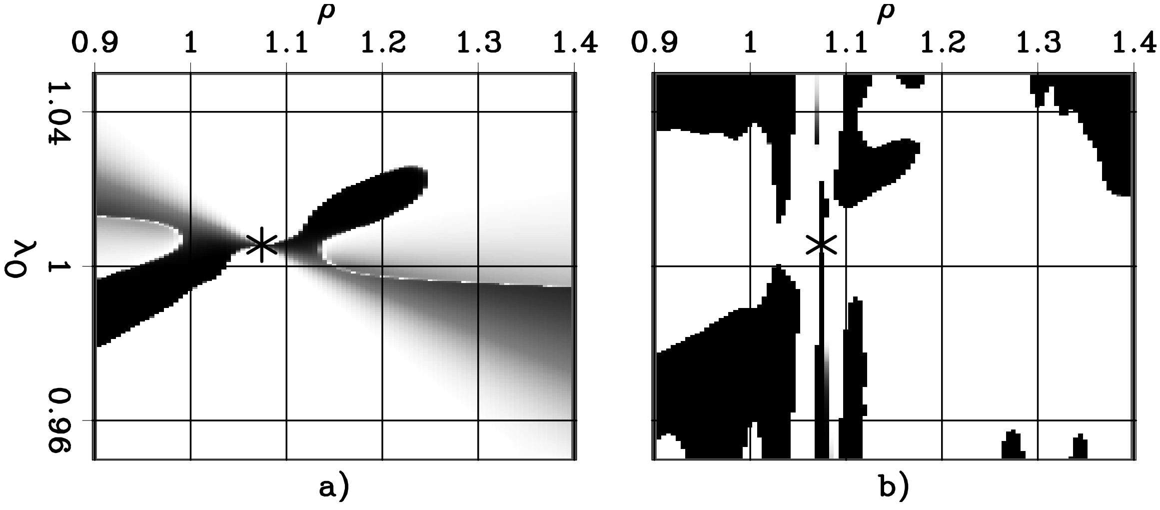


Figure 6: Correlation functions corresponding to the CIG shown in Figure 1a for: a) the "Orthogonal" two-parameter RMO function (equation 11), and b) the one-parameter RMO function (equation 12). [**CR**] biondo1/. CorrShift-OP

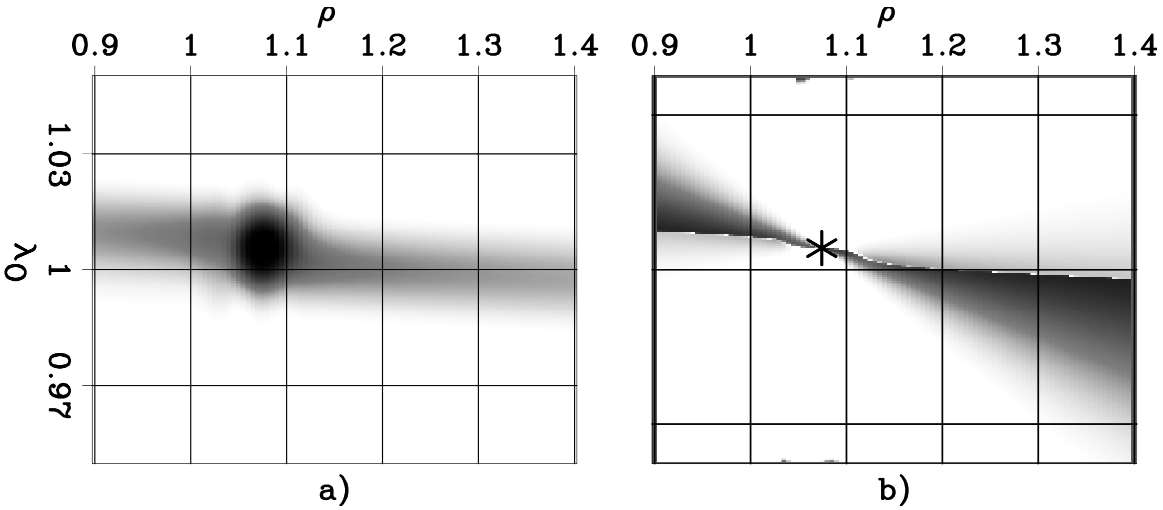


Figure 7: Panel a): Two-parameter stack-power spectra resulting from RMO analysis of the CIG shown in Figure 1a obtained using a thinner averaging window (30 m) than the one used to compute the spectrum shown in Figure 2b. Panel b): Correlation function for the "Orthogonal" two-parameter RMO function obtained using the thinner averaging window. [**CR**] biondo1/. CorrShift-OP-narrow

In this case the correlation functions shown in Figure 6 are not as similar as in the previous case. In particular, the black area around the value ($\rho = 1, \lambda_O = 1$) in Figure 6a indicate that the two-parameter RMO analysis would provide unreliable gradients. This problem is related to the diagonal artifacts visible in the power spectrum shown in in Figure 2a. These artifacts are caused by the fact that the second term in the "Orthogonal" RMO function has an extremum in the middle of the angular range, in contrast with the other RMO functions that have an extremum at normal-incidence. This mid-range extremum causes spurious local maxima of the spectrum at depths different than the normal incidence depth of the imaged reflector. These artifacts are much weaker when I averaged the power spectrum over a thinner depth interval (30 m) than the one used for computing the function displayed in Figure 6a. The new averaging window is of thickness comparable to the image of the reflector. Figure 7a shows the power spectrum obtained with this thinner averaging window, and Figure 7b corresponds to the two-parameters correlation function, which is a substantial improvement with respect to the one shown in Figure 6a. Figure 8 and Figure 9 shows the analysis of the correlation functions for the second CIG analyzed (Figure 3a); that is, the CIG suffering from the effects of anisotropy. Figure 8 corresponds to the "Taylor" RMO function, whereas Figure 9 corresponds to the "Orthogonal" RMO function. For this CIG, the two-parameter RMO analysis seems to improve the global convergence of the method, in particular when the "Orthogonal" function is applied (Figure 9).

Local convergence analysis

To gain an insight whether using a two-parameter RMO provides more accurate gradients starting from a CIG that is already close to be flat, we can zoom into the previous correlation plots around the location of the maximum. The correlation function in these

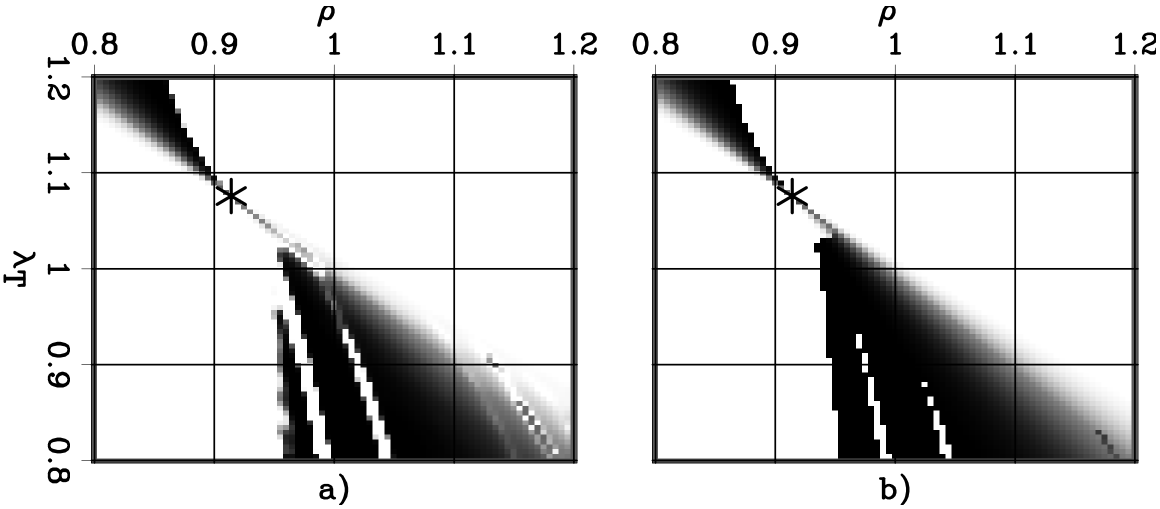


Figure 8: Correlation functions corresponding to the CIG shown in Figure 3a for: a) the "Taylor" two-parameter RMO function (equation 6), and b) the one-parameter RMO function (equation 7). [**CR**] biondo1/. CorrShift-TP-aniso

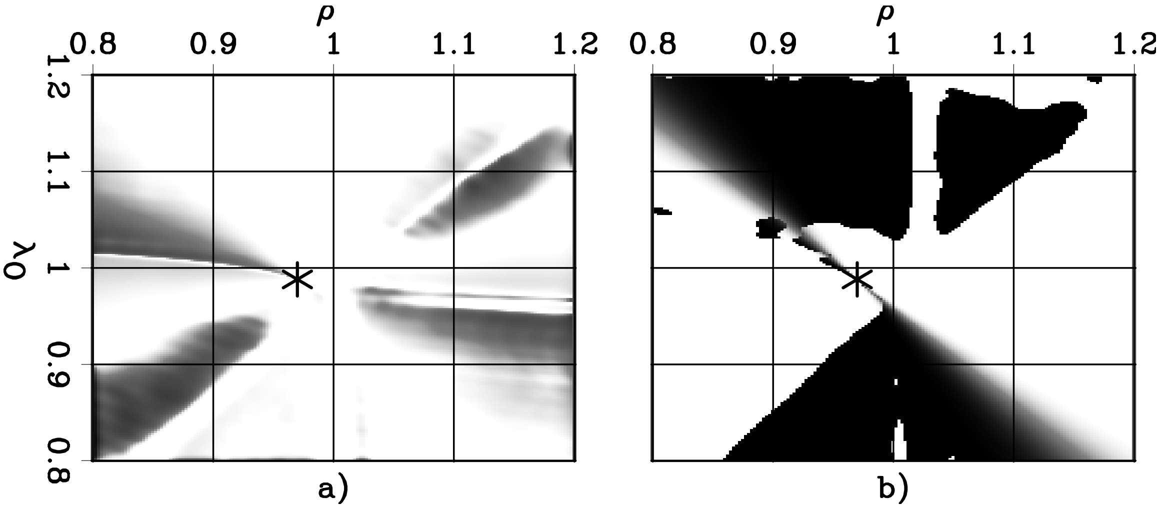


Figure 9: Correlation functions corresponding to the CIG shown in Figure 3a for: a) the "Orthogonal" two-parameter RMO function (equation 11), and b) the one-parameter RMO function (equation 12). [**CR**] biondo1/. CorrShift-OP-aniso

smaller windows is related to "local convergence" of velocity estimation methods based on the selected RMO functions.

Figure 10 shows the zooms into the plots shown in Figure 5; that is, comparing the correlation function for the first CIG (velocity anomaly) obtained using the "Taylor" RMO function (Figure 10a) and the conventional one-parameter RMO function (Figure 10b). I set the width of the close up windows to be equal to the distance of the maximum from the starting CIG; that is, the difference between the minimum and the maximum value of the ρ parameter is $\Delta\rho = (\widehat{\rho} - 1)$ and the difference between the minimum and the maximum value of the λ_T parameter is $\Delta\lambda_T = \left(\widehat{\lambda_T} - 1\right)$. The two plots shown in Figure 10 are almost identical, indicating that in this case there would be a negligible advantage to be gained by employing a two-parameter RMO function.

Figure 11 analyze the application of the "Orthogonal" RMO function to the same CIG as the previous figure. It shows the zooms into the plots shown in Figure 7b and Figure 6b, respectively. I set the window width following the same criterion described above for the "Taylor" RMO function. In this case, the one-parameter RMO function seems to provide better local convergence than the two-parameter RMO function.

The last two figures, Figure 12 and Figure 13, show similar analysis as the previous two, but applied to the CIG that suffers from the effects of anisotropy. For these CIG the two-parameters RMO function provides a better local convergence than the one-parameter function. The improvements look more substantial for the "Orthogonal" RMO function than for the "Taylor" RMO function.

DISCUSSION AND CONCLUSIONS

The introduction of a second term to the conventional RMO function for angle-domain CIGs improves the flatness of the corrected gathers for both CIGs I used in my testing and for both choices of two-parameter RMO function I proposed.

The answer to the question of whether using a two-parameter RMO function yields more reliable gradients when applied in automatic MVA methods is more ambiguous. The correlation analysis I presented indicates that the "Taylor" RMO function yields more robust gradients than the simple one-parameter RMO function for both CIGs I used as representative of situations when either strong lateral velocity variations or anisotropy occur. The impact of these improvements in real situation is difficult to predict. More testing and analysis are needed to determine whether the additional computation and code complexity introduced by the addition of a second term to the RMO function are worthy.

The "Orthogonal" RMO function may yields better gradients, but it is also more sensitive with respect to the thickness of the depth-averaging window for the power spectra. This fragility is caused by the location of the extremum of the second term of the "Orthogonal" RMO function in the middle of the angular range. Although the "Orthogonal" function has some theoretical advantages, its lack of robustness make it a less desirable choice.

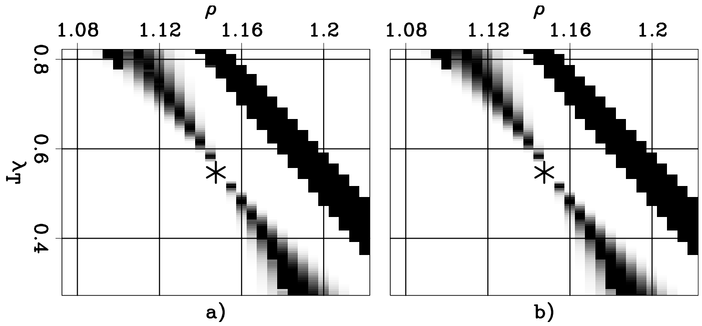


Figure 10: Zooms into the correlation functions shown in Figure 5. [**CR**] biondo1/. Zoom-CorrShift-TP

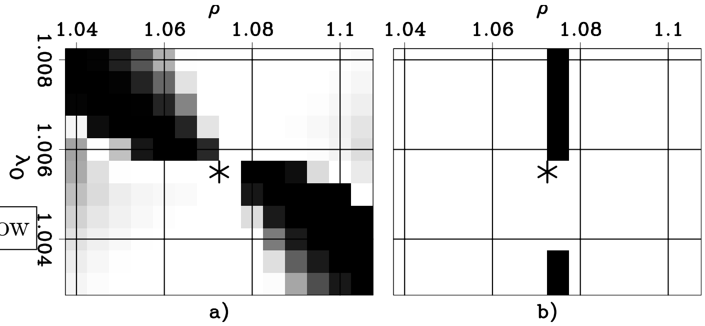


Figure 11: Zooms into the correlation functions shown in Figure 7b and Figure 6b, respectively. [**CR**] biondo1/. Zoom-CorrShift-OP-narrow

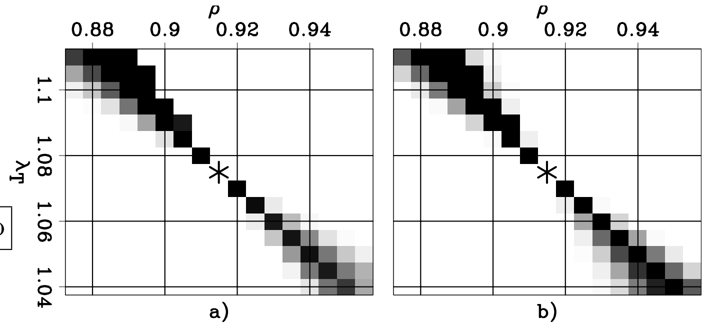


Figure 12: Zooms into the correlation functions shown in Figure 8. [**CR**] biondo1/. Zoom-CorrShift-TP-aniso

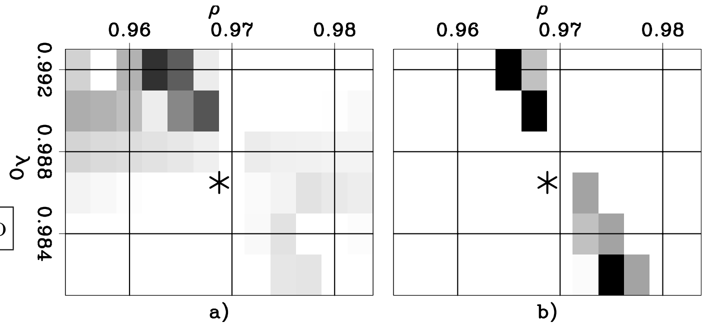


Figure 13: Zooms into the correlation functions shown in Figure 9. [**CR**] biondo1/. Zoom-CorrShift-OP-aniso

REFERENCES

Biondi, B., 2005, Angle-domain common image gathers for anisotropic migration: SEP-Report, **120**, 77–104.

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

———, 2010, Wave-equation migration velocity analysis by residual moveout fitting: SEP-Report, **142**, 25–32.

———, 2011, Migration velocity analysis by one-parameter residual moveout fitting in presence of strong lateral velocity anomalies: SEP-Report, **143**, 59–66.

Biondi, B. and W. W. Symes, 2004, Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging: Geophysics, **69**, 1283–1298.

Siliqi, R., 2009, Structurally coherent wide azimuth residual move out surfaces: SEG Technical Program Expanded Abstracts, **28**, 4039–4043.

Zhang, Y. and B. Biondi, 2011, Moveout-based wave-equation migration velocity analysis: SEP-Report, **143**, 43–58.

Velocity model evaluation through Born modeling and migration: a feasibility study

Adam Halpert and Yaxun Tang

ABSTRACT

A way to quickly test many possible migration velocity models would be a valuable interpretation tool. Here, a modified Born modeling scheme is used to simulate a new, smaller dataset from an initial image, allowing for target-oriented migrations in a fraction of the time needed for a full migration of the original dataset. Furthermore, the simulated dataset is migrated with a generalized source function derived from the original prestack image, preserving important velocity information that would be lost if a standard wavelet were used as the source function. While the method is currently limited to analysis of a single reflector, initial tests on a simple 2D synthetic model indicate that this method can accurately and efficiently produce images comparable to full standard migrations.

INTRODUCTION

Building an accurate seismic velocity model is essential for obtaining an acceptable image of the subsurface. When the subsurface is especially complex, for example in geological settings dominated by irregularly-shaped salt bodies, this task becomes particularly challenging. The large contrast between salt and sediment velocities magnifies the effects of inaccurate salt interpretation, resulting in a poor image. Unfortunately, velocity model-building is a time-consuming process that often requires several iterations. A typical salt-interpretation and model-building workflow involves iterative sediment- and salt-flood migrations to identify the top and base of the salt bodies (Mosher et al., 2007). In situations where the top or (especially) base salt interpretation is uncertain or ambiguous, several different salt scenarios may be geologically feasible. Therefore, a means of quickly testing the effects of several different possible velocity models would be extremely useful for judging and refining salt interpretations. Here, we investigate a Born modeling and migration scheme that allows for fast remigrations of data simulated from an initial image, that incorporates prestack velocity information from the initial image's subsurface offset gathers.

An interactive interpretation and imaging environment would be a valuable model-building tool, and several different approaches have been proposed. Wang et al. (2008) introduced a fast migration scheme built on Gaussian beam imaging (Hill, 1990), that can quickly test different salt scenarios. This method relies on seismic demigration and redatuming of wavefields to reduce the computational expense of migrating with several different velocity models. However, this method operates in the poststack image domain, neglecting velocity information that can be obtained from prestack images, and is limited by the approximations inherent to beam imaging. A similar approach has been proposed using reverse time migration (RTM) in a "layer-stripping" manner (Wang et al., 2011), but this remains too computationally intensive to test more than a very small number of possible models. Chauris and Benjemaa (2010) have proposed another method using RTM, which aims to reduce computational expense by summing over time-delays in the subsurface rather than sources. However, at present this method has only been demonstrated in two dimensions, and it remains unclear if an extension to 3D is feasible. Finally, fast migrations may also be achieved through the use of simulated datasets derived from an initial image. Guerra (2010) synthesized wavefields using prestack exploding reflector modeling as a means for performing wave-equation tomography in the image space. However, the significant amount of preprocessing required, especially in 3D, makes this approach less appealing for interactive testing of several velocity models.

Born modeling (Stolt and Benson, 1986) is based on a single-scattering approximation of the wave equation. By taking advantage of this approximation, we can simulate a new dataset (Tang and Biondi, 2010; Tang, 2011) from an initial image, whose size and acquisition geometry can be selected independently from those of the original dataset. Furthermore, the simulated data can be migrated using generalized sources, drastically reducing the number of shots required. In the examples we show, only a single shot is required, allowing for migrations well within an interactive time frame. In order to improve the accuracy of this method, we use a generalized source function derived from subsurface offset gathers of the initial image. This allows for a more accurate and data-driven result than if a simple wavelet were used as the source function; in addition, including non-zero subsurface offset information into this source function incorporates important velocity information available from the initial image.

In the following sections, we review the Born modeling methodology and outline the procedure for obtaining the generalized source function described above. We then demonstrate the method using simple 2D synthetic models. Crosstalk artifacts arising from the modeling procedure limit these examples to isolated image points along a single reflector in the subsurface; however, these tests indicate that this method can effectively provide information about the accuracy of different velocity models on an image. Further enhancements and the inclusion of phase-encoding (Romero et al., 2000) strategies should widen the applicability of the method. Ultimately, we hope to combine this method with elements of an automated image segmentation scheme (Halpert et al., 2011) to create a powerful tool for interactive interpretation and imaging.

METHOD

The goal of the procedure we will describe is to use Born modeling to synthesize a new dataset that is much smaller than the original dataset used to generate an initial migrated image. Since the synthesized data can be "recorded" at any location in x, y, and even z, this procedure is effectively target-oriented. There are three basic steps needed to reach our goal of efficient velocity model evaluation:

1. Generate an areal source function using one or more subsurface offset gathers from the initial prestack image.
2. Using the new source function and a reflectivity model based on the initial image, employ Born modeling to generate a new dataset with acquisition geometry best suited to image the target area.

3. Migrate the simulated data obtained in Step 2, using the source function from Step 1. This step is extremely computationally efficient compared to a full migration of the original data, allowing for testing of several possible velocity models in a fraction of the time it would take to evaluate them using standard migration techniques.

In the following sections we detail the theoretical basis for each of these steps.

Generalized source function

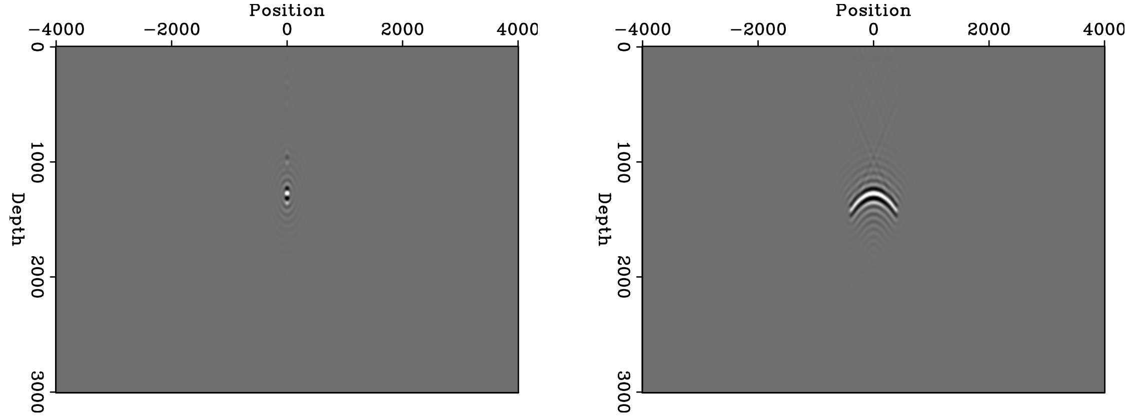
In conventional modeling and migration, a simple wavelet or plane wave is often used as the source function. However, here we can take advantage of the fact that the procedure described above begins with a migrated image. This allows us to perform post-stack "exploding reflector" (Claerbout, 2005) modeling of a reflector or point diffractor in the subsurface; the upward-continued wavefield can be recorded at any location, and then injected as an areal source function during Born modeling. Mathematically, this areal source is described as

$$S(\mathbf{x}_s) = \sum_{\mathbf{x}'} \sum_{\mathbf{h}} G^*(\mathbf{x}' - \mathbf{h}, \mathbf{x}_s, \omega, \xi(\mathbf{x}', \mathbf{h})), \tag{1}$$

where $\mathbf{x}_s = (x_s, y_s, z_s)$ are the arbitrarily defined locations where the wavefield will be recorded; $\mathbf{h}$ is the vector of subsurface half-offsets; ω is angular frequency; ξ is the location of the exploding image point in the subsurface; and G is the Green's function connecting the source to the image point (here, * denotes the adjoint). The Green's function is computed using the same velocity model that was used to image the originally-recorded data, meaning that the recorded wavefield should be independent of the original velocity model choice. However, since this velocity model is unlikely to be correct, the initial image should contain valuable information about the accuracy of this model in the form of subsurface offset gathers. Thus, the inclusion of the subsurface offset term $\mathbf{h}$ in equation 1 is designed to incorporate this information into the modeling. Since post-stack modeling is used to upward continue the wavefields, the non-zero subsurface offset data are mapped to equivalent zero subsurface offset locations:

$$\alpha(\mathbf{x} - \mathbf{h}) = \alpha(\mathbf{x} - \mathbf{h}) + \beta(\mathbf{x}, \mathbf{h}), \tag{2}$$

where α is the zero-offset data that are upward continued, and β is the original subsurface offset gather. To illustrate the advantage gained by incorporating this information, Figures 1(a) and 1(b) show two recorded source wavefields from an image point that is actually located at $z = 1000$ in the subsurface, but was initially imaged with a velocity that was 15% too slow. Both recorded wavefields have been reverse-propagated back to zero time to facilitate comparison. The source function in panel (a) was modeled using only the zero subsurface offset $h = 0$ data from the initial image, while the result in panel (b) uses the non-zero offset information as written in equation 1. When only zero subsurface offset data are used, the source appears to focus at the incorrect depth; when the nonzero offset data are used, the effects of using the wrong velocity are apparent. Using the source function in Figure 1(b) should therefore prove superior for use with the Born modeling and migration scheme described in the next section.



(a) (b)

Figure 1: Recorded source wavefields that have been reverse-propagated to zero-time; the result in (a) does not include information from the nonzero subsurface offsets of the initial image, while (b) does include this information. Both the initial migration and the modeling used a velocity model that was 15% too slow. adam1/. pt-0,pt-n0

Born modeling and migration

We now use the modeled areal source to generate a new data set via Born modeling. To do this, we define the simulated dataset d' recorded at arbitrary receiver locations $\mathbf{x}'_r$:

$$d'(\mathbf{x}'_r, \omega) = \sum_{\mathbf{x}'} \sum_{\mathbf{h}} \Gamma(\mathbf{x}_s, \mathbf{h}, \omega) G(\mathbf{x}' + \mathbf{h}, \mathbf{x}'_r, \omega) m(\mathbf{x}', \mathbf{h}). \tag{3}$$

Here, m is the reflectivity model (in our case, the initial image), and the Γ term is defined as

$$\Gamma(\mathbf{x}_s, \mathbf{h}, \omega) = \sum_{\mathbf{x}_s} S(\mathbf{x}_s) G(\mathbf{x}_s, \mathbf{x}' - \mathbf{h}, \omega), \tag{4}$$

where S is as defined in equation 1.

Because the placement of the receiver locations in equation 3 can be arbitrarily determined, they do not necessarily need to be on the surface, like the original recorded data. Placing the receivers at depth can improve the efficiency of this method by providing the capability for target-oriented imaging; if a velocity model is well-determined down to a given depth, the synthesized data can be recorded below that depth, avoiding unnecessary computation. This has a similar effect to re-datuming the wavefields, an approach taken by Wang et al. (2008) in their fast image updating strategy.

Now that we have new source and receiver wavefields, we can produce an image using standard wave-equation migration techniques:

$$m'(\mathbf{x}', \mathbf{h}) = \sum_{\omega} G^*(\mathbf{x}' - \mathbf{h}, \omega) \sum_{\mathbf{x}'_r} G^*(\mathbf{x}' + \mathbf{h}, \mathbf{x}'_r, \omega) d'(\mathbf{x}'_r, \omega). \tag{5}$$

It is important to note that the Green's functions in equation 5 can be computed using any velocity model, and not necessarily the same one used to generate the source and receiver wavefields in previous steps. This can allow for testing of multiple possible velocity models. Furthermore, since subsurface offset gathers are generated during the imaging, we now have a more quantitative means of judging the accuracy of these various models. This represents an advantage over methods such as beam migration that cannot provide this information. Unfortunately, the imaging procedure as written in equation 5 can also generate crosstalk artifacts, since areal source data is used. While various methods are available to help attenuate these artifacts (Romero et al., 2000; Tang, 2008), we restrict our examples in the next section to isolated points in the subsurface, spaced far enough apart to limit the effects of crosstalk.

EXAMPLES AND DISCUSSION

To test the feasibility of the method outlined above, we designed two simple synthetic test cases: a single flat reflector in the subsurface (Figure 2(a)), and a single reflector dipping at 20° (Figure 2(b)). We restricted these initial test cases to a single reflector in order to avoid problems related to crosstalk between multiple events or reflectors; while there are methods to attenuate this crosstalk, we hoped to gauge the feasibility of the method using only a single shot to migrate the Born-modeled data, without sophisticated phase-encoding schemes.

Both examples in Figure 2 were generated by migrating with an incorrect velocity model (15% slower than the constant-velocity model used to generate the original dataset). The effects of using an incorrect velocity can be seen clearly on the subsurface offset gather (non-focused event). A key goal of our Born modeling procedure is to replicate this behavior when the same velocity model is used to migrate the Born-modeled data. To test this, we sample isolated points from the reflectors in Figure 2, and use these points to generate the areal source function described in the previous section. In order to avoid unwelcome crosstalk between these points during the modeling process, they are separated by a distance that is twice the maximum subsurface offset, as seen in Figure 3.

Once the source function is “recorded,” Born modeling is performed using the subsampled images in Figure 3 as reflectivity models. The results of migrating this Born-modeled data, using the same velocity model used to produce the images in Figure 2, are seen in Figure 4. Because these images were migrated using an areal source function, only a single shot was necessary; this means that the images in Figure 4 were produced in seconds, nearly three orders of magnitude less time than was necessary to compute the images in Figure 2. Comparing the subsurface offset gathers for both of these figures, we see that while amplitudes differ, the kinematics have been accurately preserved in the Born-modeled result. If our goal is to evaluate the velocity model used, the quickly-obtained results in Figure 4 should be sufficient.

The importance of correctly spacing the image points we use for the modeling is illustrated in Figure 5. Here, points from the flat reflector image in Figure 2(a) have been sampled twice as frequently, at a spacing equal the maximum subsurface offset. Figure 5 shows the result of using these points to create the areal source function, and then performing Born modeling and migration as before. Now, crosstalk between the closely-spaced image points results in severe artifacts, including spurious events on the zero-subsurface offset image.

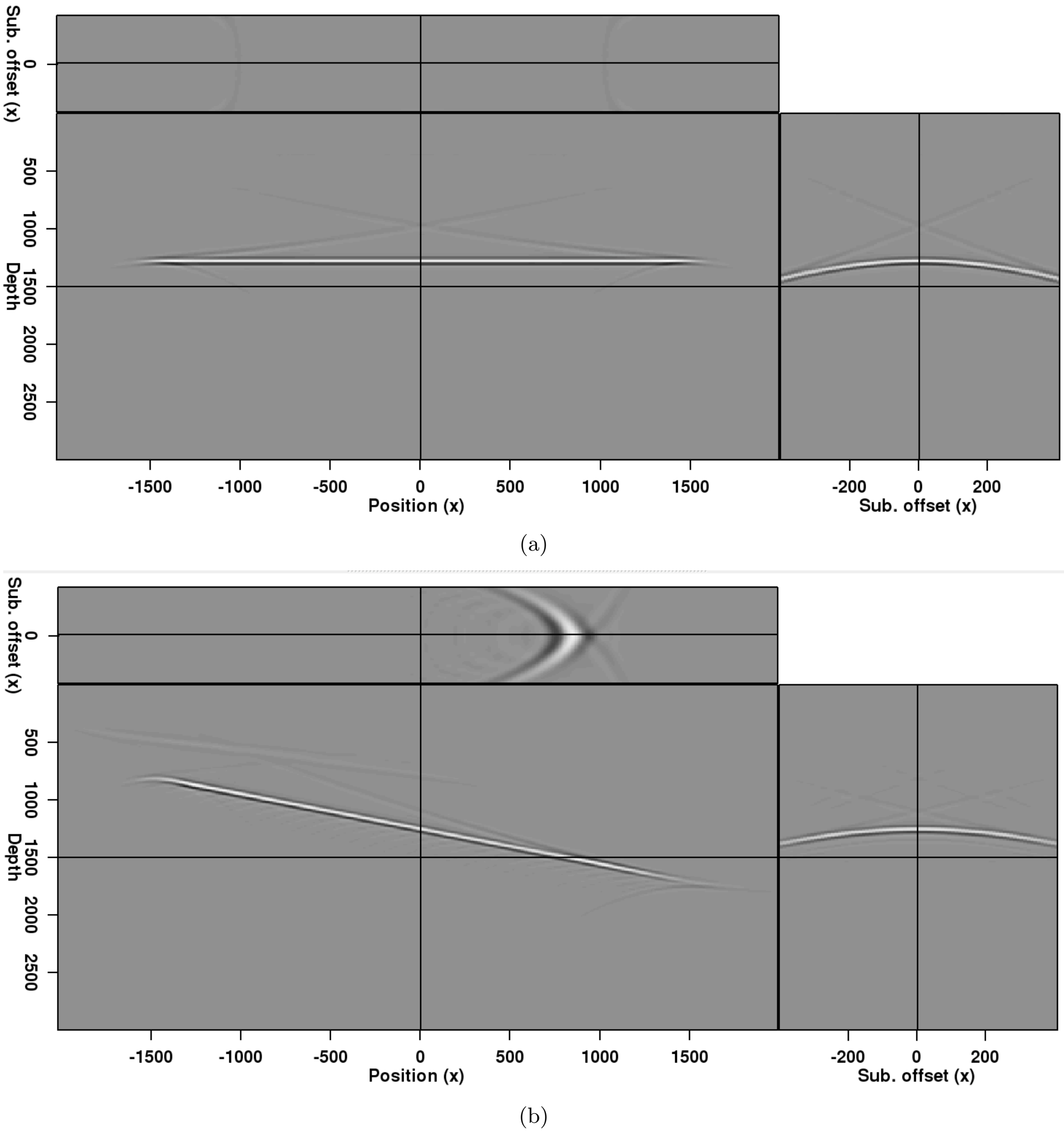
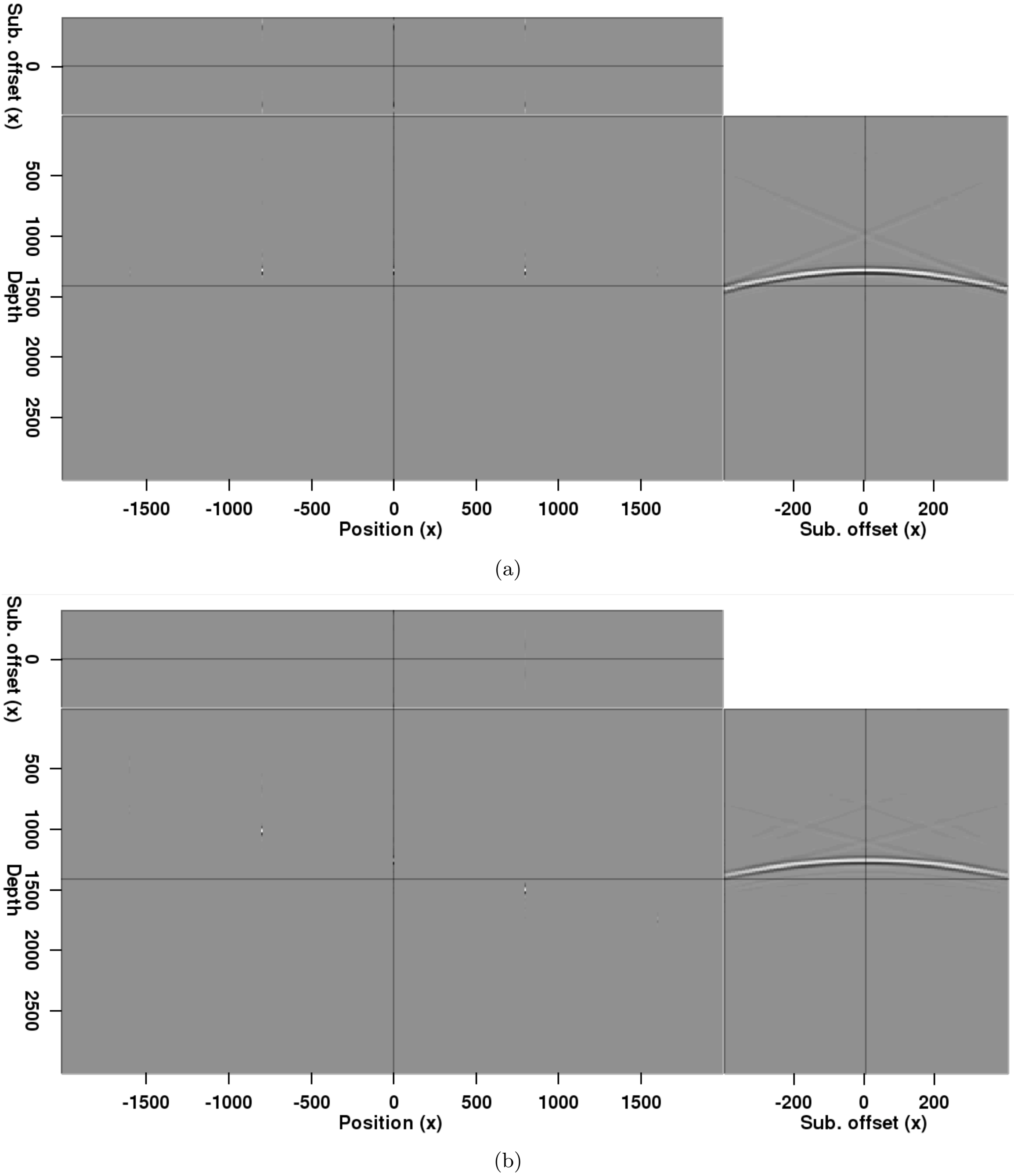


Figure 2: Prestack depth migration images of (a) a flat reflector and (b) a reflector dipping at a 20° angle. The images were migrated with a constant velocity 15% too slow compared to the true velocity, causing the noticeable artifacts and lack of focusing in the subsurface offset dimension. adam1/. flat-orig,dip-orig



(a)

(b)

Figure 3: Isolated image points from Figures 2(a) and 2(b) used for the modeling procedure. The points are separated by twice the maximum subsurface offset value in order to avoid crosstalk artifacts in the modeling. adam1/. flat-sp,dip-sp

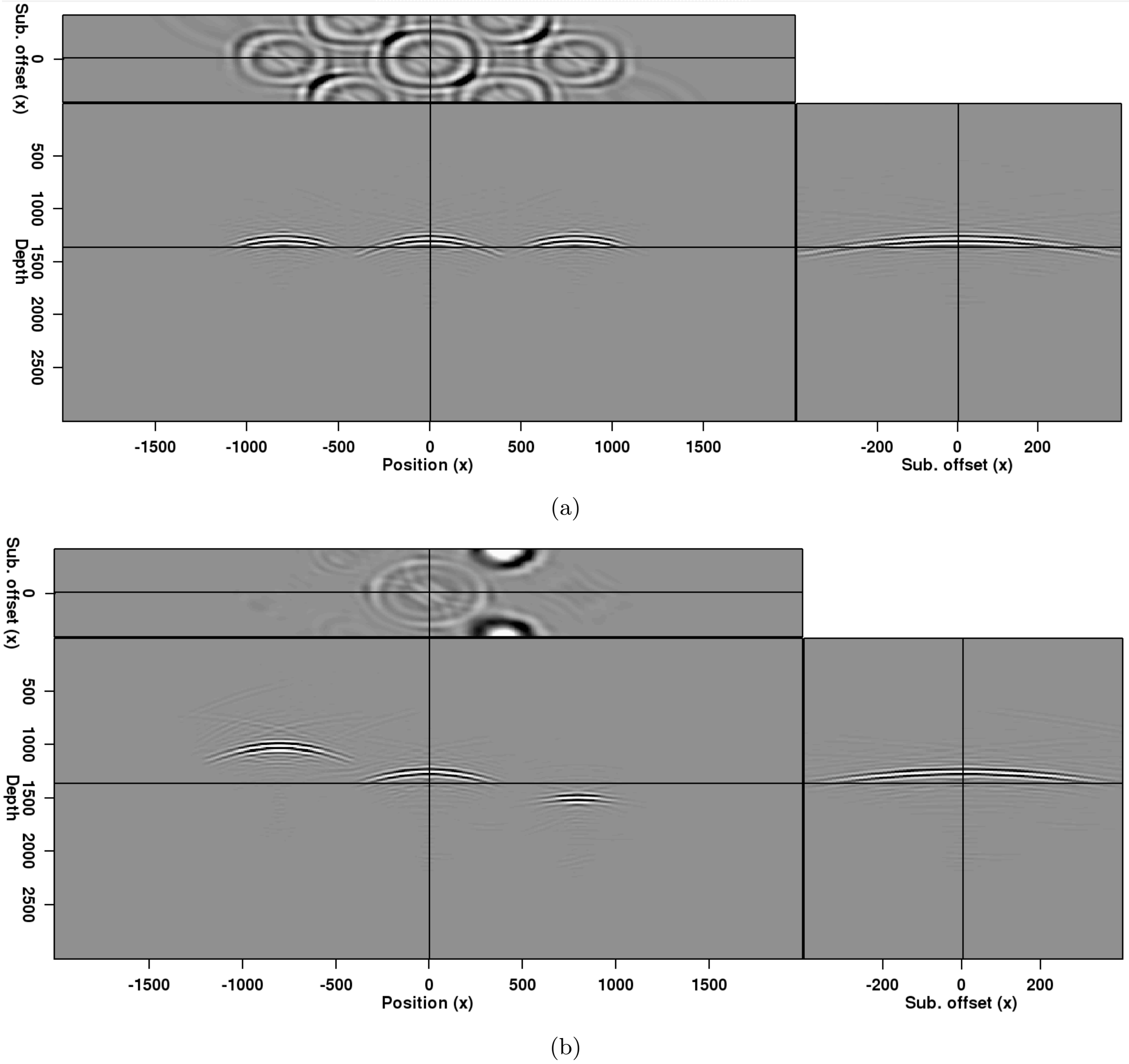


Figure 4: Migrated images after Born modeling using the images in Figure 3 as the reflectivity model. Although the amplitudes differ, the kinematics of the events in both figures match. adam1/. flat-sp-born,dip-sp-born

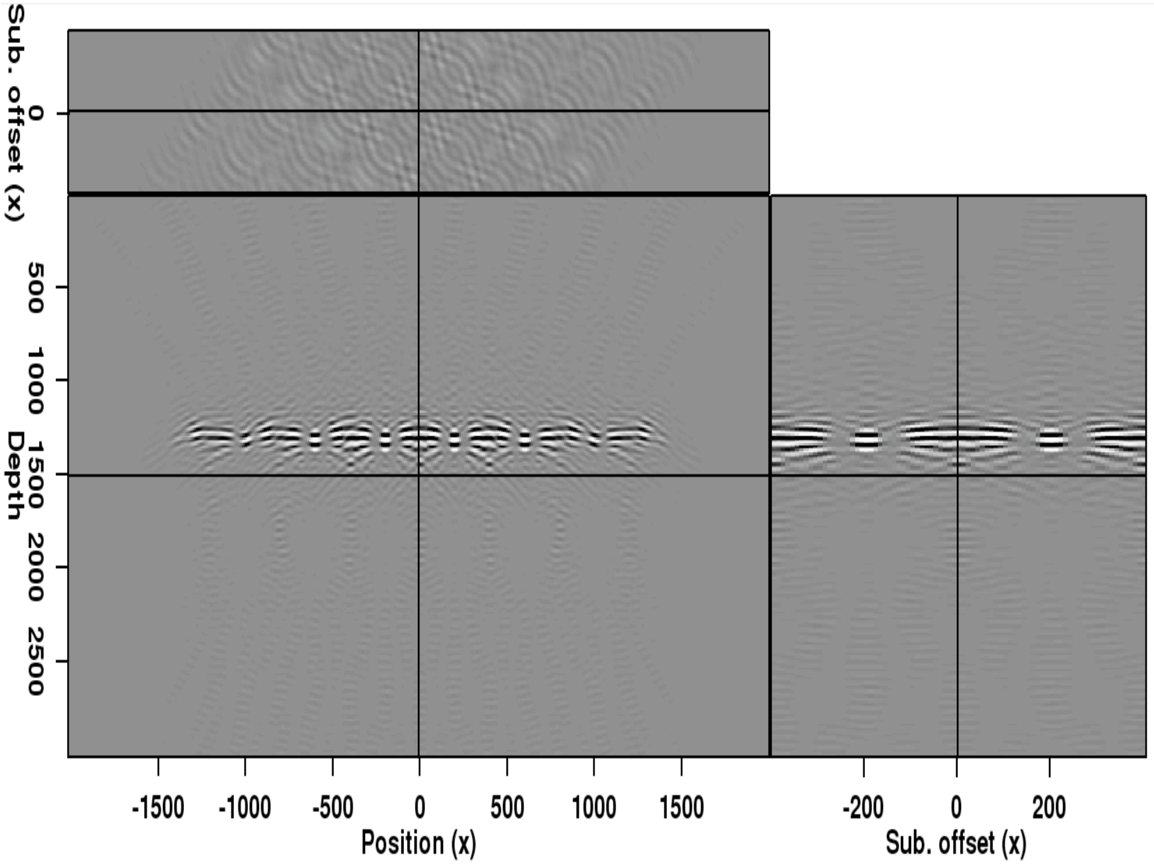


Figure 5: Migration result if the image points sampled from Figure 2(a) are spaced at less than twice the maximum subsurface offset. Crosstalk artifacts dominate the image, making interpretation extremely difficult. adam1/. flat-xtalk

As mentioned in the previous section, an advantage of this Born modeling strategy is that the synthesized data may be recorded at any depth, effectively re-datuming wavefields prior to migration. This can lead to significant computational savings, especially if velocities are well known until a certain depth. To verify that this capability does not effect the accuracy of migration results, we recorded both the areal source wavefield and the Born-modeled data at depth $z = 750$, instead of at the surface. Figure 6 shows the result of migrating this data in the dipping reflector case. Comparison with Figure 4(b) confirms that the two results are virtually identical for the area of interest.

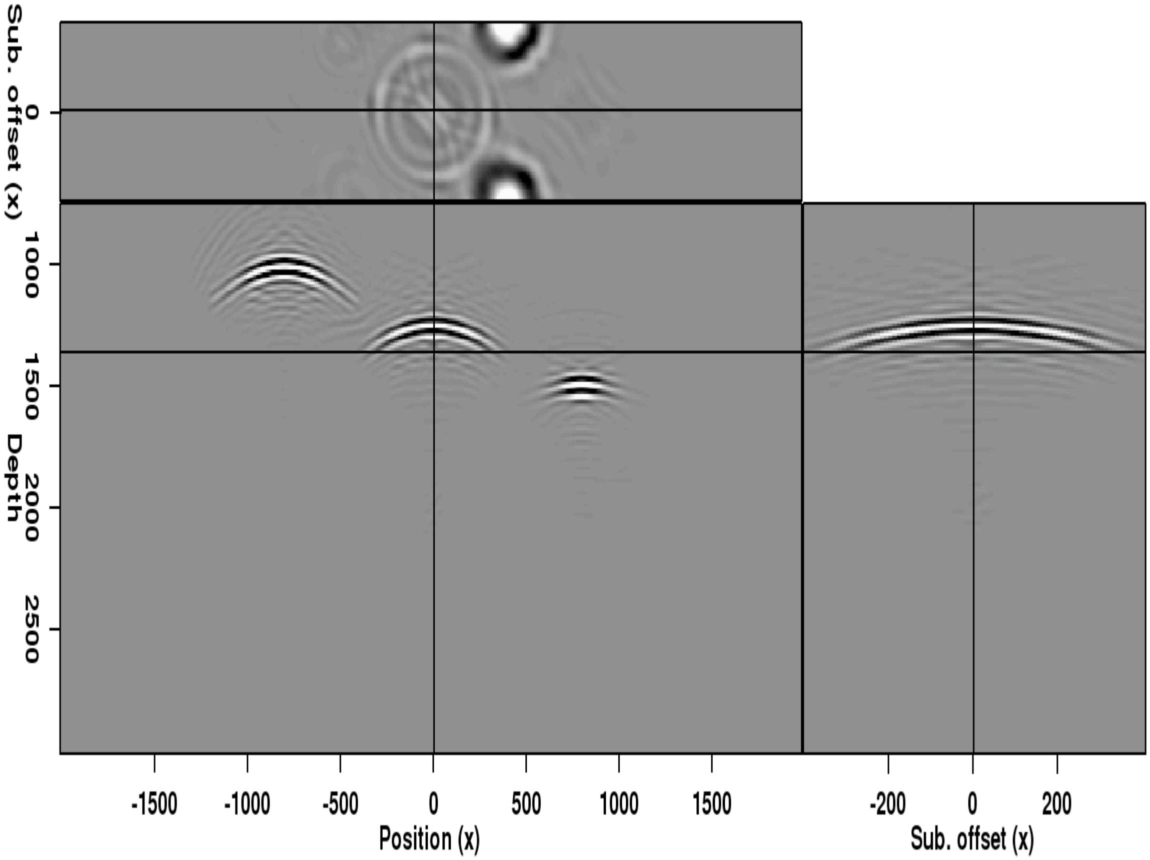


Figure 6: Migration result using Born-modeled data from the model in Figure 3(b). In this case, the synthesized data was recorded in the subsurface instead of on the surface, effectively re-datuming the wavefields. adam1/. dip-short

Finally, we wish to test the ultimate purpose of this method: quickly evaluating multiple velocity models. Once the Born-modeled dataset has been synthesized, we can use any velocity model to image the data. Again, we are able to perform these migrations very quickly, on the order of seconds for the examples here. Figure 7 compares the results of using three different velocity models to image the Born-modeled data: one that is 5% slower than the true velocity (Panel a); one that is exactly the true velocity (Panel b); and one that is 5% faster than the true velocity (Panel c). From these results, it is clear that the velocity model used to produce Panel b's result is the most accurate – the subsurface offset gather is flat and relatively focused, and, unlike Panels a and c, there are no signs of over- or under-migration on the zero-subsurface offset image. Because the velocity differences between these three models are relatively small, this is an encouraging sign that this method can ultimately allow us to quickly test more complex models for both synthetic and field data.

CONCLUSIONS

We investigated Born modeling techniques as a means to quickly evaluate multiple possible migration velocity models. By "exploding" subsurface offset gathers from an initial migrated image, we can generate an areal source function with information about the initial velocity model. This source function is used to both generate Born-modeled data from isolated points in the subsurface, and to migrate that data to form an image. While crosstalk issues limit the present implementation of this method to single reflectors, we showed that it can quickly and accurately reproduce the same velocity information (in the form of subsurface offset gathers) obtained from a full migration of the original data. Furthermore, the method allows for re-datuming of wavefields prior to imaging, and can clearly distinguish between velocity models that differ only slightly. With further improvement, this method could form the basis for an efficient and interactive model-building tool.

REFERENCES

Chauris, H. and M. Benjemaa, 2010, Seismic wave-equation demigration/migration: Geophysics, **75**, S111–S119.

Claerbout, J., 2005, Basic earth imaging: Stanford University.

Guerra, C., 2010, Migration-velocity analysis using image-space generalized wavefields: PhD thesis, Stanford University.

Halpert, A. D., R. G. Clapp, and B. L. Biondi, 2011, Interpreter guidance for automated seismic image segmentation: 74th EAGE Conference and Exhibition.

Hill, N. R., 1990, Gaussian beam migration: Geophysics, **55**, 1416–1428.

Mosher, C., E. Keskula, J. Malloy, R. Keys, H. Zhang, and S. Jin, 2007, Iterative imaging for subsalt interpretation and model building: The Leading Edge, **26**, 1424–1428.

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

Stolt, R. H. and A. Benson, 1986, Seismic migration: Theory and practice: Geophysical Press.

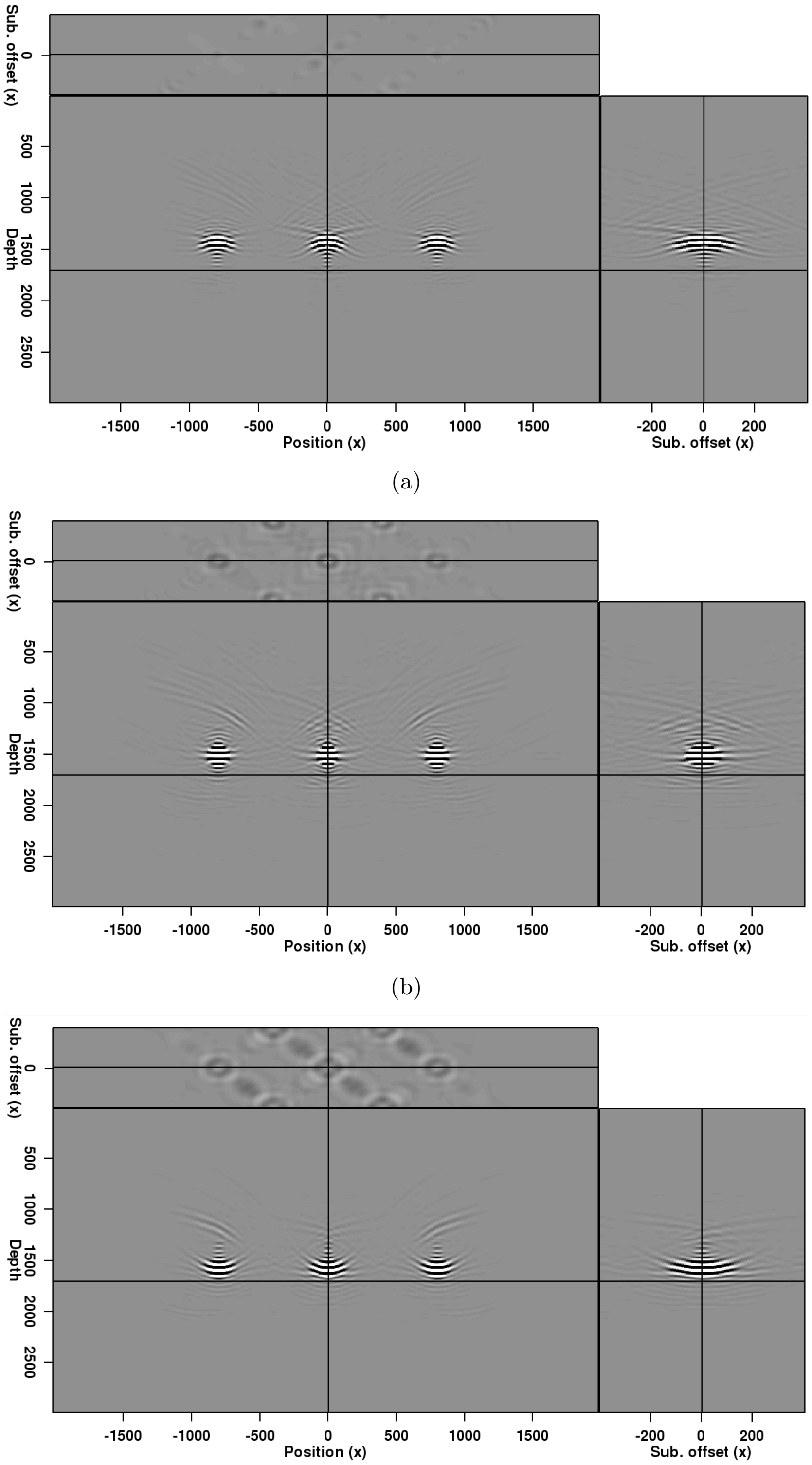
Tang, Y., 2008, Modeling, migration, and inversion in the generalized source and receiver domain: SEP-Report, **136**, 97–112.

———, 2011, Imaging and velocity analysis by target-oriented wavefield inversion: PhD thesis, Stanford University.

Tang, Y. and B. Biondi, 2010, Target-oriented wavefield tomography using demigrated born data: SEP-Report, **140**, 67–82.

Wang, B., J. Ji, C. Mason, S. Gajawada, and Y. Kim, 2008, Beam-based interactive imaging for salt interpretation and salt model building: SEG Technical Program Expanded Abstracts, **27**, 3073–3077.

Wang, B., C. Mason, K. Yoon, J. Ji, J. Cai, S. Suh, and Z. Li, 2011, Complex salt model building using a combination of interactive imaging and layer-stripping RTM: First Break, **29**, 47–54.



(a)

(b)

(c)

Figure 7: Result of migrating the Born-modeled data with (a) 5% too slow velocity; (b) correct velocity; and (c) 5% too fast velocity. Each migration was nearly instantaneous, and the effects of the different velocity models are readily apparent. adam1/. comp-slow,comp-act,comp-fast

Fast automatic wave-equation migration velocity analysis using encoded simultaneous sources

Yaxun Tang

ABSTRACT

I present a method based on source encoding for fast wave-equation migration velocity analysis (WEMVA). Instead of migrating each impulsive-source gather separately, I assemble all gathers together and migrate only one super shot gather. This procedure results in the computational cost of WEMVA to be independent from the number of impulsive-source gathers, which is typically huge for large surveys. The proposed encoding method can be applied to data acquired from any acquisition geometry, such as land or marine acquisition geometries. The velocity inversion is done automatically by solving a nonlinear optimization problem that maximizes the image stack power, which is shown to be equivalent to the data-domain inversion using only primary reflections. Preliminary results show that WEMVA with encoded sources can produce inversion results similar to those produced by conventional separate-source WEMVA, but with drastically reduced computational cost.

INTRODUCTION

Accurate reflectivity imaging requires an accurate background velocity model. As seismic exploration moves towards structurally complex areas, wave-equation migration velocity analysis (WEMVA) that better models band-limited wave phenomena becomes necessary for high-quality velocity model building. WEMVA, however, is still expensive for industrial-scale applications (Biondi and Sava, 1999; Shen et al., 2005; Albertin et al., 2006; Fei et al., 2009), both because the method uses expensive wavefield modeling engines, and because the computation needs to be carried out for each shot, resulting in a cost proportional to the number of sources, which is huge for large surveys.

Source encoding has been used in both seismic acquisition (Beasley et al., 1998; Beasley, 2008; Hampson et al., 2008; Berkhout, 2008; Tang and Biondi, 2009) and processing (Romero et al., 2000; Whitmore, 1995; Zhang et al., 2005; Liu et al., 2006; Krebs et al., 2009) to reduce the cost. The idea is that instead of firing one impulsive source at a time, we fire all encoded impulsive sources simultaneously for acquisition or/and processing. By doing so, the acquisition or/and processing effort is reduced to just one super areal shot gather instead of many impulsive source gathers, significantly reducing the acquisition or/and processing cost. In this paper, I mainly focus on applying the source-encoding method to seismic processing, hence I assume that the data are acquired using conventional impulsive separate sources without any overlaps.

Source encoding has been widely used in migration processing, where random-phase encoding and plane-wave-phase encoding are the most popular encoding schemes. The random-phase-encoding migration, however, has had limited success. This is because the more shots randomly encoded together, the more crosstalk present in the migration image. Consequently, images obtained with many realizations need to be computed and stacked in order to attenuate the crosstalk (Romero et al., 2000). Plane-wave phase-encoding migration, on the other hand, has wider applications than random phase-encoding migration. This is because plane-wave phase-encoding function has good properties in terms of converging to a Dirac delta function (Liu et al., 2006). However, multiple plane waves need to be synthesized and migrated to remove the crosstalk artifacts. As a result, source encoding in migration can usually achieve a cost reduction by a factor of about 10 or less.

As opposed to seismic migration processing, source encoding (especially random phase encoding) seems to be more effective in seismic inversion processing, such as least-squares migration (Tang and Biondi, 2009; Dai and Schuster, 2009; Dai et al., 2010) and full waveform inversion (Krebs et al., 2009; Tang and Lee, 2010; Ben-Hadj-Ali et al., 2011). The key element in encoded simultaneous-source inversion is the regeneration of random codes at each iteration (Krebs et al., 2009; Tang and Lee, 2010; Dai et al., 2010; Ben-Hadj-Ali et al., 2011). Different sets of random codes at each iteration enable destructive summation of the crosstalk over iterations, and consequently the residual crosstalk in the inverted model is gradually removed as inversion proceeds. Encoded simultaneous-source inversion operates on one super shot gather instead of many impulsive-source gathers at each iteration, therefore the computational cost is independent of the number of sources. Although more or less counter-intuitive, Krebs et al. (2009) have reported that encoded simultaneous-source inversion has a similar convergence rate compared to separate-source inversion. As a result, source encoding in inversion can achieve a cost reduction by a factor of the number of sources, which can be significant for large surveys.

One commonality of the above mentioned inversion processing (least-squares migration and full waveform inversion) is the minimization of a data-domain objective function, which compares the differences between the modeled and the observed data. The difference-based objective function, however, restricts the application of encoded simultaneous-source inversion to only data acquired with a fixed receiver spread, such as in land or ocean bottom cable (OBC) acquisition geometries (Krebs et al., 2009). This is because modeling using encoded simultaneous sources implicitly assumes that each receiver listens to all shots. This is obviously not the case for marine acquisition geometries, where the towed receiver spread moves along with the sources. The mismatch in acquisition is irreconcilable and will cause wrong model updates.

In this paper, I apply the source-encoding method to WEMVA, which optimizes an objective function formulated in the image domain instead of the data domain. In particular, I optimize the velocity by maximizing the image stack power (or minimizing its negative). I will show that the objective function to maximize (or minimize) is based on the cross-correlation between the source and receiver wavefields, and that source encoding can be applied to arbitrary acquisition geometries, regardless of whether or not the receiver spread is fixed. Similar to the data-domain multi-source inversion, encoded simultaneous-source WEMVA also generates gradients contaminated by crosstalk. Therefore, regeneration of random codes at each iteration becomes necessary to mitigate the impact of crosstalk on velocity updates.

In the subsequent sections, I first review the theory of WEMVA based on image-stack-power maximization (or equivalently negative image-stack-power minimization). I prove that minimizing the negative image stack power is equivalent to the data domain Born wavefield inversion, which minimizes the difference between the modeled and observed primary reflections. I then show how source encoding can be applied to WEMVA. Finally, I apply both separate-source WEMVA and encoded simultaneous-source WEMVA to invert a truncated Marmousi model.

THEORY

I pose the velocity estimation problem as an optimization problem that tries to maximize the image stack power across the reflection angle, taking advantage of the fact that seismic events should be aligned and hence most constructively stacked in the angle domain, if migrated using an accurate velocity model (Soubaras and Gratacos, 2007). Instead of solving it as a maximization problem, I actually solve it as a minimization problem that minimizes the negative image stack power. Because the reflection-angle stacked section is equivalent to the zero-subsurface offset image, the objective function that I use to minimize is therefore defined as follows:

$$J = -\frac{1}{2}\sum_{\mathbf{x}} m_{\text{mig}}^2(\mathbf{x}), \tag{1}$$

where $m_{\text{mig}}(\mathbf{x})$ is the zero-subsurface-offset image at image point $\mathbf{x} = (x, y, z)$, obtained by crosscorrelating the forward propagated source wavefield with the backward propagated receiver wavefield as follows:

$$m_{\text{mig}}(\mathbf{x}) = \sum_{\mathbf{x}_s}\sum_{\omega} S(\mathbf{x}, \mathbf{x}_s, \omega) R(\mathbf{x}, \mathbf{x}_s, \omega), \tag{2}$$

where $S(\mathbf{x}, \mathbf{x}_s, \omega)$ and $R(\mathbf{x}, \mathbf{x}_s, \omega)$ are the source and receiver wavefield at image point $\mathbf{x}$, respectively, for a source located at $\mathbf{x}_s = (x_s, y_s, 0)$ and at angular frequency ω. If a one-way extrapolator is used, S and R satisfy the following one-way wave equations:

$$\left\{ \begin{array}{l} \left(\frac{\partial}{\partial z} + i\sqrt{\frac{\omega^2}{v^2(\mathbf{x})} + \nabla^2}\right) S(\mathbf{x}, \mathbf{x}_s, \omega) = 0 \\ S(x, y, z = 0, \mathbf{x}_s, \omega) = \delta(x - x_s, y - y_s) f_s^*(\omega) \end{array} \right. , \tag{3}$$

and

$$\left\{ \begin{array}{l} \left(\frac{\partial}{\partial z} + i\sqrt{\frac{\omega^2}{v^2(\mathbf{x})} + \nabla^2}\right) R(\mathbf{x}, \mathbf{x}_s, \omega) = 0 \\ R(x, y, z = 0, \mathbf{x}_s, \omega) = Q(x, y, \mathbf{x}_s, \omega) \end{array} \right. , \tag{4}$$

where * denotes taking the adjoint; $v(\mathbf{x})$ is the velocity at image point $\mathbf{x}$; $f_s(\omega)$ is the source signature; $\delta(\cdot)$ is the Dirac delta function; $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is the Laplacian operator. Q is the observed data mapped onto the computation grid, which is defined as follows:

$$Q(x, y, \mathbf{x}_s, \omega) = \sum_{\mathbf{x}_r} W(\mathbf{x}_r, \mathbf{x}_s) \delta(x - x_r, y - y_r) d_{\text{obs}}(\mathbf{x}_r, \mathbf{x}_s, \omega), \tag{5}$$

where d_{obs} is the observed data recorded at $\mathbf{x}_r = (x_r, y_r, 0)$ due to a source at $\mathbf{x}_s$; $W(\mathbf{x}_r, \mathbf{x}_s)$ is the acquisition mask operator, which contains ones where we record data and zeros where we do not.

Since flat angle gathers generate the most coherent stack, the negative image-stack-power minimization objective function defined by equation 1 is intuitive to understand.

Objective function 1, however, has an alternative interesting interpretation as shown in Appendix A, which proves that under the least-squares assumption, minimization of objective function 1 is equivalent to the data-domain Born wavefield inversion, which minimizes the differences between the modeled and observed primary reflections.

Objective function 1 is usually minimized using local optimization techniques, which require explicit calculation of the gradient. The gradient is obtained by taking the derivative of J with respect to velocity $v(\mathbf{y})$ ($\mathbf{y}$ is the velocity coordinates) as follows:

$$g(\mathbf{y}) = \frac{\partial J}{\partial v(\mathbf{y})} = -\sum_{\mathbf{x}} \frac{\partial m_{\text{mig}}(\mathbf{x})}{\partial v(\mathbf{y})} m_{\text{mig}}(\mathbf{x}), \tag{6}$$

where the sensitivity kernel, or tomographic operator, $\frac{\partial m_{\text{mig}}(\mathbf{x})}{\partial v(\mathbf{y})}$, can be easily obtained as follows:

$$\frac{\partial m_{\text{mig}}(\mathbf{x})}{\partial v(\mathbf{y})} = \sum_{\mathbf{x}_s} \sum_{\omega} \left(\frac{\partial S(\mathbf{x}, \mathbf{x}_s, \omega)}{\partial v(\mathbf{y})} R(\mathbf{x}, \mathbf{x}_s, \omega) + S(\mathbf{x}, \mathbf{x}_s, \omega) \frac{\partial R(\mathbf{x}, \mathbf{x_s}, \omega)}{\partial v(\mathbf{y})} \right). \tag{7}$$

Note the summations over $\mathbf{x}_s$ in equations 2 and 7. This means that the computation for the image m_{mig} and the gradient g needs to be carried out for each source independently, resulting in a cost proportional to the number of sources. The gradient g is usually calculated using the adjoint-state technique without explicitly constructing the sensitivity kernel (Shen, 2004; Sava and Vlad, 2008; Tang et al., 2008).

For encoded simultaneous-source WEMVA, the objective function to be minimized is defined as follows:

$$\widetilde{J} = -\frac{1}{2} \sum_{\mathbf{x}} \widetilde{m}^2_{\text{mig}}(\mathbf{x}), \tag{8}$$

where the zero-subsurface-offset image $\widetilde{m}_{\text{mig}}$ is obtained by crosscorrelating the encoded source wavefield, $\widetilde{S}$, and the encoded receiver wavefield, $\widetilde{R}$, as follows:

$$\widetilde{m}_{\text{mig}}(\mathbf{x}) = \sum_{\omega} \widetilde{S}(\mathbf{x}, \omega) \widetilde{R}(\mathbf{x}, \omega). \tag{9}$$

The encoded source and receiver wavefields satisfy the following one-way wave equations:

$$\left\{ \begin{array}{l} \left(\frac{\partial}{\partial z} + i\sqrt{\frac{\omega^2}{v^2(\mathbf{x})} + \nabla^2} \right) \widetilde{S}(\mathbf{x}, \omega) = 0 \\ \widetilde{S}(x, y, z = 0, \omega) = \sum_{\mathbf{x}_s} \delta(x - x_s, y - y_s) f_s^*(\omega) \alpha^*(\mathbf{x}_s, \omega) \end{array} \right. , \tag{10}$$

and

$$\left\{ \begin{array}{l} \left(\frac{\partial}{\partial z} + i\sqrt{\frac{\omega^2}{v^2(\mathbf{x})} + \nabla^2} \right) \widetilde{R}(\mathbf{x}, \omega) = 0 \\ \widetilde{R}(x, y, z = 0, \omega) = \sum_{\mathbf{x_s}} Q(x, y, \mathbf{x}_s, \omega) \alpha(\mathbf{x}_s, \omega) \end{array} \right. , \tag{11}$$

where $\alpha(\mathbf{x}_s, \omega)$ is the phase encoding function. In this paper, I mainly focus on random phase encoding, therefore α is defined as follows:

$$\alpha(\mathbf{x}_s, \omega) = e^{i\gamma(\mathbf{x}_s, \omega)}, \tag{12}$$

where $i = \sqrt{-1}$, and $\gamma(\mathbf{x}_s, \omega)$ is a uniformly distributed random sequence from 0 to 2π. Tang (2011) shows that with this choice of random phase function, α has a zero expectation. Note that the source encoding can be applied to data recorded from arbitrary types of acquisition geometries. The simultaneous-source migrated image ($\widetilde{m}_{\text{mig}}$) will always converge to the separate-source migrated image (m_{mig}) as long as the encoding function satisfies $\alpha^*(\mathbf{x}_s, \omega)\alpha(\mathbf{x}'_s, \omega) \approx \delta(\mathbf{x}_s - \mathbf{x}'_s)$.

The gradient of objective function 8 is

$$\widetilde{g}(\mathbf{y}) = \frac{\partial \widetilde{J}}{\partial v(\mathbf{y})} = -\sum_{\mathbf{x}} \frac{\partial \widetilde{m}_{\text{mig}}(\mathbf{x})}{\partial v(\mathbf{y})} \widetilde{m}_{\text{mig}}(\mathbf{x}), \tag{13}$$

where the tomographic operator, $\frac{\partial \widetilde{m}_{\text{mig}}(\mathbf{x})}{\partial v(\mathbf{y})}$, in the encoded-source domain is defined as follows:

$$\frac{\partial \widetilde{m}_{\text{mig}}(\mathbf{x})}{\partial v(\mathbf{y})} = \sum_{\omega} \left(\frac{\partial \widetilde{S}(\mathbf{x}, \omega)}{\partial v(\mathbf{y})} \widetilde{R}(\mathbf{x}, \omega) + \widetilde{S}(\mathbf{x}, \omega) \frac{\partial \widetilde{R}(\mathbf{x}, \omega)}{\partial v(\mathbf{y})} \right). \tag{14}$$

Note that equations 9 and 14 do not have a summation over the sources. Therefore, the cost of computing the image $\widetilde{m}_{\text{mig}}$ and the gradient $\widetilde{g}$ is independent of the number of sources, as opposed to the separate-source case. The gradient $\widetilde{J}$ is also calculated using the adjoint-state technique using encoded simultaneous sources (Tang et al., 2008).

Although the computational cost of WEMVA is significantly reduced, encoded simultaneous sources add crosstalk artifacts into the gradient. This becomes clear if we express the encoded source and receiver wavefield as follows using the fact that wavefield propagation is linear with respect to sources:

$$\widetilde{S}(\mathbf{x}, \omega) = \sum_{\mathbf{x}_s} S(\mathbf{x}, \mathbf{x}_s, \omega)\alpha^*(\mathbf{x}_s, \omega), \tag{15}$$

and

$$\widetilde{R}(\mathbf{x}, \omega) = \sum_{\mathbf{x}_s} R(\mathbf{x}, \mathbf{x}_s, \omega)\alpha(\mathbf{x}_s, \omega). \tag{16}$$

Substituting equations 15 and 16 into 14 and using the fact that $\alpha^*(\mathbf{x}_s, \omega)\alpha(\mathbf{x}'_s, \omega) = 1$ if $\mathbf{x}_s = \mathbf{x}'_s$ yield

$$\widetilde{g}(\mathbf{y}) = g(\mathbf{y}) + g_c(\mathbf{y}), \tag{17}$$

where g_c is the crosstalk:

$$\begin{aligned} g_c(\mathbf{y}) = \sum_{\omega}\sum_{\mathbf{x}_s}\sum_{\mathbf{x}'_s \neq \mathbf{x}_s} & \left(\frac{\partial S(\mathbf{x}, \mathbf{x}_s, \omega)}{\partial v(\mathbf{y})} R(\mathbf{x}, \mathbf{x}'_s, \omega) + S(\mathbf{x}, \mathbf{x}_s, \omega) \frac{\partial R(\mathbf{x}, \mathbf{x}'_s, \omega)}{\partial v(\mathbf{y})} \right) \\ & \times \alpha^*(\mathbf{x}_s, \omega)\alpha(\mathbf{x}'_s, \omega). \end{aligned} \tag{18}$$

A way to mitigate the influence of crosstalk is to change the random encoding function at each iteration (Krebs et al., 2009), so that the crosstalk will be destructively stacked over WEMVA iterations and consequently converge to zero because it has a zero expectation. It is important to point out that regeneration of the random code will result in the objective function (equation 8) changing at each iteration. Therefore, the objective function may not be monotonically decreasing over iterations, as opposed to the case in conventional separate-source WEMVA. The optimization algorithm using encoded simultaneous sources is summarized in Algorithm 1.

Algorithm 1 Encoded simultaneous-source WEMVA algorithm

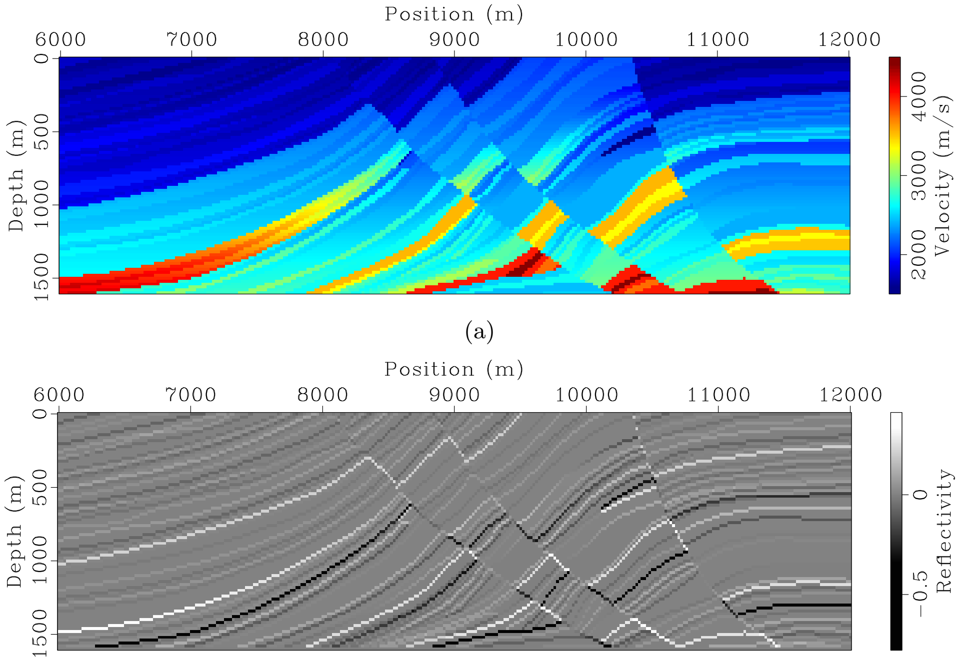
```
generate random code and assemble all shot gathers together
compute the migrated image: $\widetilde{\mathbf{m}}^0_{\text{mig}}$
compute the gradient: $\widetilde{\mathbf{g}}_0$
initialize the search direction: $\mathbf{p}_0 = -\widetilde{\mathbf{g}}_0$
for $k = 1 \cdots N_k$ do
    perform line search: optimize $\lambda$, $\underset{\lambda}{\text{argmin}}\ \widetilde{J}(\mathbf{v}_{k-1} + \lambda \mathbf{p}_{k-1})$
    update the velocity model: $\mathbf{v}_k = \mathbf{v}_{k-1} + \lambda \mathbf{p}_{k-1}$
    generate random code and assemble all shot gathers together
    compute the migrated image: $\widetilde{\mathbf{m}}^k_{\text{mig}}$
    compute the gradient: $\widetilde{\mathbf{g}}_k$
    find the search direction: $\mathbf{p}_k = -\widetilde{\mathbf{g}}_k + \frac{(\widetilde{\mathbf{g}}_k)^T(\widetilde{\mathbf{g}}_k - \widetilde{\mathbf{g}}_{k-1})}{(\widetilde{\mathbf{g}}_{k-1})^T \widetilde{\mathbf{g}}_{k-1}}$
end for
```

NUMERICAL EXAMPLES

I apply the encoded simultaneous-source WEMVA on a truncated Marmousi model. The data used for inversion are generated using prestack Born wavefield modeling (Stolt and Benson, 1986; Tang, 2011). Hence the data only contain primary reflections and fit the theory perfectly. I use one-way wavefield extrapolation to carry out the numerical experiments. Since one-way wavefield extrapolation does not generate back scatterings, the velocity model used for Born modeling does not need to be smooth. Figures 1(a) and 1(b) show the velocity model and reflectivity model used for Born modeling. I use a Ricker wavelet with a dominant frequency of 15 Hz as the source function for modeling. The source function is assumed to be known in the subsequent inversion tests.

The initial velocity model used for inversion is shown in Figure 2. It is a smoothed version of the true velocity model (Figure 1(a)). The initial velocity model is accurate enough so that no cycle skipping occurs during inversion. The goal of the experiments shown here is to demonstrate that with an initial velocity model that guarantees the convergence of inversion, encoded simultaneous-source WEMVA produces similar inversion result as does conventional separate-source WEMVA, but with a significantly reduced computational cost. However, the convergence property using an initial velocity model far from the correct one still needs to be studied, and it remains an area for further investigation.

I test the inversion on data sets acquired using both land and marine acquisition geometries. The data acquired from a land acquisition geometry contains 101 shots ranging from 5 km to 13 km with a 80 m sampling interval. The receiver spread ranges from 5 km to 13 km and is fixed for all shots. The receiver sampling interval is 20 m. For the data acquired using a marine acquisition geometry, the 101 sources also range from 5 km to 13 km and sampled at 80 m. The minimum and maximum offsets for each shot is 0 and 6 km.



(a)

(b)

Figure 1: The velocity model (a) and reflectivity model (b) used for Born wavefield modeling. [**ER**] yaxun1/. marm-vmod,marm-refl

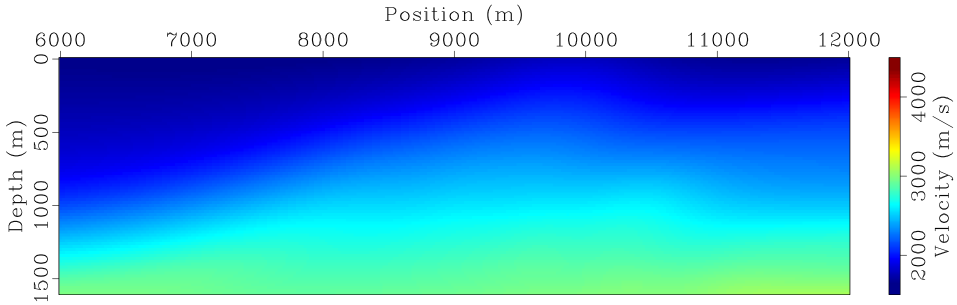
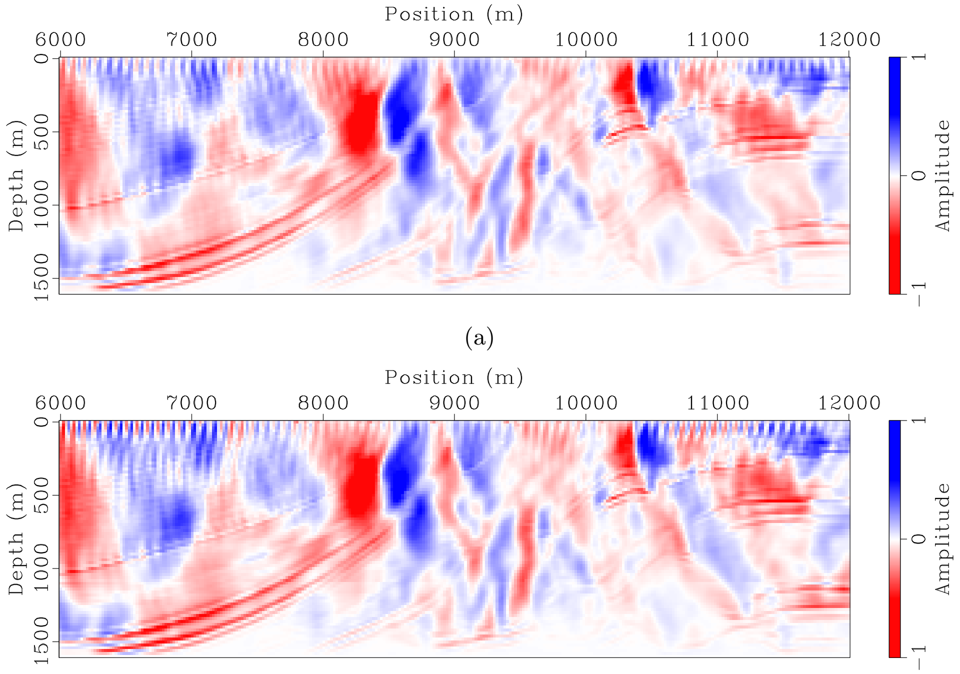


Figure 2: The initial velocity model. [**ER**] yaxun1/. marm-bvel

The receiver sampling interval is also 20 m.

I run inversion using both separate sources and encoded simultaneous sources after the same number of iterations. Figures 3 and 4 compare the WEMVA gradients at the first iteration for different methods and different acquisition geometries. Note the randomized crosstalk present in the simultaneous-source WEMVA gradients. Because I regenerate the random code at the beginning of each iteration, the crosstalk is expected to be incoherently stacked over iterations. Therefore, the impact of crosstalk will be mitigated.



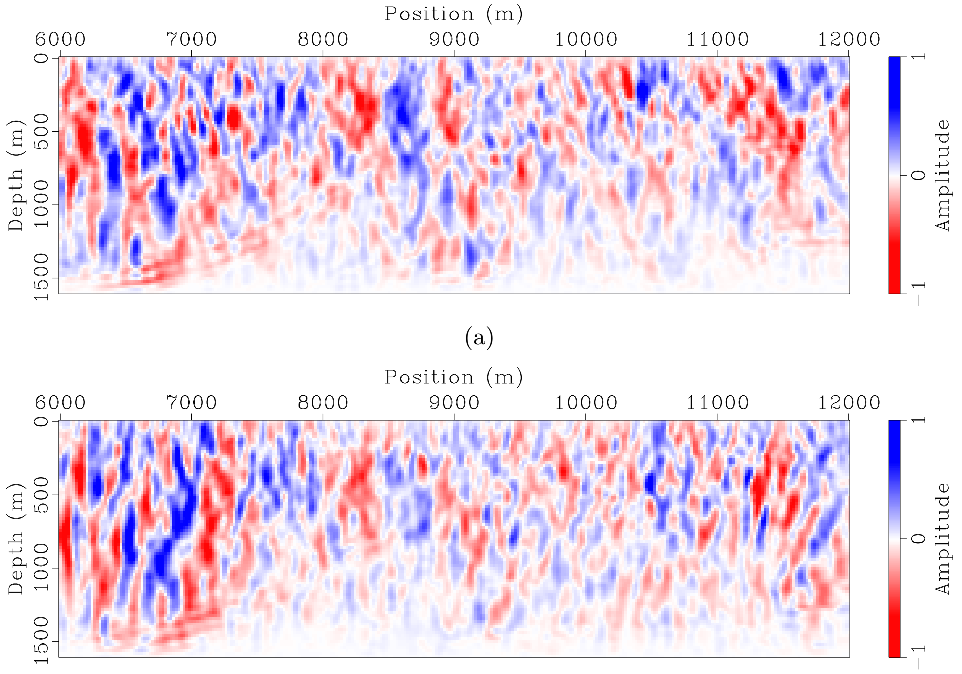
(a)

(b)

Figure 3: Gradient of separate-source WEMVA at the first iteration for (a) land acquisition geometry and (b) marine acquisition geometry. [**ER**] yaxun1/. marm-grad-separate,marm-grad-separate-marine

Figures 5 and 6 show the separate-source inversion results at different iterations for land and marine acquisition geometries, respectively. The velocity model has been successfully recovered in both cases. But inversion using land acquisition geometry produces a slightly better final inversion result (Figure 5(d)) than the one obtained using marine acquisition geometry (Figure 6(d)). This is because, for this particular example, the land acquisition geometry has wider offsets and hence gives better coverage to the model.

For comparison, Figures 7 and 8 present the encoded simultaneous-source inversion results for land and marine acquisition geometries, respectively. As expected, the inverted velocity model at early iterations (Figures 7(a) and 7(b) for land acquisition geometry and Figures 8(a) and 8(b) for marine acquisition geometry) have been strongly affected by the crosstalk artifacts in the gradients (Figure 4). As inversion proceeds, the crosstalk artifacts are destructively stacked, and hence the influence of crosstalk is decreasing over iterations. The final inversion results (Figures 7(d) and 8(d)) also successfully recover the velocity model. However, the encoded simultaneous-source WEMVA seems to be more sensitive to



(a)

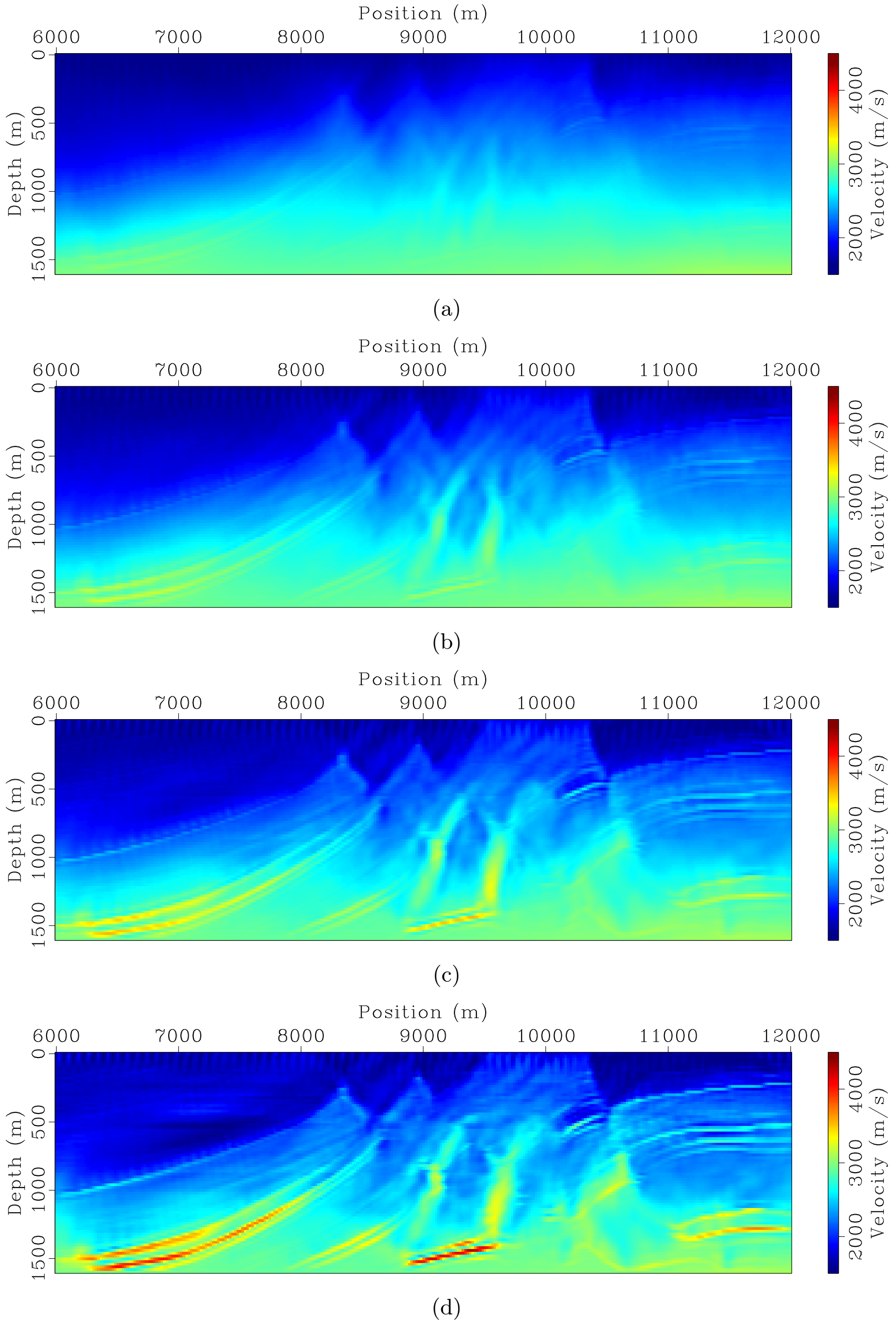
(b)

Figure 4: Gradient of encoded simultaneous-source WEMVA at the first iteration for (a) land acquisition geometry and (b) marine acquisition geometry. [**ER**] yaxun1/. marm-grad,marm-grad-marine

the model coverage, and the convergence of inversion using the data acquired with a marine geometry (Figure 8(d)) is considerably slower than that obtained using the data acquired from a land acquisition geometry (Figure 7(d)).

As a further comparison of the convergence, Figure 9 shows the data misfit (the objective function) obtained using different methods and for different acquisition geometries. The data misfit curve for each case has been normalized with its value at the first iteration. Note that the data misfit functions decrease monotonically for separate-source inversions. This is because the objective function J is consistent over iterations, and therefore the nonlinear conjugate gradient algorithm tries to minimize the same objective function over iterations. In contrast, the data misfit functions for encoded simultaneous-source inversions fluctuate significantly, and they do not show monotonically decreasing behavior as do separate-source inversions. This is because the random phase encoding function keeps changing at each iteration, and consequently the objective function $\widetilde{J}$ varies over iterations. The nonlinear conjugate gradient algorithm cannot guarantee the monotonic decrease of the objective function. But the misfit functions do show an overall decreasing trend.

Since this is a synthetic-data example, the true velocity model is known. I calculate the model misfit in ℓ_2 norm and the results are plotted in Figure 10. It is interesting to note that although encoded simultaneous-source inversion does not show monotonic decrease of the data misfit, it does show monotonic decrease of the model misfit, which suggests that the inversion is going in the correct direction. Also note that the model convergence of



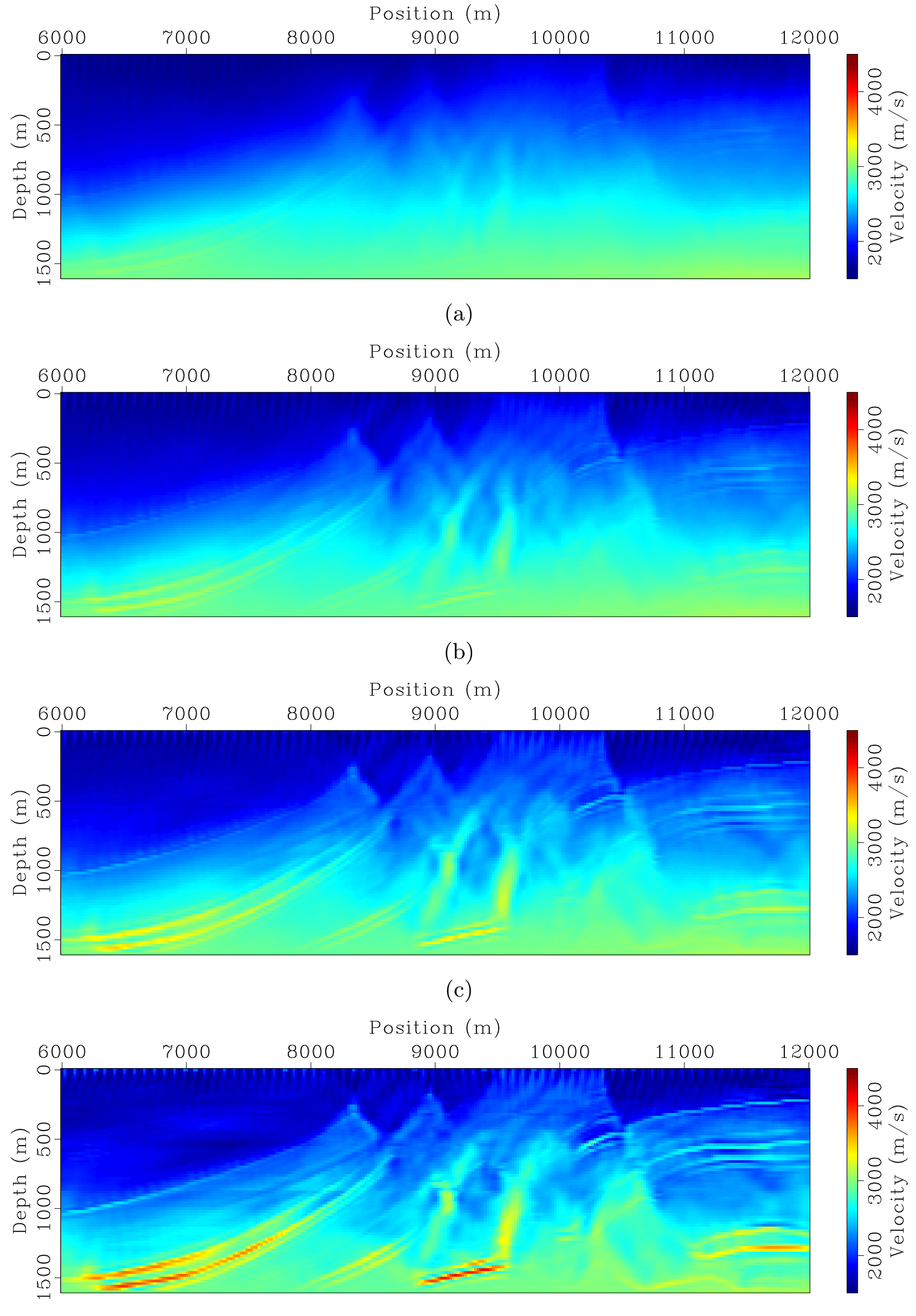
(a)

(b)

(c)

(d)

Figure 5: Separate-source WEMVA inversion result for land acquisition geometry at (a) 5, (b) 20, (c) 50 and (d) 120 iterations, respectively. [**CR**] yaxun1/. sinvt5,sinvt20,sinvt50,sinvt120



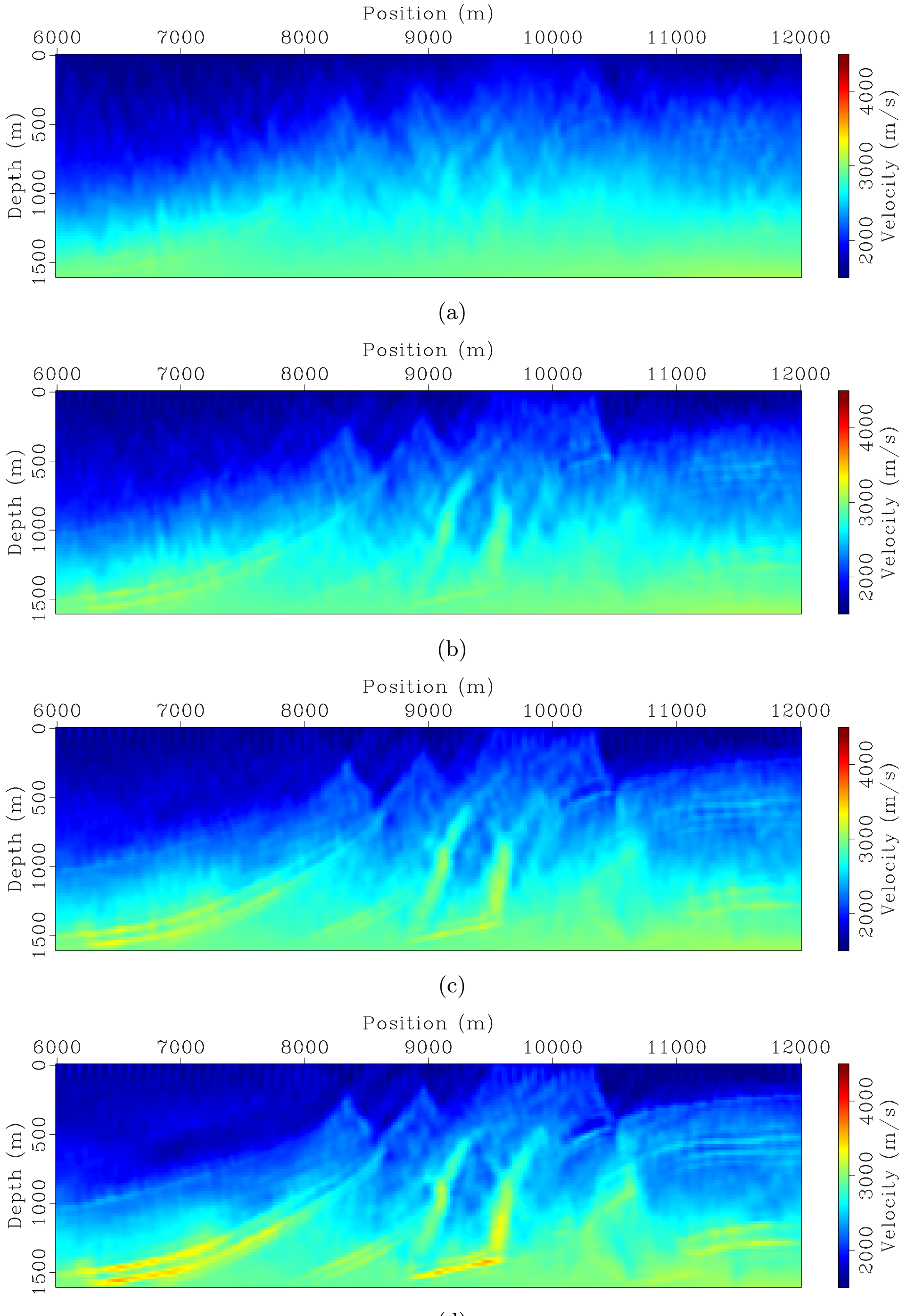
(a)

(b)

(c)

(d)

Figure 6: Separate-source WEMVA inversion result for marine acquisition geometry at (a) 5, (b) 20, (c) 50 and (d) 120 iterations, respectively. [**CR**] yaxun1/. sminvt5,sminvt20,sminvt50,sminvt120



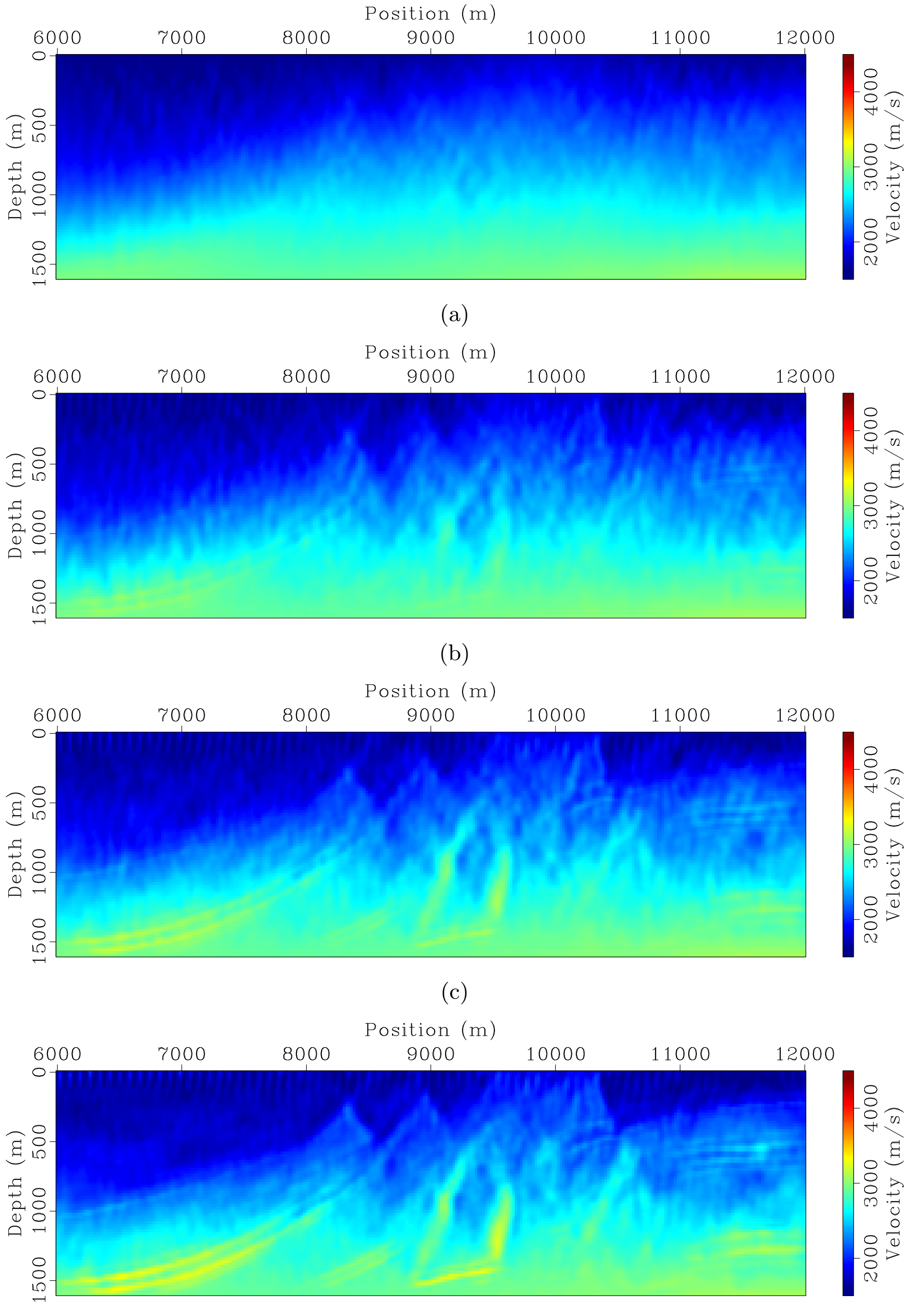
(a)

(b)

(c)

(d)

Figure 7: Encoded simultaneous-source WEMVA inversion result for land acquisition geometry at (a) 5, (b) 20, (c) 50 and (d) 120 iterations, respectively. [**ER**] yaxun1/. invt5,invt20,invt50,invt120



(a)

(b)

(c)

(d)

Figure 8: Encoded simultaneous-source WEMVA inversion result for marine acquisition geometry at (a) 5, (b) 20, (c) 50 and (d) 120 iterations, respectively. [**ER**] yaxun1/. minvt5,minvt20,minvt50,minvt120

encoded simultaneous-source inversion is slower than that of separate-source inversion. The difference seems to be insignificant for land acquisition geometries, where the receivers are fixed and the offsets are longer. The difference for marine acquisition geometries, however, is much bigger. This is probably because the marine acquisition geometry used in this example has shorter offsets and the data coverage is much less than the land acquisition geometry. The lack of data coverage may require more iterations to remove the crosstalk artifacts. This speculation, however, still needs more investigation to verify.

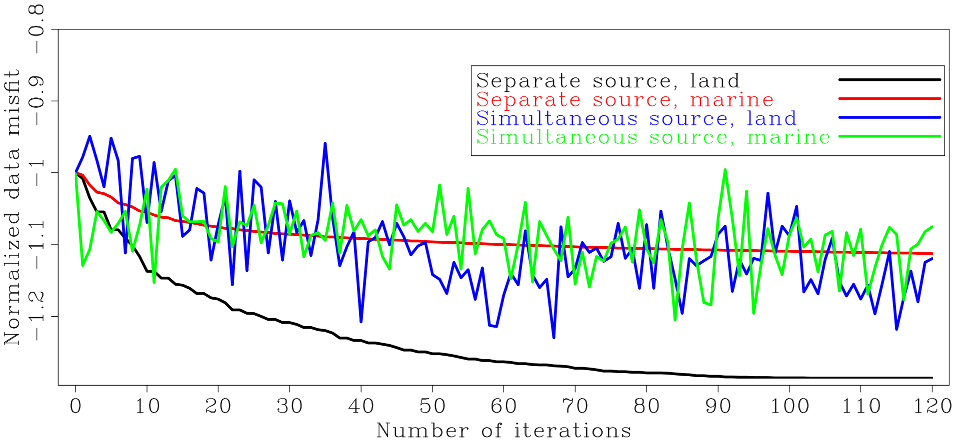


Figure 9: Normalized data misfit as a function of iterations for different methods. [**CR**] yaxun1/. marm-dfit

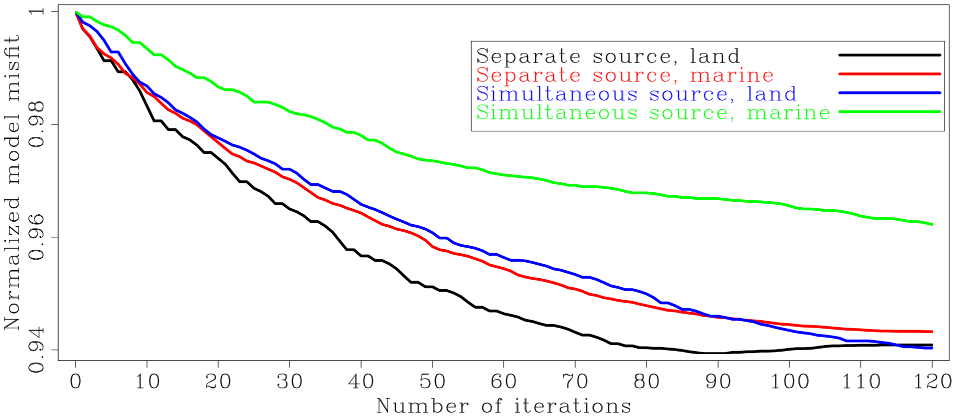
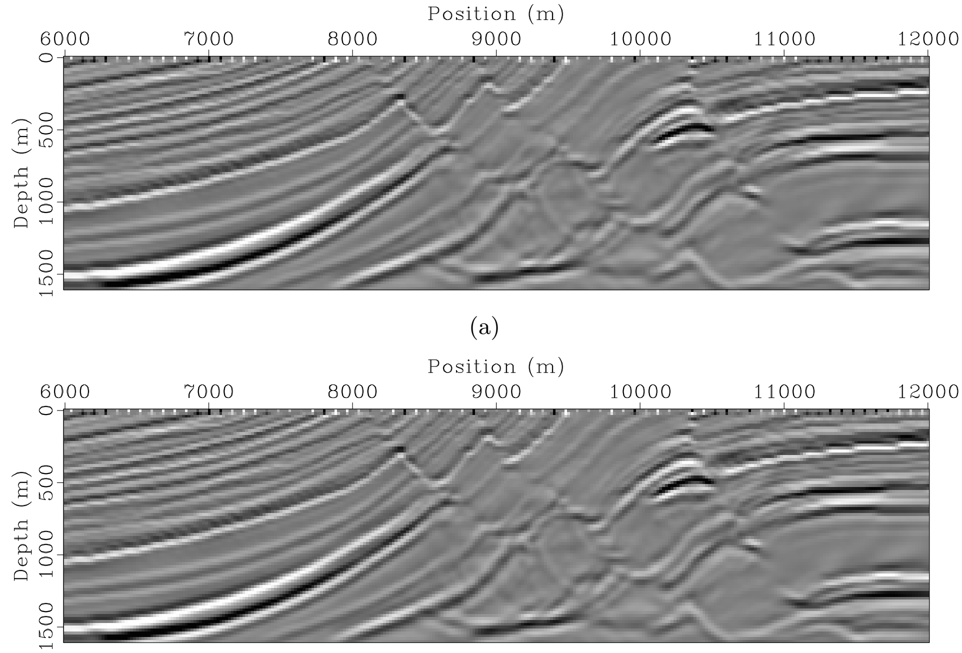


Figure 10: Normalized model misfit as a function of iterations for different methods. [**CR**] yaxun1/. marm-mfit

A final comparison is made among the images obtained using the inverted velocity model produced by different methods and for different acquisition geometries. The initial images (Figure 11) show poor focusing due to the velocity errors. The updated images using velocities obtained with separate sources (Figure 13) and encoded simultaneous sources (Figure 13) show significantly improvements on image focusing and coherence. The updated images using both separate-source inversion and encoded simultaneous-source inversion show very similar overall qualities, although separate-source inversion does produces slightly better images. But if we take the cost into account, the encoded simultaneous-source inversion is about 101 times faster than the separate-source inversion, which is a significant advantage. For comparison, Figure 14 presents images obtained using the true velocity model (Figure 1(a)).



(a)

(b)

Figure 11: Image obtained using the initial velocity model (Figure 2) for (a) land acquisition geometry and (b) marine acquisition geometry. [**ER**] yaxun1/. bimg,bimg-marine

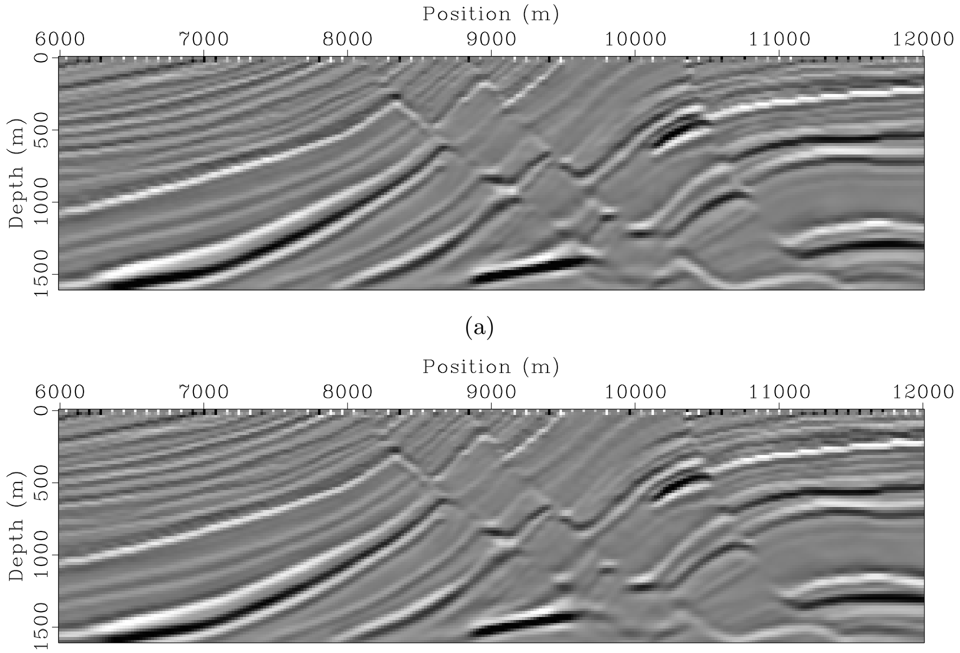
CONCLUSIONS

I have presented an efficient method for velocity optimization using wavefields. The method is automatic because it maximizes the image stack power (or minimizes its negative) and no picking is necessary. The method is extremely fast because it assembles all shot gathers together and migrates them at once, instead of migrating them separately. The shot gathers need to be reassembled with regeneration of random phase-encoding functions at each iteration to mitigate the impact of crosstalk present in the gradient. The encoding strategy can be applied to both land and marine acquisition geometries, regardless of whether or not a fixed receiver spread has been used. Numerical examples demonstrate that encoded simultaneous-source inversion gives reasonably good recovery of the velocity model, with the advantage that the computational cost is independent from the number of sources.

REFERENCES

Albertin, U., P. Sava, J. Etgen, and M. Maharramov, 2006, Adjoint wave-equation velocity analysis: SEG Technical Program Expanded Abstracts, **25**, 3345–3349.

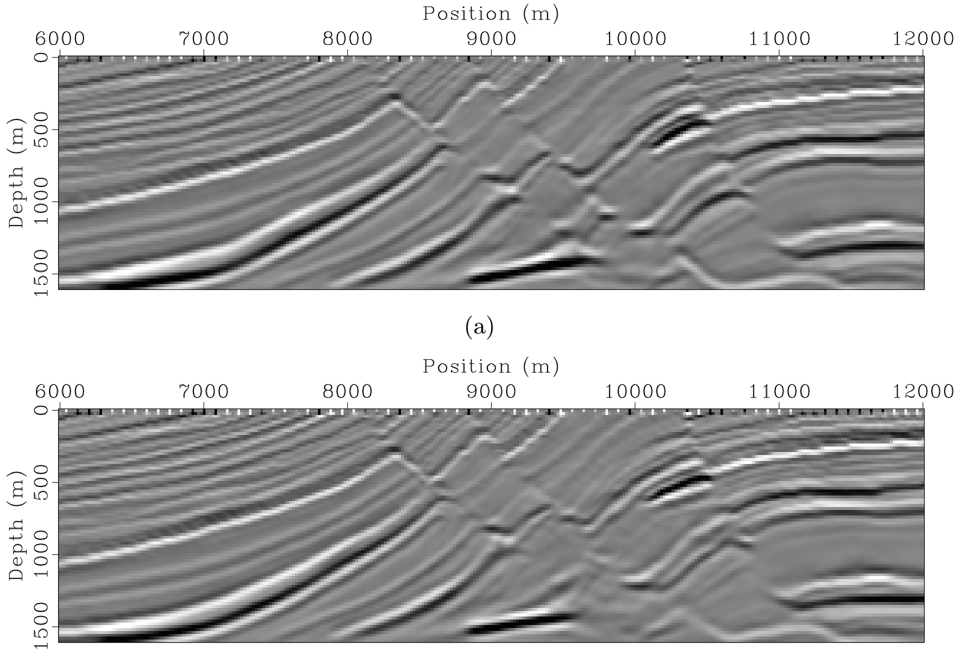
Beasley, C. J., 2008, A new look at marine simultaneous sources: The Leading Edge, **27**, 914–917.



(a)

(b)

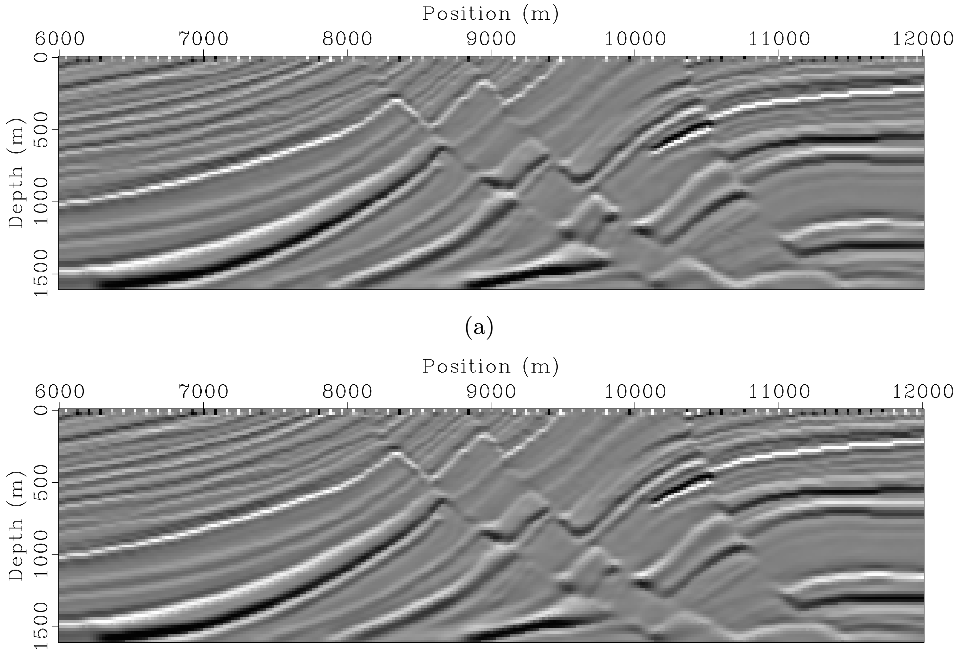
Figure 12: Image obtained using the separate-source inverted velocity model for (a) land acquisition geometry and (b) marine acquisition geometry. [**CR**] yaxun1/. imag-invt-separate,imag-invt-separate-marine



(a)

(b)

Figure 13: Image obtained using the encoded simultaneous-source inverted velocity model for (a) land acquisition geometry and (b) marine acquisition geometry. [**ER**] yaxun1/. imag-invt,imag-invt-marine



(a)

(b)

Figure 14: Image obtained using the true velocity model (Figure 1(a)) for (a) land acquisition geometry and (b) marine acquisition geometry. [**ER**] yaxun1/. imag,imag-marine

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

Ben-Hadj-Ali, H., S. Operto, and J. Virieux, 2011, An efficient frequency-domain full waveform inversion method using simultaneous encoded sources: Geophysics, **76**, R109–R124.

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

Biondi, B. and P. Sava, 1999, Wave-equation migration velocity analysis: SEG Technical Program Expanded Abstracts, **18**, 1723–1726.

Dai, W., C. Boonyasiriwat, and G. T. Schuster, 2010, 3D multi-source least-squares reverse time migration: SEG Technical Program Expanded Abstracts, **29**, 3120–3124.

Dai, W. and J. Schuster, 2009, Least-squares migration of simultaneous sources data with a deblurring filter: SEG Technical Program Expanded Abstracts, **28**, 2990–2994.

Fei, W., P. Williamson, and A. Khoury, 2009, 3-D common-azimuth wave-equation migration velocity analysis: SEG Technical Program Expanded Abstracts, **28**, 2283–2287.

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

Krebs, J. R., J. E. Anderson, D. Hinkley, R. Neelamani, S. Lee, A. Baumstein, and M.-D. Lacasse, 2009, Fast full-wavefield seismic inversion using encoded sources: Geophysics, **74**, WCC177–WCC188.

Liu, F., D. W. Hanson, N. D. Whitmore, R. S. Day, and R. H. Stolt, 2006, Toward a unified analysis for source plane-wave migration: Geophysics, **71**, no. 4, S129–S139.

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

Sava, P. and I. Vlad, 2008, Numeric implementation of wave-equation migration velocity analysis operators: Geophysics, **73**, VE145–VE159.

Shen, P., 2004, Wave-equation migration velocity analysis by differential semblance optimization: PhD thesis, Rice University.

Shen, P., W. W. Symes, S. Morton, A. Hess, and H. Calandra, 2005, Differential semblance velocity analysis via shot profile migration: SEG Technical Program Expanded Abstracts, **24**, 2249–2252.

Soubaras, R. and B. Gratacos, 2007, Velocity model building by semblance maximization of modulated-shot gathers: Geophysics, **72**, U67–U73.

Stolt, R. H. and A. Benson, 1986, Seismic migration: Theory and practice: Geophysical Press.

Tang, Y., 2011, Imaging and velocity analysis by target-oriented wavefield inversion: PhD thesis, Stanford University.

Tang, Y. and B. Biondi, 2009, Least-squares migration/inversion of blended data: SEG Technical Program Expanded Abstracts, **28**, 2859–2863.

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

Tang, Y. and S. Lee, 2010, Preconditioning full waveform inversion with phase-encoded Hessian: SEG Technical Program Expanded Abstracts, **29**, 1034–1038.

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

Zhang, Y., J. Sun, C. Notfors, S. H. Gray, L. Chernis, and J. Young, 2005, Delayed-shot 3D depth migration: Geophysics, **70**, E21–E28.

APPENDIX A

EQUIVALENCE OF IMAGE-STACK-POWER MAXIMIZATION AND DATA-DOMAIN BORN WAVEFIELD INVERSION

This appendix shows that maximizing the image stack power (or minimizing its negative) is equivalent to Born wavefield inversion, which minimizes the difference between the modeled and observed primaries. The difference-based objective function for data-domain Born wavefield inversion can be defined as follows:

$$J = \frac{1}{2}(\mathbf{Lm} - \mathbf{d}_{\text{obs}})^*(\mathbf{Lm} - \mathbf{d}_{\text{obs}}), \tag{A-1}$$

where $\mathbf{d}_{\text{obs}}$ is the observed data vector, $\mathbf{m}$ is the reflectivity vector; $\mathbf{L}$ is the Born modeling operator that only modeled the angle stacked reflectivity (zero-subsurface-offset reflectivity), which is a function of the velocity vector $\mathbf{v}$. Objective function A-1 is minimized by optimizing both $\mathbf{v}$ and $\mathbf{m}$. Expanding equation A-1 yields

$$J = \frac{1}{2}\left(\mathbf{m}^*\mathbf{L}^*\mathbf{Lm} - \mathbf{m}^*\mathbf{L}^*\mathbf{d}_{\text{obs}} - \mathbf{d}^*_{\text{obs}}\mathbf{Lm} + \mathbf{d}^*_{\text{obs}}\mathbf{d}_{\text{obs}}\right). \tag{A-2}$$

In the least-squares sense, the reflectivity model $\mathbf{m}$ can be formally obtained as follows, assuming the Hessian $\mathbf{H}$ is invertible:

$$\mathbf{m} = (\mathbf{L}^*\mathbf{L})^{-1}\mathbf{L}^*\mathbf{d}_{\text{obs}} = \mathbf{H}^{-1}\mathbf{L}^*\mathbf{d}_{\text{obs}}. \tag{A-3}$$

Substituting equations A-3 into A-2 and simplifying yield

$$J = \frac{1}{2}\left(-\mathbf{d}^*_{\text{obs}}\mathbf{L}\mathbf{H}^{-1}\mathbf{L}^*\mathbf{d}_{\text{obs}} + \mathbf{d}^*_{\text{obs}}\mathbf{d}_{\text{obs}}\right) \tag{A-4}$$

Since $\mathbf{d}^*_{\text{obs}}\mathbf{d}_{\text{obs}}$ is a constant, it can be ignored in the above equation, therefore

$$J \approx -\frac{1}{2}\mathbf{d}^*_{\text{obs}}\mathbf{L}\mathbf{H}^{-1}\mathbf{L}^*\mathbf{d}_{\text{obs}}. \tag{A-5}$$

Note that the migration image $\mathbf{m}_{\text{mig}}$ is defined as follows:

$$\mathbf{m}_{\text{mig}} = \mathbf{L}^*\mathbf{d}_{\text{obs}}. \tag{A-6}$$

Substituting equations A-6 into A-5 yields

$$J \approx -\frac{1}{2}\mathbf{m}^*_{\text{mig}}\mathbf{H}^{-1}\mathbf{m}_{\text{mig}}. \tag{A-7}$$

To simplify the problem, I ignore the Hessian $\mathbf{H}$ in equation A-7 and assume it to be an identity matrix. Therefore, equation A-7 becomes

$$J \approx -\frac{1}{2}\mathbf{m}^*_{\text{mig}}\mathbf{m}_{\text{mig}}, \tag{A-8}$$

which is the same as equation 1 defined in the body of the paper. However, the Hessian $\mathbf{H}$ in equation A-7 might be important, especially in complex geologies, where the illumination is distorted by complex overburdens. The importance of the Hessian in equation A-7 remains an area for further investigation.

Stanford Exploration Project, SEP145, September 13, 2011

Preconditioned least-squares reverse-time migration using random phase encoding

Ali Almomin

ABSTRACT

Least-squares reverse-time migration (LSRTM) provides very accurate images of the subsurface. However, the computational cost of this technique is extremely high. One way to reduce that cost is to encode the sources using a random phase function and create a "super source". This encoding method introduces crosstalk artifacts that require averaging several realizations of the random encodings to suppress. I compare the convergence rates of the conventional and phase-encoded LSRTM for a fixed-spread geometry and show that the performance gain for the phase-encoded LSRTM far exceeds the loss due to the additional realizations. I also reduce the inversion cost by using the diagonal of the Hessian matrix as a preconditioner to the gradient. I also compare the convergence rates of different encoding methods used to estimate the true Hessian matrix. Then, I introduce a new source-based Hessian approximation and compare it to the other methods of approximating the Hessian matrix. Finally, I show the effect of each preconditioner on the LSRTM inversion. Results from the Marmousi synthetic model show that, for the same cost, preconditioning with the source-based Hessian gives the most accurate results.

INTRODUCTION

Reverse-time migration (RTM) uses the full wave equation to image the subsurface with high accuracy. However, RTM images suffer from several operator artifacts such as low-frequency noise, decreased resolution due to squaring the wavelet, and imbalanced amplitudes. These artifacts appear because, by migrating the data, we apply only the adjoint of the linear modeling operator, as opposed to its inverse. The imaging operator can be inverted in several ways, including iterative least-squares inversion. Although inversion can remove these artifacts, it is not widely used because the computational cost of applying the forward and adjoint operators in each iteration is extremely high.

Several methods have been proposed to reduce the computational cost of LSRTM. One of these methods is to reduce the data size by encoding the sources to create a super source (Morton and Ober, 1998; Jing et al., 2000; Romero et al., 2000; Sun et al., 2002). This technique has also been applied to full waveform inversion (Krebs et al., 2009; Gao et al., 2010; Boonyasiriwat and Schuster, 2010; Ben-hadj ali et al., 2011). The cost of applying the forward or adjoint of the modeling operator becomes independent of the number of sources, which greatly reduces the computational cost of the inversion. On the other hand, combining the sources causes crosstalk artifacts in the estimated image. These artifacts can be suppressed by changing the encoding function over iterations. Romero et al. (2000) and Krebs et al. (2009) showed that one-sample random phase encoding gives the best convergence rate.

In this paper, I compare the convergence rate of conventional LSRTM to phase-encoded LSRTM to test whether the computational reduction justifies the additional realizations of the encoding function. I first measure the norm of the model error of the two models. Then, I measure the norm of the model error after processing the estimated model at each iteration. The processing steps, which are a low-cut filter and an automatic gain control (AGC), are used to reduce the low-frequency noise and amplitude imbalance in order to measure the error in the image resolution.

The convergence rate of LSRTM inversion can be accelerated by preconditioning the gradient with the diagonal of the Hessian matrix. The Hessian matrix can be estimated by applying the encoding function either to the receiver side only or to both the receiver and source sides (Tang, 2009). A cheaper approximation to the Hessian matrix is the source intensity function, which ignores the receiver side of the Hessian matrix (Tang and Lee, 2010). The source intensity function can also be encoded along the source axis. I compare the convergence rates of these methods and also introduce a new approximation to the Hessian matrix that is based on the blended source-wavefield only. Finally, I test each of these preconditioners in the LSRTM and compare the results as a function of their cost.

METHOD

Least-squares RTM

The two-way wave equation is linearized over the slowness squared as follows:

$$s^2(\mathbf{x}) = b(\mathbf{x}) + m(\mathbf{x}), \tag{1}$$

where s is the slowness, b is the background model, which is a smooth version of the slowness squared, m is the model, and $\mathbf{x}$ is the model coordinate. Then, the Green's functions that satisfy the acoustic wave equation using the background model are defined as follows:

$$\left(\nabla^2 + \omega^2 b(\mathbf{x})\right) G_0(\mathbf{x}, \mathbf{x}_s, \omega) = -\delta(\mathbf{x} - \mathbf{x}_s), \tag{2}$$

$$\left(\nabla^2 + \omega^2 b(\mathbf{x})\right) G_0(\mathbf{x}, \mathbf{x}_r, \omega) = -\delta(\mathbf{x} - \mathbf{x}_r), \tag{3}$$

where G_0 is the Green's function, $\mathbf{x}_s$ and $\mathbf{x}_r$ are the source and receiver coordinates, and ω is frequency. The forward modeling operator is defined using the Green's functions as follows:

$$d(\mathbf{x}_r, \mathbf{x}_s, \omega) = -\omega^2 f(\omega) \sum_{\mathbf{x}} G_0(\mathbf{x}, \mathbf{x}_s, \omega) G_0(\mathbf{x}, \mathbf{x}_r, \omega) m(\mathbf{x}), \tag{4}$$

where d is the surface data and f is the source function. The forward modeling operator $\mathbf{F}$ can be written in matrix form as follows:

$$\mathbf{d} = \mathbf{F}\mathbf{m}. \tag{5}$$

We now define the objective function J as:

$$J(\mathbf{m}) = \|\mathbf{F}\mathbf{m} - \mathbf{d}_{\text{obs}}\|_2^2, \tag{6}$$

where $\mathbf{d}_{\text{obs}}$ is the observed surface data. The quadratic objective function J can be minimized iteratively using the following scheme (Claerbout and Fomel, 2011):

```
$\mathbf{r} \longleftarrow \mathbf{Fm} - \mathbf{d}_{\text{obs}}$
iterate {
        $\Delta\mathbf{m} \longleftarrow \mathbf{F}^*\mathbf{r}$
        $\Delta\mathbf{r} \longleftarrow \mathbf{F}\Delta\mathbf{m}$
        $(\mathbf{m}) \longleftarrow \text{stepper}(\mathbf{m}, \mathbf{r}, \Delta\mathbf{m}, \Delta\mathbf{r})$
        }
```

The * indicates the adjoint operator. The cost of each iteration equals the cost of the forward and adjoint operators. The stepper function is either steepest-descent or conjugate gradient.

There are several encoding functions that can be used in LSRTM (Perrone and Sava, 2009; Godwin and Sava, 2011). However, a single-sample random phase function gives the best convergence results (Romero et al., 2000; Krebs et al., 2009). This encoding function results in crosstalk artifacts in the estimated models. These artifacts are reduced by averaging several realizations of the encoding function. The source-side encoding function α is defined as follows (Tang, 2009):

$$\alpha(\mathbf{x}_s, p_s) = \frac{1}{\sqrt{N_{\text{realize}}}} \gamma(\mathbf{x}_s, p_s), \tag{7}$$

where p_s is the realization index, N_{realize} is the number of realizations, and γ is a random sequence of signs (i.e. +1 and -1). The encoding function is used to blend the observed data as follows:

$$\widetilde{d}_{\text{obs}}(\mathbf{x}_r, p_s, \omega) = \sum_{\mathbf{x}_s} \alpha(\mathbf{x}_s, p_s) d_{\text{obs}}(\mathbf{x}_r, \mathbf{x}_s, \omega). \tag{8}$$

Similarly, the same encoding function is used to blend the modeled data:

$$S(\mathbf{x}, p_s, \omega) = \sum_{\mathbf{x}_s} \alpha(\mathbf{x}_s, p_s) f(\omega) G_0(\mathbf{x}, \mathbf{x}_s, \omega), \tag{9}$$

where S is the blended source wavefield. Due to the linearity of the wave equation, this wavefield can be simply computed by simultaneously injecting the source functions at different locations after multiplying by the proper weight. Once S is computed, the blended forward modeling operator can defined as follows:

$$\widetilde{d}(\mathbf{x}_r, p_s, \omega) = -\omega^2 \sum_{\mathbf{x}} S(\mathbf{x}, p_s, \omega) G_0(\mathbf{x}, \mathbf{x}_r, \omega) m(\mathbf{x}), \tag{10}$$

where$\widetilde{\ }$is used to indicate blending. The blended forward modeling operator $\widetilde{\mathbf{F}}$ can also be expressed in matrix form:

$$\widetilde{\mathbf{d}} = \widetilde{\mathbf{F}}\mathbf{m}. \tag{11}$$

Finally, the objective function of the blended operator can be written as follows:

$$\widetilde{J}(\mathbf{m}) = \|\widetilde{\mathbf{F}}\mathbf{m} - \widetilde{\mathbf{d}}_{\text{obs}}\|_2^2. \tag{12}$$

Notice that the objective functions changes between realizations, since the encoding function changes. This change of the objective function requires a modification in the minimization scheme as follows:

```
iterate {
        $\widetilde{\mathbf{r}} \longleftarrow \widetilde{\mathbf{F}}\mathbf{m} - \widetilde{\mathbf{d}}_{\mathrm{obs}}$
        $\Delta\mathbf{m} \longleftarrow \widetilde{\mathbf{F}}^{*}\widetilde{\mathbf{r}}$
        $\Delta\widetilde{\mathbf{r}} \longleftarrow \widetilde{\mathbf{F}}\Delta\mathbf{m}$
        $\mathbf{m} \longleftarrow \mathrm{stepper}(\mathbf{m}, \Delta\mathbf{m}, \Delta\widetilde{\mathbf{r}})$
        }
```

There are two changes in the minimization scheme of the blended objective function compared to the conventional one. First, the computation of the residual is moved inside the loop, because the encoding function changes in each iteration. This change adds the cost of a forward modeling operator to each iteration. Second, the stepper algorithm can only be steepest-descent if the step size is determined with linear optimization. Otherwise, a non-linear conjugate gradient can be performed, requiring a line search in each iteration. In this paper I present only the result of using steepest-descent stepper, because the iteration cost is consistent.

Hessian Estimation

For any linear operator $\mathbf{F}$, the Hessian matrix is computed as follows:

$$H(\mathbf{x}, \mathbf{y}) = \mathbf{F}^{*}(\mathbf{y})\mathbf{F}(\mathbf{x}), \tag{13}$$

where $\mathbf{x}$ and $\mathbf{y}$ are model coordinates. There are several ways to utilize the Hessian matrix in the inversion process, but I will focus on using its diagonal as a preconditioner to the gradient:

$$\mathbf{s}_k = \frac{\mathbf{g}_k}{\mathrm{diag}\{\mathbf{H}\} + \epsilon\mathbf{I}}, \tag{14}$$

where $\mathbf{g}_k$ is gradient at the k^{th} iteration, $\mathbf{s}_k$ is the preconditioned search direction, $\mathbf{I}$ is an identity operator, and ϵ is a constant used to stabilize the division. For the modeling operator, the diagonal of the Hessian matrix can be written as follows:

$$\mathrm{diag}\{\mathbf{H}\} = D(\mathbf{x}) = \sum_{\omega} \omega^4 |f(\omega)|^2 \sum_{\mathbf{x}_s} |G_0(\mathbf{x}, \mathbf{x}_s, \omega)|^2 \sum_{\mathbf{x}_r} |G_0(\mathbf{x}, \mathbf{x}_r, \omega)|^2. \tag{15}$$

Unlike with forward and adjoint modeling operators, the computations must be done on each source-receiver pair separately. As with LSRTM, the expense of this operation can be reduced by encoding the source or receiver side, or both sides. I first define a receiver-side encoding function β as follows:

$$\beta(\mathbf{x}_r, p_r) = \frac{1}{\sqrt{N_{\mathrm{realize}}}} \gamma(\mathbf{x}_r, p_r), \tag{16}$$

where p_r is the realization index, and the other variables are the same as in the encoding function α. I now define an encoded receiver wavefield:

$$R(\mathbf{x}, p_r, \omega) = \sum_{\mathbf{x}_r} \beta(\mathbf{x}_r, p_r) G_0(\mathbf{x}, \mathbf{x}_r, \omega). \tag{17}$$

This encoding results in the receiver-side blended function $\widetilde{\mathbf{D}}$, which can be written as follows:

$$\widetilde{D}(\mathbf{x}, p_r) = \sum_{\omega} \omega^4 |f(\omega)|^2 \sum_{\mathbf{x}_s} |G_0(\mathbf{x}, \mathbf{x}_s, \omega) R(\mathbf{x}, p_r, \omega)|^2. \tag{18}$$

The cost of one realization of the function $\widetilde{\mathbf{D}}$ is equivalent to an unblended migration of all the shots. Additionally, the source side can also be blended:

$$\widetilde{\widetilde{D}}(\mathbf{x}, p_s, p_r) = \sum_{\omega} \omega^4 |S(\mathbf{x}, p_s, \omega) R(\mathbf{x}, p_r, \omega)|^2. \tag{19}$$

The cost of one realization of the function $\widetilde{\widetilde{\mathbf{D}}}$ is equivalent to migrating one shot only. However, the additional blending results in more crosstalk. Hence, the function $\widetilde{\widetilde{\mathbf{D}}}$ requires more realizations to reduce the crosstalk artifacts than does the function $\widetilde{\mathbf{D}}$.

Although the cost of computing the function $\mathbf{D}$ can be reduced with blending, additional propagation(s) are still required to compute the receiver side. Therefore, preconditioning with the source intensity function $\mathbf{D}_S$ can be done by ignoring the receiver side of the Hessian matrix. The source intensity function can be written as follows:

$$D_S(\mathbf{x}) = \sum_{\omega} \omega^4 |f(\omega)|^2 \sum_{\mathbf{x}_s} |G_0(\mathbf{x}, \mathbf{x}_s, \omega)|^2. \tag{20}$$

The previous formulation computes the exact source function in one iteration of LSRTM. However, if the inversion is done with the blended operator, the source intensity function can be computed using the blended source wavefield as follows:

$$\widetilde{D}_S(\mathbf{x}, p_s) = \sum_{\omega} \omega^4 |S(\mathbf{x}, \mathbf{p}_s, \omega)|^2. \tag{21}$$

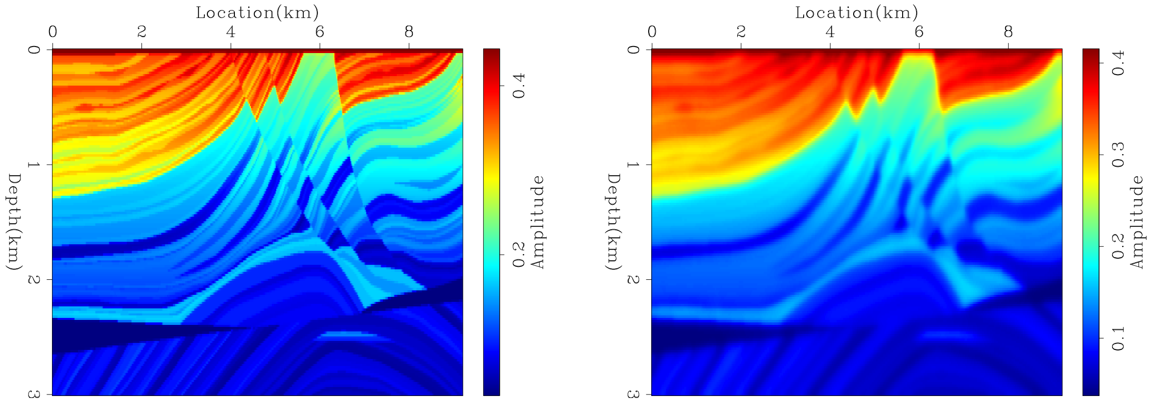
By comparing equation (21) to equation (19), we can see that ignoring the receiver side in the Hessian matrix can be physically interpreted as having receivers everywhere in the subsurface. As a result, the source intensity function overestimates the Hessian matrix. Therefore, I propose a better approximation to the Hessian matrix by using the blended source wavefiled to approximate the receiver-side. This source-based Hessian can be written as follows:

$$\widetilde{D}_{SS}(\mathbf{x}, p_s) = \sum_{\omega} \omega^4 |S(\mathbf{x}, \mathbf{p}_s, \omega)|^4. \tag{22}$$

The function $\widetilde{\mathbf{D}}_{SS}$ approximates the receiver wavefield by the source wavefield. Physically, this assumes that the receivers exist at the same location as the sources. This is a better approximation than the original source intensity function, especially for the fixed-spread geometry. This formulation requires no additional propagation if the source side is blended. However, there are two sources of error in equation (22). First, the receiver spacing could be different than the source spacing, even if their spreads cover the same area. Second, the receiver side should have a different encoding function than the source side. These errors will prevent the source-based Hessian from approaching the true Hessian matrix, regardless of the number of realizations.

SYNTHETIC EXAMPLES

A decimated Marmousi model is used for the synthetic examples. The sampling for both spatial axes is 20 m. A Ricker wavelet with a fundamental frequency of 15 Hz and temporal sampling of 1.5 ms is used to model the data. The receiver spacing is 20 m, and the source



(a) (b)

Figure 1: (a) Marmousi slowness squared and (b) the background model. [**ER**] ali1/. marmousi,marmousi-smooth

spacing is 100 m. Born modeling with a time-domain finite-difference propagator was used in both the observed and the calculated data.

Figures 1(a) and 1(b) show the Marmousi slowness squared and the background model, respectively. The reflectivity model, obtained by subtracting the Marmousi slowness squared from the background model, is shown in Figure 2.

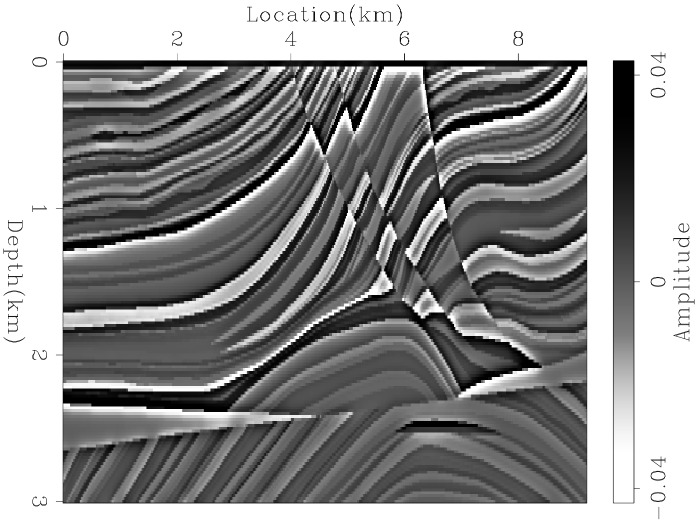
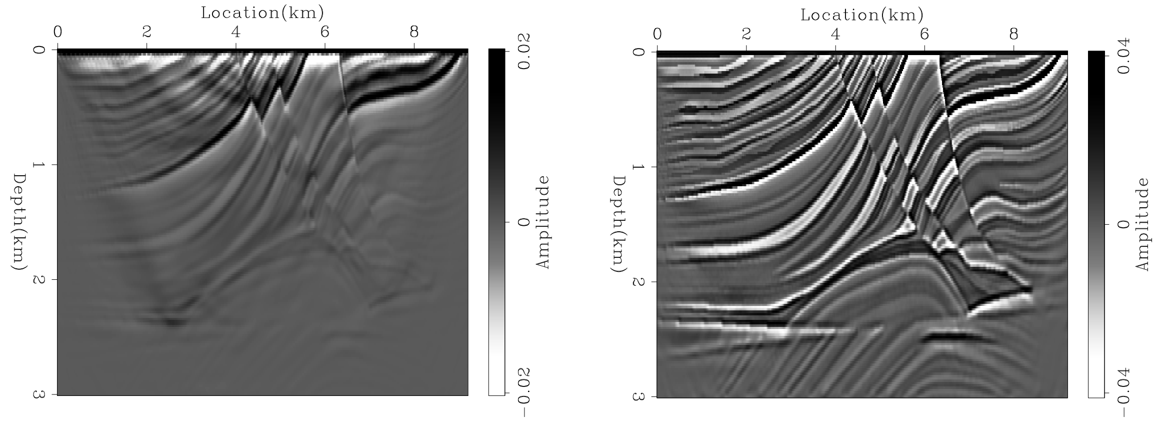


Figure 2: The Marmousi reflectivity model. [**ER**] ali1/. marmousi-refl

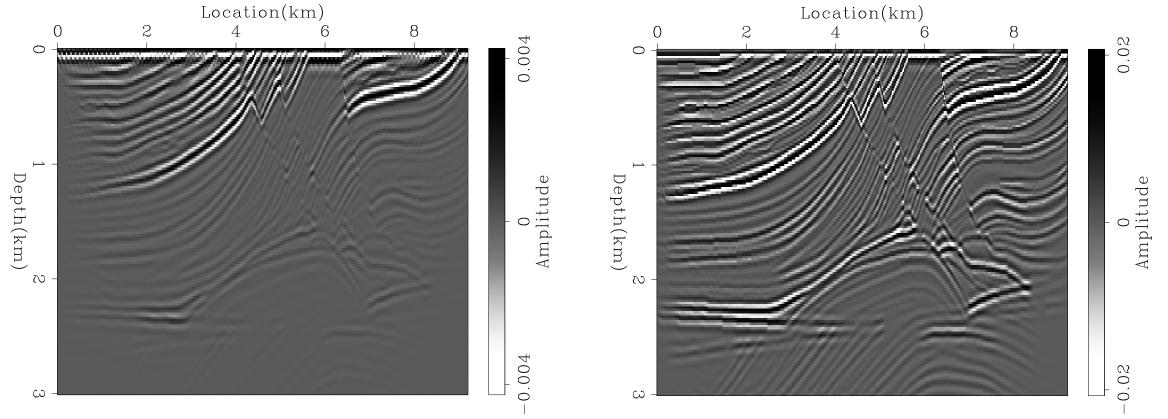
First, I ran a conventional LSRTM without any preconditioning for 50 iterations. The result of the fifth iteration is shown in Figure 3(a). This result is dominated by low-frequency noise as well as strong imbalance in amplitudes of the model. Next, I repeated the experiment, this time blending the sources. Figure 3(b) shows the result after 310 iterations, which is equivalent in cost to five iterations of unblended LSRTM. By comparing Figures 3(a) and 3(b), we see the blended LSRTM gives much better results.



(a) (b)

Figure 3: (a) Conventional LSRTM after 5 iterations and (b) blended LSRTM after 310 iterations. [**ER**] ali1/. marmousi-lsrtmcd,marmousi-lsrtmb

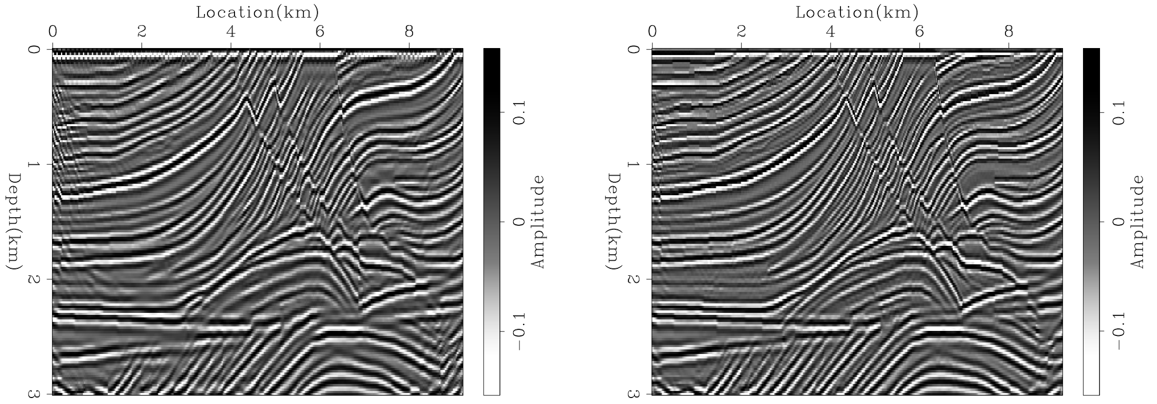
In order to determine whether the difference between the two results is only caused by the low frequency noise, I applied a low-cut filter to both results, as shown in Figures 4(a) and 4(b). Next, I further processed the results to remove the amplitude imbalance by applying an AGC, as shown in Figures 5(a) and 5(b). For comparison, Figures 6(a) and 6(b) show the true reflectivity model after applying the same processing steps. We can see that the blended LSRTM gives sharper and more accurate images even after processing.



(a) (b)

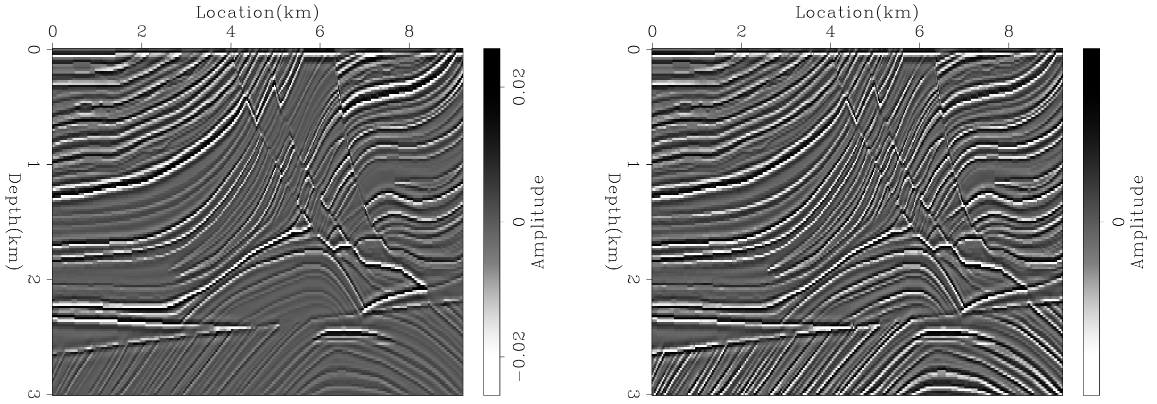
Figure 4: The result of applying a low-cut filter to Figures 3(a) and 3(b). [**ER**] ali1/. marmousi-lsrtmcd-bp,marmousi-lsrtmb-bp

For a more accurate measure of error, I computed the RMS error between the true reflectivity model and the result of each iteration before processing. Figure 7(a) shows three curves: the unblended LSRTM using steepest-descent stepper (LSRTM-SD), the unblended LSRTM using conjugate-direction stepper (LSRTM-CD), and blended LSRTM using steepest-descent stepper (B-LSRTM). It is interesting to see that the blended LSRTM is converging at a similar rate to the unblended LSRTM with the steepest-descent stepper,



(a) (b)

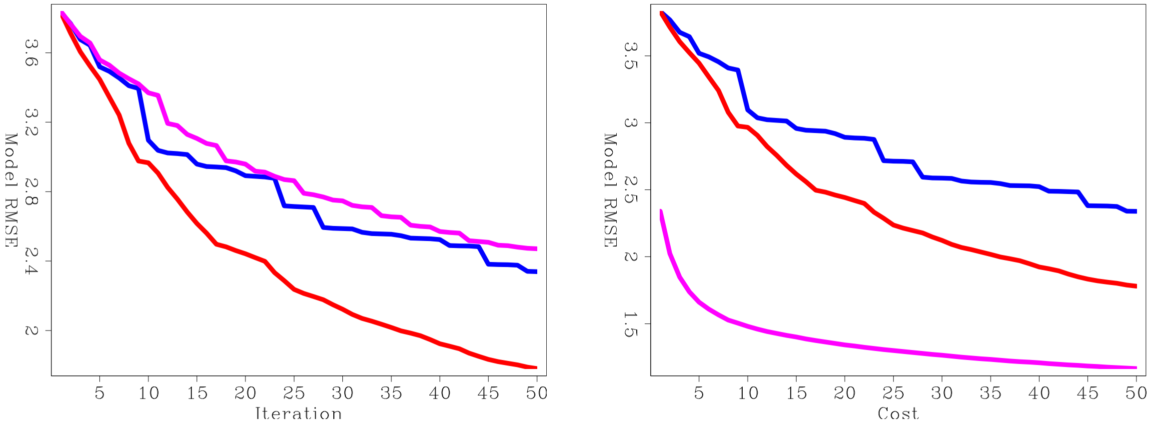
Figure 5: The result of applying an AGC to Figures 4(a) and 4(b). [**ER**] ali1/. marmousi-lsrtmcd-agc,marmousi-lsrtmb-agc



(a) (b)

Figure 6: The result of applying (a) a low-cut filter and (b) an AGC to Figure 2. [**ER**] ali1/. marmousi-refl-bp,marmousi-refl-agc

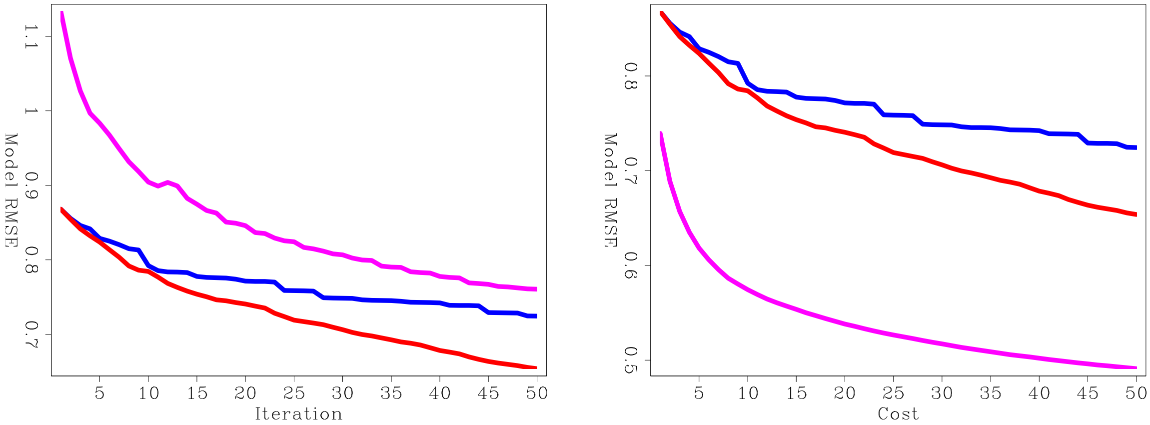
although their costs are very different. This seems to indicate that while suppressing the crosstalk, the blended LSRTM is also inverting the operator without much loss of efficiency. The only advantage for unblended LSRTM seems to result from having a better stepper.



(a) (b)

Figure 7: The model RMS error versus (a) iteration and (b) migration cost. LSRTM-SD is blue, SLRTM-CD is red, and B-LSRTM is purple. [**CR**] ali1/. lsrtm-obj,lsrtm-obj-cost

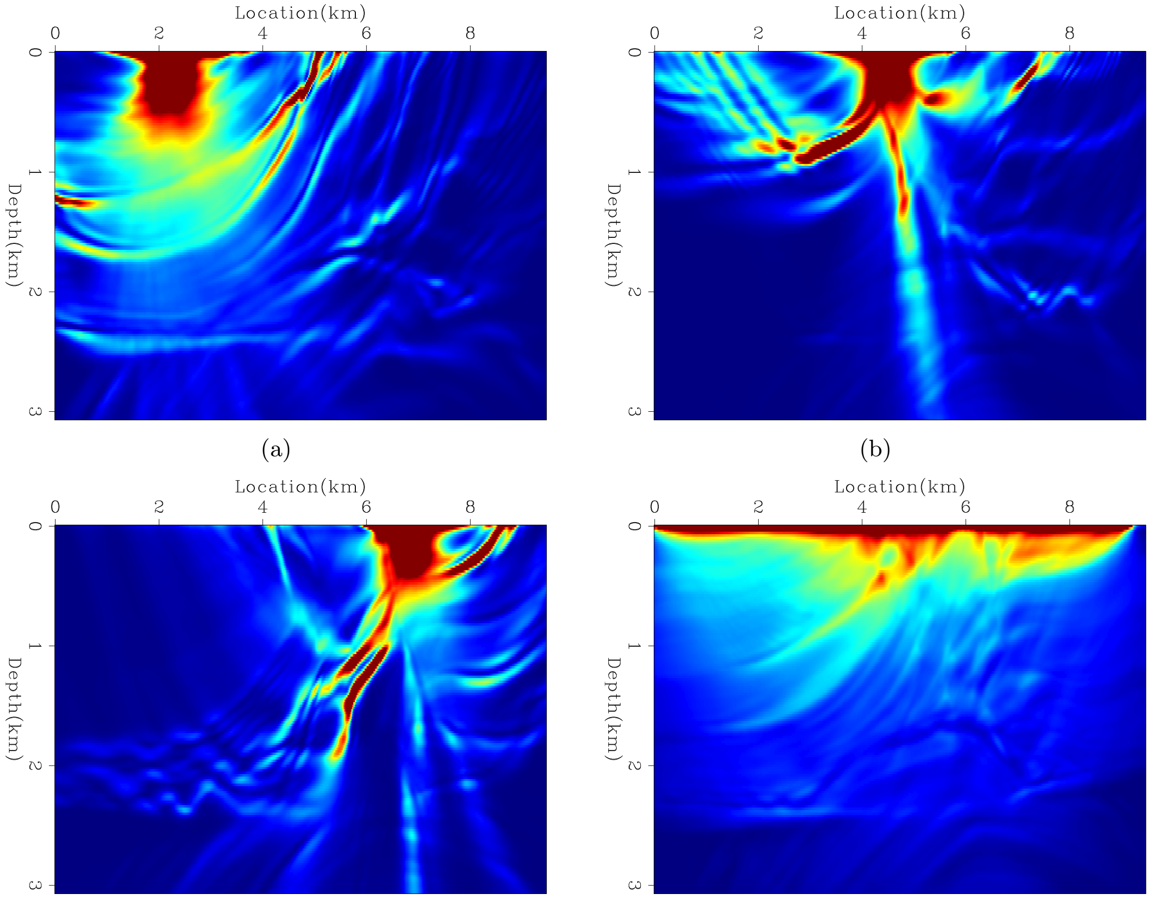
Figure 7(b) shows the RMS error curves as a function of cost. The cost unit is equivalent to a conventional RTM of all the shots. This Figure clearly shows that for the same cost, the B-LSRTM gives much better results than the conventional LSRTM regardless of the stepper algorithm.



(a) (b)

Figure 8: The same as Figures 7(a) and 7(b) but after processing the inversion results with a low-cut filter and AGC. LSRTM-SD is blue, SLRTM-CD is red, and B-LSRTM is purple. [**CR**] ali1/. lsrtm-obj-agc,lsrtm-obj-agc-cost

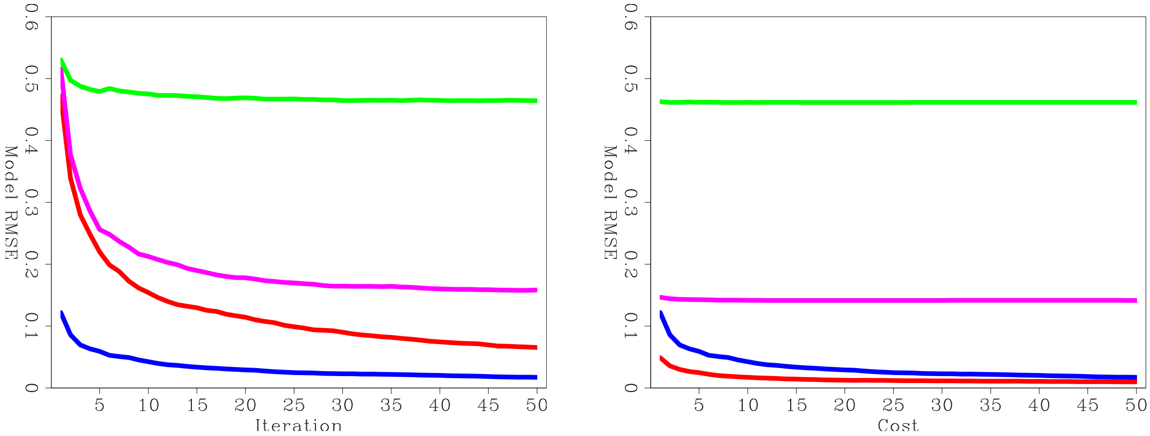
Figure 8(a) is the same as Figure 7(a), but after processing the inversion results. The B-LSRTM seems to have a much larger initial error because the crosstalk artifacts, which are high frequency, are amplified by applying a low-cut filter. However, by comparing the results at the same cost, as shown in Figure 8(b), the B-LSRTM is still superior to the other unblended LSRTM inversions.



(a) (b)

(c) (d)

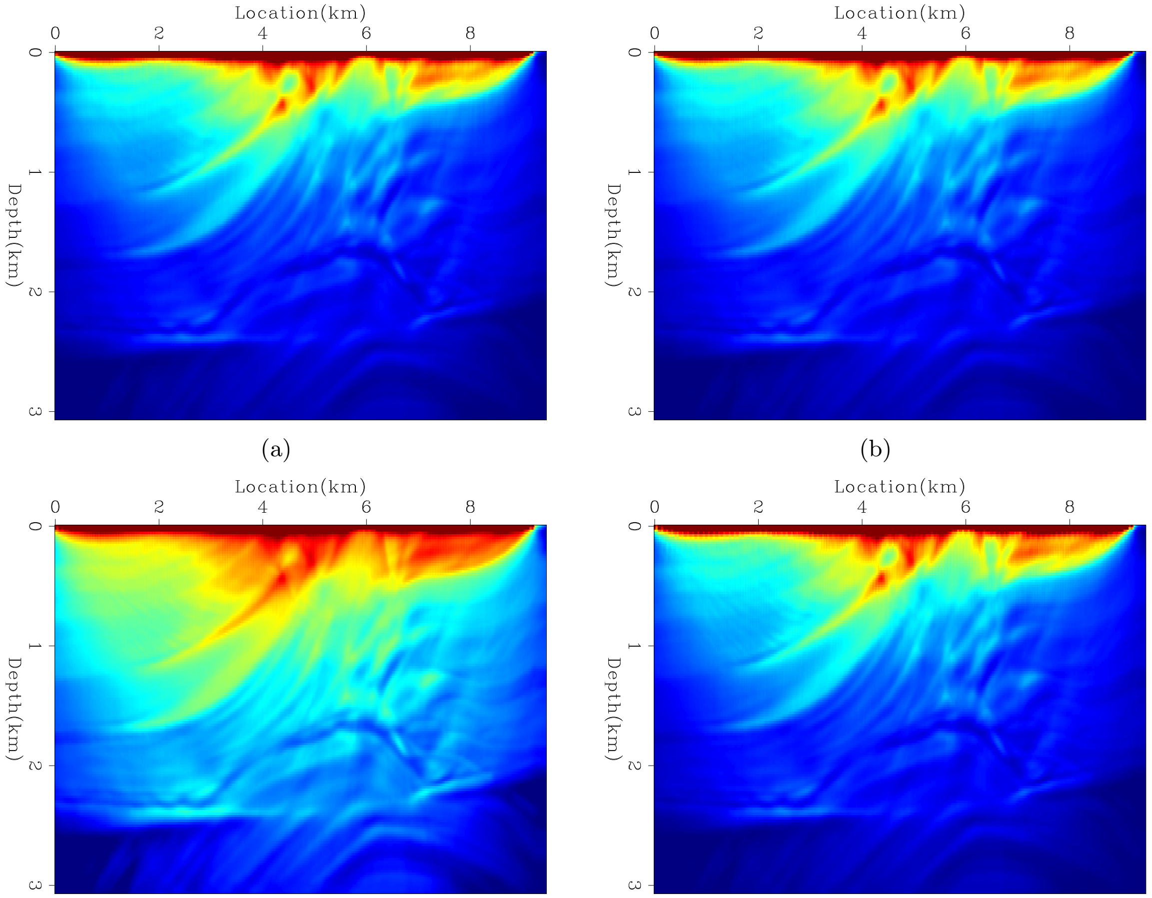
Figure 9: (a), (b), and (c) show the contributions of three different shots and (d) shows the total diagonal of the Hessian matrix. [**CR**] ali1/. hess1,hess2,hess3,hess



(a) (b)

Figure 10: The model RMS error of (a) every iterations/realization and (b) every migration cost. HRB is blue, HSRB is red, SISB is green, SHSB and is purple. [**CR**] ali1/. hess-obj,hess-obj-cost

Next, I computed the exact diagonal of the Hessian matrix. Figures 9(a), 9(b), and 9(c) shows the contribution from three different shots, and Figure 9(d) shows the total diagonal of the Hessian matrix. I then tested the convergence rates of four preconditioners: the Hessian matrix with receiver-side blending (HRB), the Hessian matrix with source- and receiver-side blending (HSRB), the source intensity function with source-side blending (SISB), and the source-based Hessian with source-side blending (SHSB).



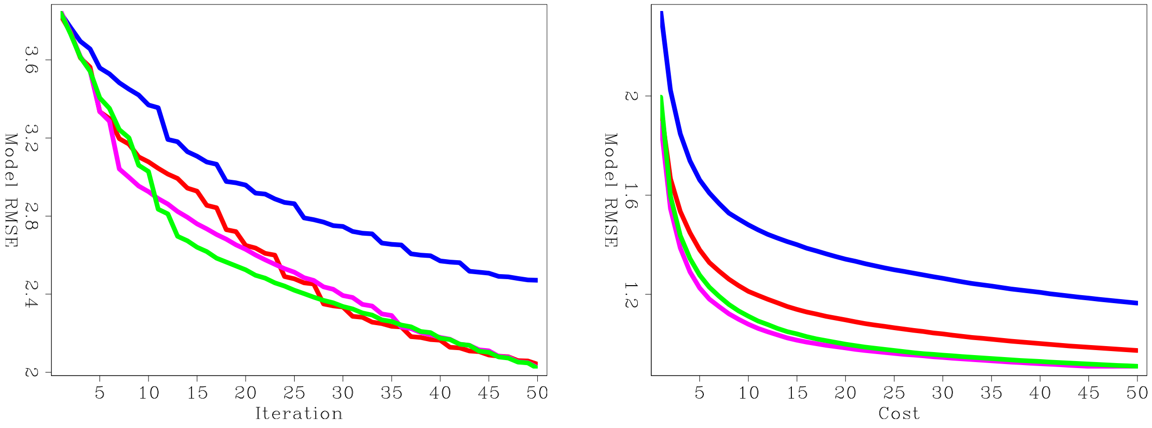
(a) (b)

(c) (d)

Figure 11: The results after 50 migrations equivalent cost of (a) HRB, (b) HSRB, (c) SISB, and (d) SHSB. [**CR**] ali1/. hess-hrb,hess-hsrb,hess-sisb,hess-shsb

Figures 10(a) and 10(b) show the RMS error of each preconditioner compared to the exact Hessian diagonal versus iteration and versus cost, respectively. Both HRB and HSRB approach the true Hessian diagonal, but encoding both sides gives a faster convergence rate per cost. The SISB converges the fastest, but it converges to a solution with a large error. On the other hand, SHSB converges to a much better solution and at a similar rate to SISB. The preconditioners after 50 migrations equivalent cost are shown in Figures 11(a), 11(b), 11(c), and 11(d).

Finally, I ran three LSRTM inversions using HRSB, SISB, and SHSB as preconditioners and compared them to the unpreconditioned B-LSRTM. Figure 12(a) shows the model RMS error of the four curves as a function of iteration. The three preconditioned curves seems

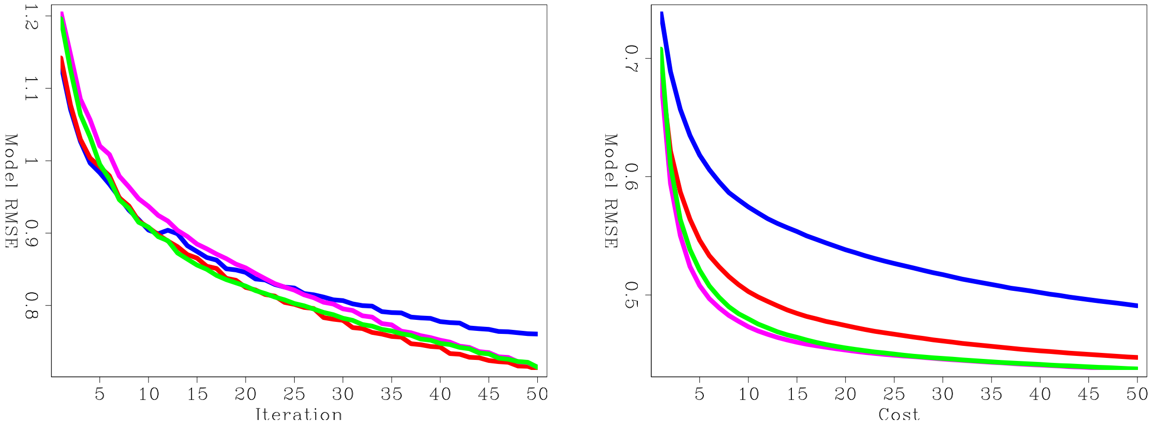


(a) (b)

Figure 12: The model RMS error versus (a) iteration and (b) migration cost. No preconditioning is blue, HRSB is green, SISB is red, SHSB and is purple. [**CR**] ali1/. lsrtm-objp,lsrtm-objp-cost

to converge at a simiar rate per iteration. However, the convergence rates per cost are different, as shown in Figure 12(b). Although SHSB had larger error in estimating the Hessian matrix, it resulted in the best convergence rate for LSRTM due to its cheaper cost. The second-best method was HRSB.

As with the previous LSRTM, I processed the results of preconditioned LSRTM and compared it to the processed, true reflectivity. Figures 13(a) and 13(b) show the model RMS error versus iteration and cost, respectively. Processing the models did not change the order of the curves, but, as expected, it reduced the difference between them in the early iterations.



(a) (b)

Figure 13: The same as Figures 12(a) and 12(b) but after processing the inversion results with a low-cut filter and AGC. No preconditioning is blue, HRSB is green, SISB is red, SHSB and is purple. [**CR**] ali1/. lsrtm-objp-agc,lsrtm-objp-agc-cost

DISCUSSION AND CONCLUSIONS

By examining the results using the Marmousi model, we see that encoding the sources in LSRTM is more efficient than the conventional method. This is true despite the fact that each iteration requires an additional forward operator and the stepper is not as efficient.

When estimating the Hessian matrix, encoding both the source and receiver sides is more efficient than encoding the receiver side only. Moreover, the source-based Hessian is more accurate than the conventional source intensity function. As a preconditioner, the source-based Hessian is the most efficient due to its cheaper computational cost compared to estimating both sides of the Hessian matrix. However, using this source-based Hessian in different acquisition geometries needs to be further tested.

REFERENCES

Ben-hadj ali, H., S. Operto, and J. Virieux, 2011, An efficient frequency-domain full waveform inversion method using simultaneous encoded sources: Geophysics, **76**, R109–R124.

Boonyasiriwat, C. and G. T. Schuster, 2010, 3D Multisource Full-Waveform Inversion using Dynamic Random Phase Encoding: SEG Technical Program Expanded Abstracts, 1044–1049.

Claerbout, J. F. and S. Fomel, eds., 2011, Image estimation by example: Geophysical soundings image construction.

Gao, F., A. Atle, and P. Williamson, 2010, Full waveform inversion using deterministic source encoding: SEG Technical Program Expanded Abstracts, 1013–1017.

Godwin, J. and P. C. Sava, 2011, A comparison of shot-encoding schemes for wave-equation migration: SEG Technical Program Expanded Abstracts, 32–36.

Jing, X., C. J. Finn, T. a. Dickens, and D. E. Willen, 2000, Encoding multiple shot gathers in prestack migration: SEG Technical Program Expanded Abstracts, 786–789.

Krebs, J. R., J. E. Anderson, D. Hinkley, R. Neelamani, S. Lee, A. Baumstein, and M.-D. Lacasse, 2009, Fast full-wavefield seismic inversion using encoded sources: Geophysics, **74**, WCC177–WCC188.

Morton, S. A. and C. C. Ober, 1998, Faster shot-record depth migration using phase encoding: SEG Technical Program Expanded Abstracts.

Perrone, F. and P. C. Sava, 2009, Comparison of shot encoding functions for reverse-time migration: SEG Technical Program Expanded Abstracts, 2980–2984.

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

Sun, P., S. Zhang, and F. Liu, 2002, Prestack migration of areal shot records with phase encoding: SEG Technical Program Expanded Abstracts, 1172–1175.

Tang, Y., 2009, Target-oriented wave-equation least-squares migration/inversion with phase-encoded Hessian: Geophysics, **74**, WCA95–WCA107.

Tang, Y. and S. Lee, 2010, Preconditioning full waveform inversion with phase-encoded Hessian: SEG Technical Program Expanded Abstracts, 1034–1038.

Least-squares wave-equation inversion of time-lapse seismic data sets – A Valhall case study

Gboyega Ayeni and Biondo Biondi

ABSTRACT

We demonstrate an application of least-squares wave-equation inversion using time-lapse data sets from the Valhall field. We pose time-lapse imaging as a joint least-squares problem that utilizes target-oriented approximations to the Hessian of the objective function. Because this method accounts for illumination mismatches—caused by differences in acquisition geometries—and for band-limited wave-propagation effects, it provides better estimates of production-related changes in reservoir acoustic properties than conventional time-lapse processing methods. We show that our method improves image resolution (compared to migration) and that it attenuates obstruction artifacts in time-lapse images.

INTRODUCTION

Reservoir rock and fluid property changes can be obtained from seismic amplitude and/or travel-time changes. There is a wide range of published work on the most important considerations for time-lapse seismic imaging. For example, Batzle and Wang (1992) and Mavko et al. (2003) outline important rock and fluid relationships; Lumley (1995), Rickett and Lumley (2001), Calvert (2005), and Johnston (2005) discuss important processing and practical applications; and Lefeuvre et al. (2003), Whitcombe et al. (2004), Zou et al. (2006) and Ebaid et al. (2009) present successful case studies. Because of the recorded successes, time-lapse seismic imaging is now an integral part of many reservoir management projects.

In practice, production-related changes in time-lapse seismic images can be masked by non-repeatability artifacts (e.g., changes in geometry, ambient noise) or by effects of complex overburden (e.g., salt canopy). To correctly interpret time-lapse seismic differences, such artifacts must be attenuated—a prerequisite conventionally achieved through image cross-equalization methods (Rickett and Lumley, 2001). Although cross-equalization methods are well developed and provide reliable results in many practical applications, they are inadequate where large inconsistencies exist between the geometries used to acquire the data sets or where the reservoir overburden is complex. Where these conventional methods fail, wave-equation inversion provides a way to attenuate unwanted artifacts in time-lapse images, thereby enhancing production-related changes.

The proposed method is based on linear least-squares migration/inversion of seismic data sets (Nemeth et al., 1999; Kühl and Sacchi, 2003; Clapp, 2005). Because each iteration of a data-space implementation of least-squares migration/inversion is approximately twice the migration cost, this approach is expensive. However, by posing this problem in the image space, it can be efficiently solved in a target-oriented manner (Valenciano et al., 2006; Tang, 2009). For the time-lapse imaging problem, we can either invert for the complete baseline and monitor images or invert for a static baseline and time-lapse images between surveys. Inputs in the resulting formulations are migrated images (or combinations thereof) and the outputs are the inverted images (or time-lapse images). The operators are a concatenation of target-oriented approximations to the Hessian of the least-squares objective function (Ayeni and Biondi, 2010). We regularize the inversion using spatial (dip) and temporal (difference) constraints. Because we assume that the data contain only primaries, robust multiple/noise attenuation and data preprocessing is required prior to inversion. Furthermore, we assume compaction and velocity changes between surveys are small relative to the baseline; therefore the effects of these—which inherently neglected by migrating all data sets with the baseline velocity—can be removed by multidimensional warping of the monitor images to the baseline.

First, we summarize wave-equation inversion of time-lapse data sets. Then, we apply this method to a subset of the Valhall Life of Field Seismic (LoFS) data with a synthesized obstruction in the monitor. We show that the proposed method improves the image resolution (compared to migration) and that it attenuates obstruction artifacts in time-lapse images.

METHODOLOGY

Given a linearized modeling operator $\mathbf{L}$, the seismic data $\mathbf{d}$ for survey i due to a reflectivity model $\mathbf{m}$ is

$$\mathbf{d}_i = \mathbf{L}_i \mathbf{m}_i. \tag{1}$$

Assuming we have two data sets (baseline $\mathbf{d}_0$ and monitor $\mathbf{d}_1$) acquired at different times over an evolving reservoir, joint least-squares migration/inversion involves solving the following regression:

$$\left[\begin{array}{cc} \mathbf{L}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_1 \\ \hline \epsilon_0 \mathbf{R}_0 & \mathbf{0} \\ \mathbf{0} & \epsilon_1 \mathbf{R}_1 \\ \hline -\zeta_0 \Lambda_0 & \zeta_1 \Lambda_1 \end{array}\right] \left[\begin{array}{c} \mathbf{m}_0 \\ \mathbf{m}_1 \end{array}\right] \approx \left[\begin{array}{c} \mathbf{d}_0 \\ \mathbf{d}_1 \\ \hline \mathbf{0} \\ \mathbf{0} \\ \hline \mathbf{0} \end{array}\right], \tag{2}$$

where $\mathbf{R}_i$ and Λ_i are the spatial and temporal regularization operators respectively, and ϵ_i and ζ_i are the corresponding regularization parameters. Although we can directly solve equation 2 by minimizing the quadratic-norm of the regression (Ajo-Franklin et al., 2005), we choose to transform it to an image space problem of the form (Ayeni and Biondi, 2011)

$$\left[\begin{array}{cc} \mathbf{H_0} & \mathbf{0} \\ \mathbf{0} & \mathbf{H_1} \\ \hline \mathbf{R_{00}} & \mathbf{0} \\ \mathbf{0} & \mathbf{R_{11}} \\ \hline \Lambda_{00} & -\Lambda_{01} \\ -\Lambda_{10} & \Lambda_{11} \end{array}\right] \left[\begin{array}{c} \hat{\mathbf{m}}_0 \\ \hat{\mathbf{m}}_1 \end{array}\right] \approx \left[\begin{array}{c} \tilde{\mathbf{m}}_0 \\ \tilde{\mathbf{m}}_1 \\ \hline \mathbf{0} \\ \mathbf{0} \\ \hline \mathbf{0} \\ \mathbf{0} \end{array}\right], \tag{3}$$

where $\mathbf{H}_i = \mathbf{L}_i^T \mathbf{L}_i$ is the wave-equation Hessian, and $\mathbf{R}_{ii} = \epsilon_i^2 \mathbf{R}_i^T \mathbf{R}_i$ and $\Lambda_{ij} = \zeta_i {\Lambda_i}^T \zeta_j \Lambda_j$ are the spatial and temporal constraints. The inverted time-lapse image $\Delta\hat{\mathbf{m}}$ is then the difference between the inverted baseline and monitor images ($\hat{\mathbf{m}}_0$ and $\hat{\mathbf{m}}_1$). Equation 3 can be extended to multiple seismic data sets (Ayeni and Biondi, 2010). Alternatively, we can re-write equation 3 to invert directly for the time-lapse image and a static baseline image (Ayeni and Biondi, 2011). Due to physical movements of reflectors and velocity changes (e.g., due to reservoir depletion and compaction) between surveys, the baseline and monitor images will not be aligned. Such misalignments must be accounted for before or during inversion. As is the case in many practical time-lapse monitoring problems, we assume that the monitor data are migrated with the baseline velocity, which has been estimated to a high accuracy. However this method can be applied where an accurate monitor velocity has been available. The updated inversion problem is then given by (Ayeni and Biondi, 2011)

$$\left[\begin{array}{cc} \mathbf{H}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_1^b \\ \hline \mathbf{R}_{00} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{11}^b \\ \hline \boldsymbol{\Lambda}_{00} & -\boldsymbol{\Lambda}_{01}^b \\ -\boldsymbol{\Lambda}_{10}^b & \boldsymbol{\Lambda}_{11}^b \end{array}\right] \left[\begin{array}{c} \hat{\mathbf{m}}_0 \\ \hat{\mathbf{m}}_1^b \end{array}\right] \approx \left[\begin{array}{c} \tilde{\mathbf{m}}_0 \\ \tilde{\mathbf{m}}_1^b \\ \hline \mathbf{0} \\ \mathbf{0} \\ \hline \mathbf{0} \\ \mathbf{0} \end{array}\right], \tag{4}$$

where $\tilde{\mathbf{m}}_1^b$ and $\hat{\mathbf{m}}_1^b$ are respectively the migrated and inverted monitor images repositioned (warped) to the baseline image. The superscript ^{b} on the operators denotes that they are referenced to the baseline image. For example, $\mathbf{H}_1^b$ is the Hessian computed with the monitor geometry but with the baseline velocity. Note that the conventional time-lapse image $\Delta\tilde{\mathbf{m}}^b$ estimated at the baseline position is given by

$$\Delta\tilde{\mathbf{m}}^b = \tilde{\mathbf{m}}_1^b - \tilde{\mathbf{m}}_0, \tag{5}$$

while the inverted time-lapse image $\Delta\hat{\mathbf{m}}^b$ is given by

$$\Delta\hat{\mathbf{m}}^b = \hat{\mathbf{m}}_1^b - \hat{\mathbf{m}}_0. \tag{6}$$

For any practical application, it is infeasible (and unnecessary) to compute the full Hessian matrix. Because the problem is posed in the image space, we only need to compute the Hessian for a target region of interest around the reservoir. In addition, we only compute off-diagonal elements sufficient to capture the dominant structure of the Hessian. This target-oriented approximation of the Hessian is given by (Valenciano et al., 2006)

$$\begin{aligned} H\left(\mathbf{x_T}, \mathbf{x_{T+a_x}}\right) = \sum_{\omega} \omega^4 \sum_{\mathbf{x_s}} &|f\ (s)|^2 G(\mathbf{x_s}, \mathbf{x_T}, \omega)\bar{G}(\mathbf{x_s}, \mathbf{x_{T+a_x}}, \omega) \\ &\sum_{\mathbf{x_r}} G(\mathbf{x_T}, \mathbf{x_r}, \omega)\bar{G}(\mathbf{x_{T+a_x}}, \mathbf{x_r}, \omega), \end{aligned} \tag{7}$$

where $\mathbf{x_T}$ is an image point within the target area, and $\mathbf{x_{T+a_x}}$ represents points within a *small* region around $\mathbf{x_T}$. For any image point, elements of $\mathbf{H}\left(\mathbf{x_T}, \mathbf{x_{T+a_x}}\right)$ represent a row of a sparse Hessian matrix $\mathbf{H}$ whose non-zero components are defined by $\mathbf{a_x}$. Therefore, $\mathbf{a_x}$ defines the number of off-diagonal elements of the Hessian that are computed — which represents the size of the point spread function (PSF) at each image point (Lecomte and Gelius, 1998; Chavent and Plessix, 1999; Valenciano et al., 2006). $\bar{G}$ is the complex conjugate of Green's function G at frequency ω; f_s is the source function; and $\mathbf{x}_s$ and $\mathbf{x}_r$ are the source and receiver positions, respectively. Note that because of symmetry, only one half of the approximate Hessian is required. In this paper, we follow the phase-encoding approach of

Tang (2009) to efficiently compute the target-oriented Hessian. The spatial regularization operators in equation 4 are non-stationary directional Laplacians (Hale, 2007), whereas the temporal constraint is the difference between the aligned images. Further review of the methodology is given by Ayeni and Biondi (2010, 2011)

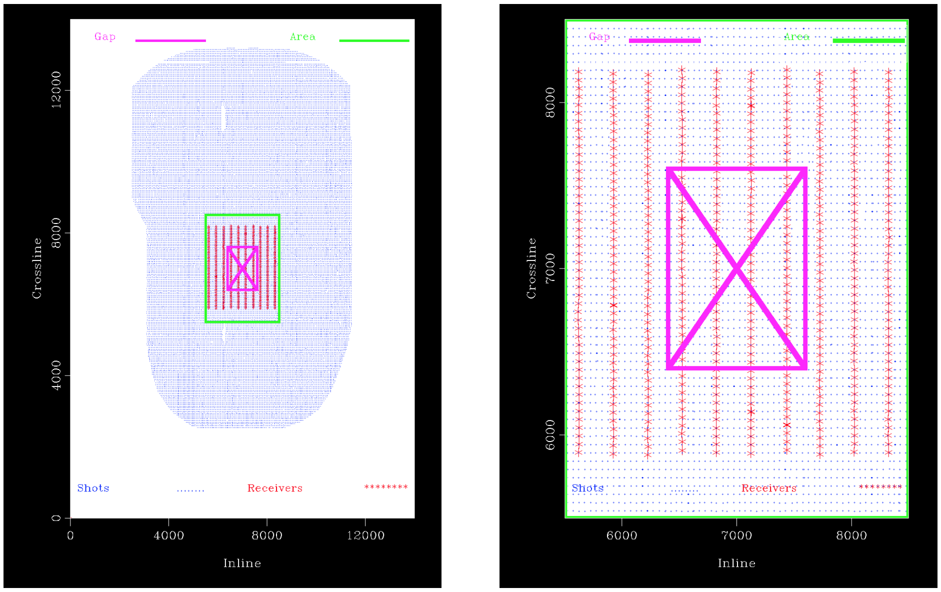
CASE STUDY

We consider a subset of the Life of Field Seismic (LoFS) data sets acquired at Valhall, a giant oil field located in the Norwegian North Sea. There is a wide range of published work on the exploration and development effort in the Valhall field and on different aspects of the LoFS project at Valhall. For example, Munns (1985) discusses Valhall geology in detail; Barkved et al. (2003) discuss the production history and development plans for the field; Barkved (2004) discusses the permanent acquisition array; van Gestel et al. (2008) discuss aspects of the data acquisition, processing, and analysis; and Hatchell et al. (2005) and van Gestel et al. (2011) discuss aspects of the data interpretation and integration with other reservoir data.

In this paper, we consider data from the first (LoFS 1) and the ninth (LoFS 9) surveys acquired in November 2003 and December 2007, respectively. For this study, to avoid imaging challenges caused by a gas cloud located above the crest of the Valhall structure, we choose a subset of the original data covering the Southern flank of the structure. Whereas the original (full) data consists of approximately $50,000$ shots and 2400 receivers, the data subset consists of approximately $33,000$ shots and 470 receivers. Shots are spaced at 50 m in both the inline and crossline directions, while the receivers, located along 10 permanent cables at approximately 70 m depth, are spaced at 50 m in the inline and 300 m in the crossline directions (Figure 1). The maximum absolute source-receiver offset is 5 km. The data have been preprocessed, preserving only the up-going primary compressional wave data. To simulate an obstruction, we create a 1.44 sq. km gap in the monitor data at the center of the 9 sq. km study area (Figure 1(b)). Figure 2 shows the resulting common-midpoint (CMP) fold for the complete (baseline) and incomplete (monitor) geometries. Using reciprocity, shot and receiver locations are swapped, such that receiver gathers are treated as shot records. The data are migrated using 320 frequencies (up to 35 Hz) with a split-step one-way wave-equation shot-profile migration algorithm. All data are migrated with the baseline velocity model (Figure 3) obtained—to a satisfactory degree of accuracy—by full waveform inversion (Sirgue et al., 2010). The target area is a small (700 x 3000 x 3000 m) window around the reservoir, located outside the area most affected by the gas cloud. For both the baseline and monitor geometries, we compute the target-oriented Hessian using 64 frequencies spaced equally within the migration frequency band.

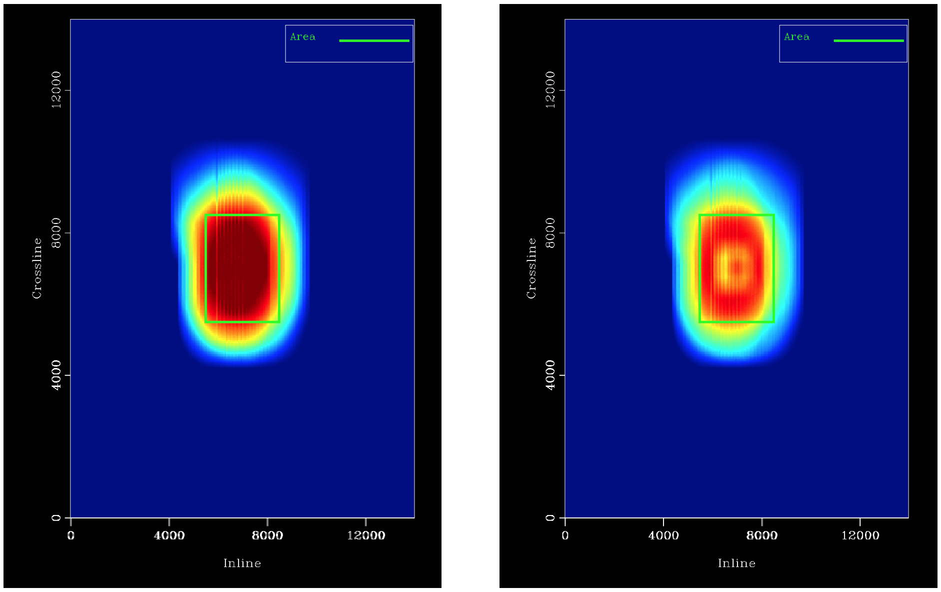
The diagonals of the Hessian matrices (subsurface illumination/fold) for the study area obtained using the complete (baseline) and incomplete (monitor) geometries are shown in Figure 4. Note that in both cases, illumination distribution is highly non-stationary throughout the study area. The ratio between the Hessian diagonals for the two geometries are shown in Figure 5. Note that although the illumination discrepancy is simple at the ocean bottom (Figure 5(a)), this discrepancy becomes highly complex at the reservoir depth (Figure 5(b)).

The migrated baseline and monitor images of the study area are shown in Figure 6. Note that the differences between the images at the reservoir depth are due to a combination



(a) (b)

Figure 1: Acquisition geometry showing locations of all shots and receivers (a) and a zoom showing only the study area (b). Apart from the introduction of a gap, the source-receiver geometry is closely repeated for both data sets. Note that the gap is located at the center of the study area. The coordinate axes in these figures (and in all figures) are distances in meters. [**CR**] gayeni1/. gapped1,gapped2



(a) (b)

Figure 2: Surface (CMP) fold for the baseline (a) and monitor (b). Red indicates high fold, whereas blue indicates low fold. Note that whereas the baseline fold is mostly uniform within the study area, the gapped monitor geometry causes significant non-uniformity of fold. The box indicates the same study area shown in Figure 1(b). [**CR**] gayeni1/. cfold1,cfold2

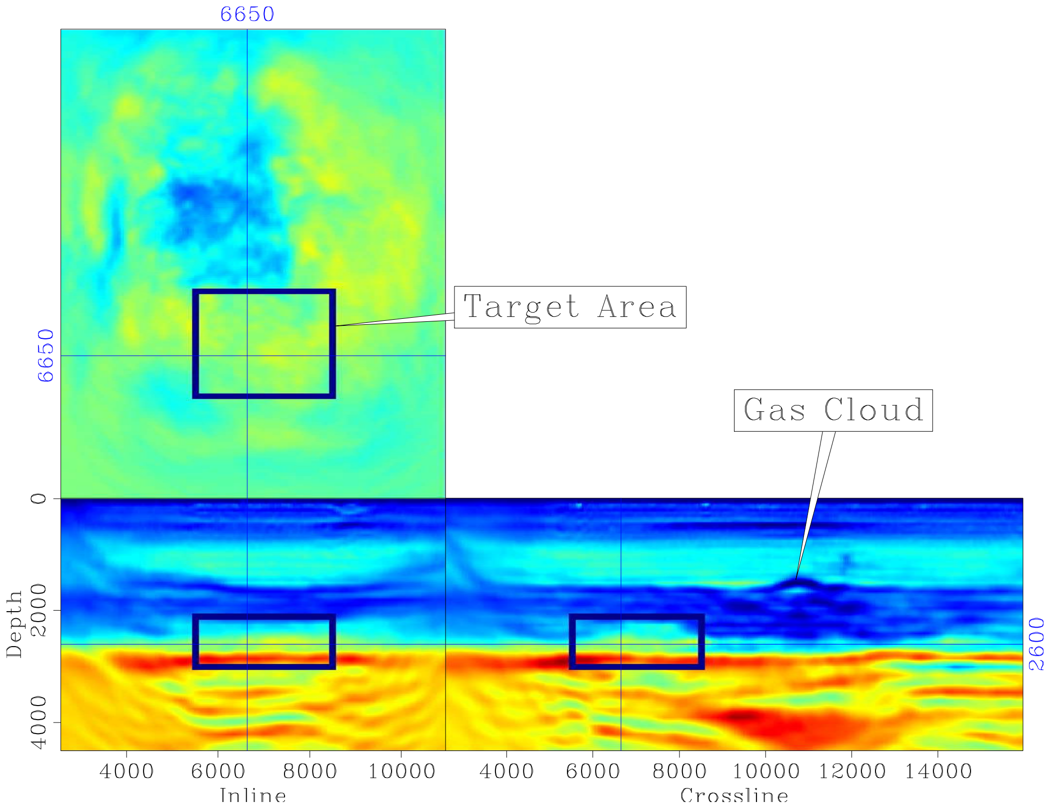


Figure 3: Baseline migration velocity obtained by full waveform inversion (Sirgue et al., 2010). Red indicates high velocity, whereas blue indicates low velocity. This velocity model was used to image all data sets in this study. Note that the target area—indicated by the box—is restricted to a small area of interest around the reservoir. The gas cloud, located outside the study area does not cause significant imaging challenge in the target area. [**NR**] gayeni1/. vel-1l

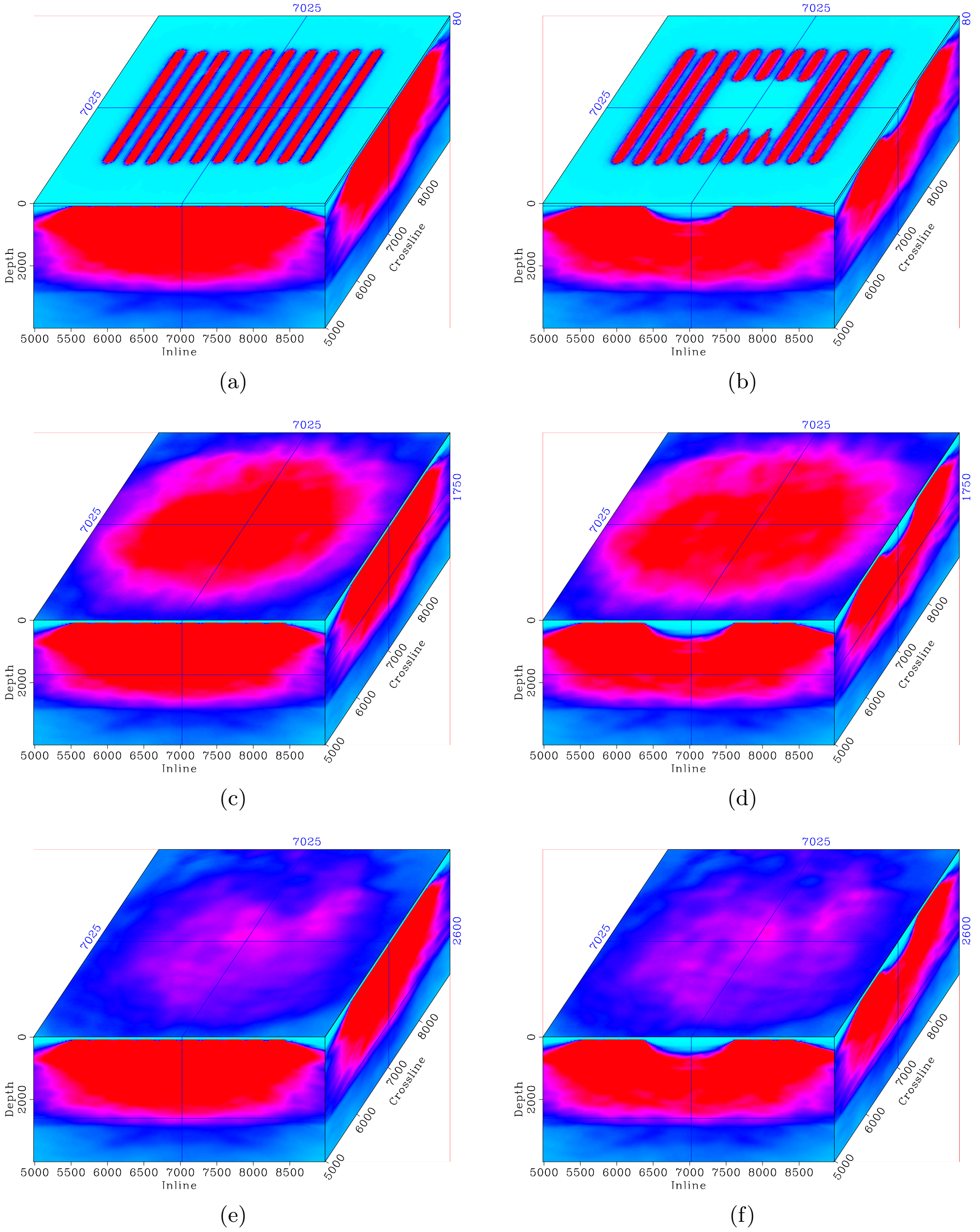
of production-related changes and the gap in the monitor data. In addition, the panels in Figure 6 show the target area for inversion. Because of fluid changes caused by production and injection, and compaction caused by pressure depletion, imaging the monitor data with the baseline velocity causes apparent displacements between the baseline and monitor images. Components of the apparent displacements between the baseline and monitor images (Figure 7) are obtained using a cyclic 1D correlation approach (Ayeni, 2011). Before estimating time-lapse images, and prior to inversion, the baseline and monitor are aligned using these apparent displacements. Time-lapse amplitudes extracted within a 60 m window around the reservoir after migration and inversion are shown in Figure 9.

DISCUSSION

A common problem in many time-lapse seismic monitoring studies is the presence of obstructions that create gaps in the monitor data. Such obstructions, usually caused by production and drilling facilities, generate artifacts that contaminate production-related seismic amplitudes changes, thereby limiting our ability to accurately interpret observed time-lapse amplitudes. The Valhall LoFS project provides data with high repeatability of both source and receiver locations (Figure 1). Therefore, in this case study, the major source of time-lapse amplitude contamination is the synthesized gap in the monitor data. Because CMP fold provides only limited information about the geometry difference (Figure 2), it is insufficient to compensate for subsurface illumination differences. As shown in Figure 4, the Hessian diagonal provides a robust measure of the subsurface illumination for any given geometry. A measure of the subsurface illumination differences can be obtained from the ratio of the Hessian diagonal for the different survey geometries (Figure 5). Although the Hessian diagonal provides information about subsurface illumination and differences, the band-limited wave-propagation effects are provided by the Hessian off-diagonals (not shown). Because the least-squares problem is in the image space, we are able to solve it for a small target around the reservoir (Figure 6). This enables us to try different combinations of inversion parameters efficiently and to focus on improving the results in the region around the reservoir, where the most important production/injection-related changes are expected. Because the Hessian serves as a geometry- and propagation-dependent deconvolution operator, it provides images with improved resolution compared to migration (Figure 8). Because of the gap in the monitor data, there is a large disparity in the distribution of time-lapse amplitudes in the migrated images (Figures 9(a) and 9(b)). Inversion corrects for this disparity, thereby leading to comparable time-lapse amplitude distributions in both the complete and incomplete data examples (Figures 9(c) and 9(d)).

CONCLUSIONS

Because least-squares wave-equation migration/inversion accounts for illumination mismatches—caused by differences in acquisition geometries—and for band-limited wave-propagation effects, it provides images with improved resolution and better definition of seismic amplitude changes. Using subsets of the Valhall LoFS data, we showed that this method can be used to attenuate artifacts caused by obstructions in the acquisition geometries.

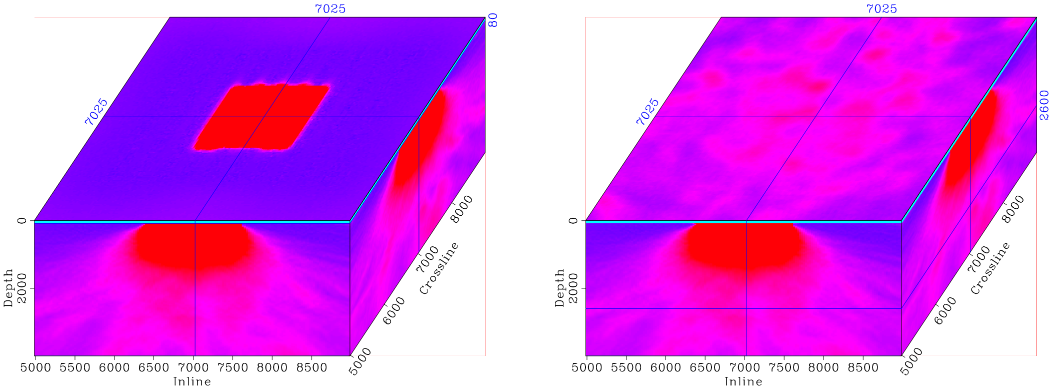


(a) (b)

(c) (d)

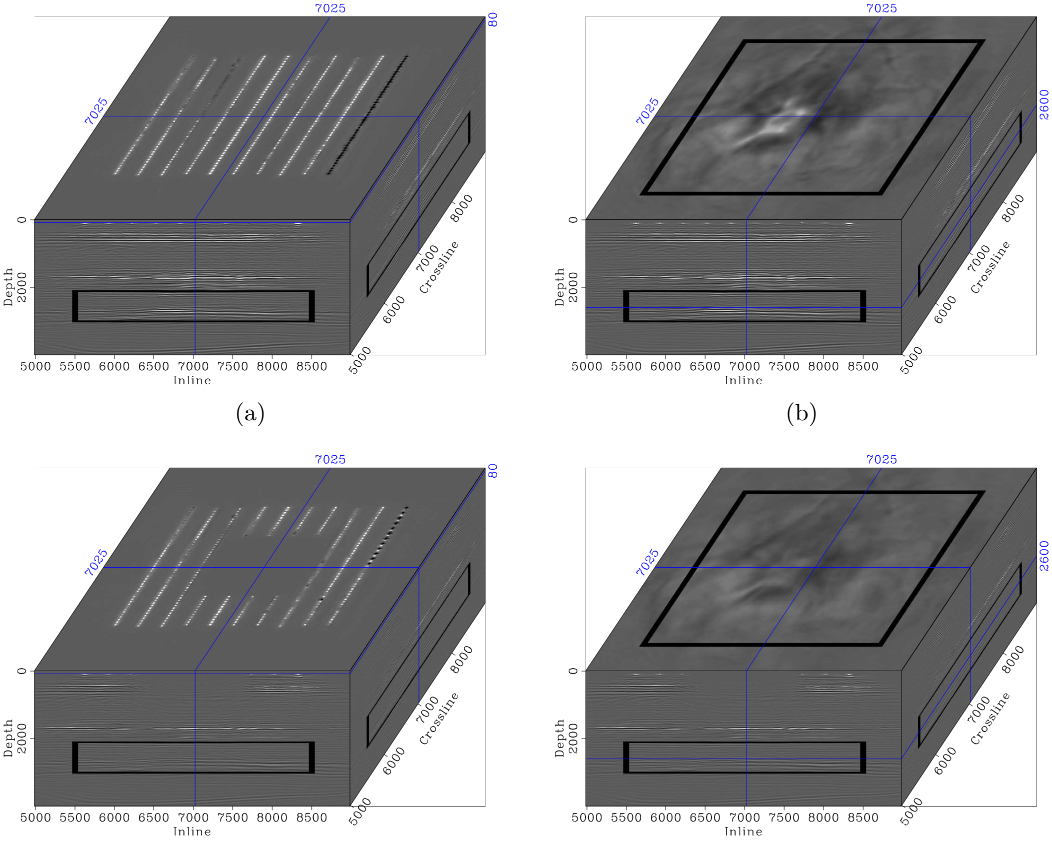
(e) (f)

Figure 4: Hessian diagonal for the complete baseline (left) and incomplete monitor (right). In these (and similar) displays throughout this paper, the top panel is a depth slice and the side panels are the inline and crossline slices. The crosshairs show the position of the slices in the image cube. The depth slices show the illumination at the ocean bottom (a) and (b); above the reservoir (c) and (d); and within the reservoir (e) and (f). Note the locations of the complete receiver lines in (a) and the gap in (b). Red indicates high illumination, whereas cyan indicates low illumination. [**CR**] gayeni1/. ilum-1,ilumg-1,ilum-3,ilumg-3,ilum-4,ilumg-4



(a) (b)

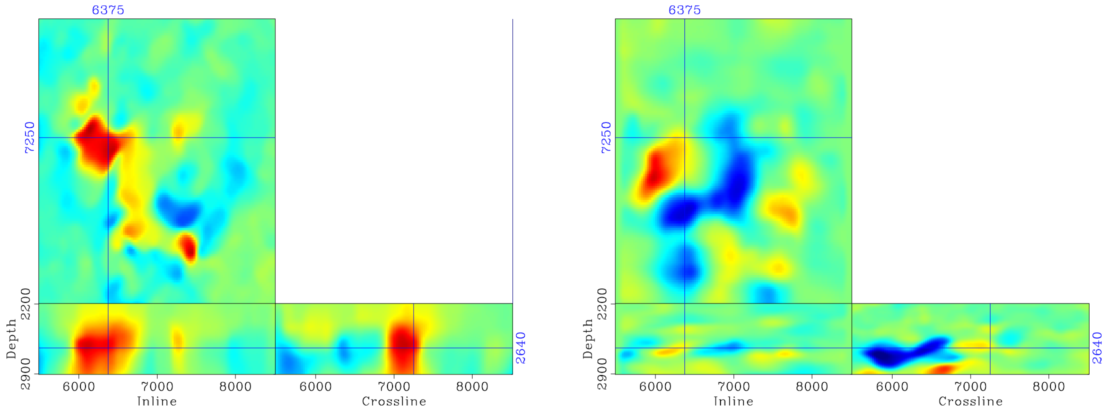
Figure 5: Illumination ratio between the baseline and monitor at the ocean bottom (a) and at the reservoir depth (b). Note that the simple rectangular illumination disparity at the ocean bottom becomes more complex at the reservoir depth. [**CR**] gayeni1/. ilumr-1,ilumr-4



(a) (b)

(c) (d)

Figure 6: Migrated images showing depth slices at the ocean bottom (left) and at the reservoir depth (right). The box indicates the target area in the baseline image (a) & (b), and in the monitor image (c) & (d). Note the location of the gap in the monitor. [**CR**] gayeni1/. mig-1-box,mig-4-box,migg-1-box,migg-4-box



(a) (b)

Figure 7: Vertical (a) and inline (b) components of apparent displacement vectors between the baseline and monitor images within the target area. In both Figures, red indicates positive (downward or rightward) apparent displacements, whereas blue indicates negative (upward/leftward) apparent displacements. Similar results were obtained for the crossline displacement components (not shown). Prior to inversion, the baseline and monitor images are aligned using these apparent displacements. [**CR**] gayeni1/. ts-1,ts-2

ACKNOWLEDGMENTS

We thank BP and Valhall partners for donating the field data used in this study. We thank the Stanford Center for Computational Earth & Environmental Science (CEES) for providing the computer resources used in this study.

REFERENCES

Ajo-Franklin, J. B., J. Urban, and J. M. Harris, 2005, Temporal integration of seismic traveltime tomography: SEG Technical Program Expanded Abstracts, **24**, 2468–2471.

Ayeni, G., 2011, Cyclic 1D matching of time- lapse seismic data sets: A case study of the norne field: SEG Technical Program Expanded Abstracts, **30**, 4234–4238.

Ayeni, G. and B. Biondi, 2010, Target-oriented joint least-squares migration/inversion of time-lapse seismic data sets: Geophysics, **75**, R61–R73.

———, 2011, Wave-equation inversion of time-lapse seismic data sets: SEP Report, **143**.

Barkved, O., P. Heavey, R. Kjelstadli, T. Kleppan, and T. G. Kristiansen, 2003, Valhall Field - still on plateau after 20 years of production: Offshore Europe, **1**.

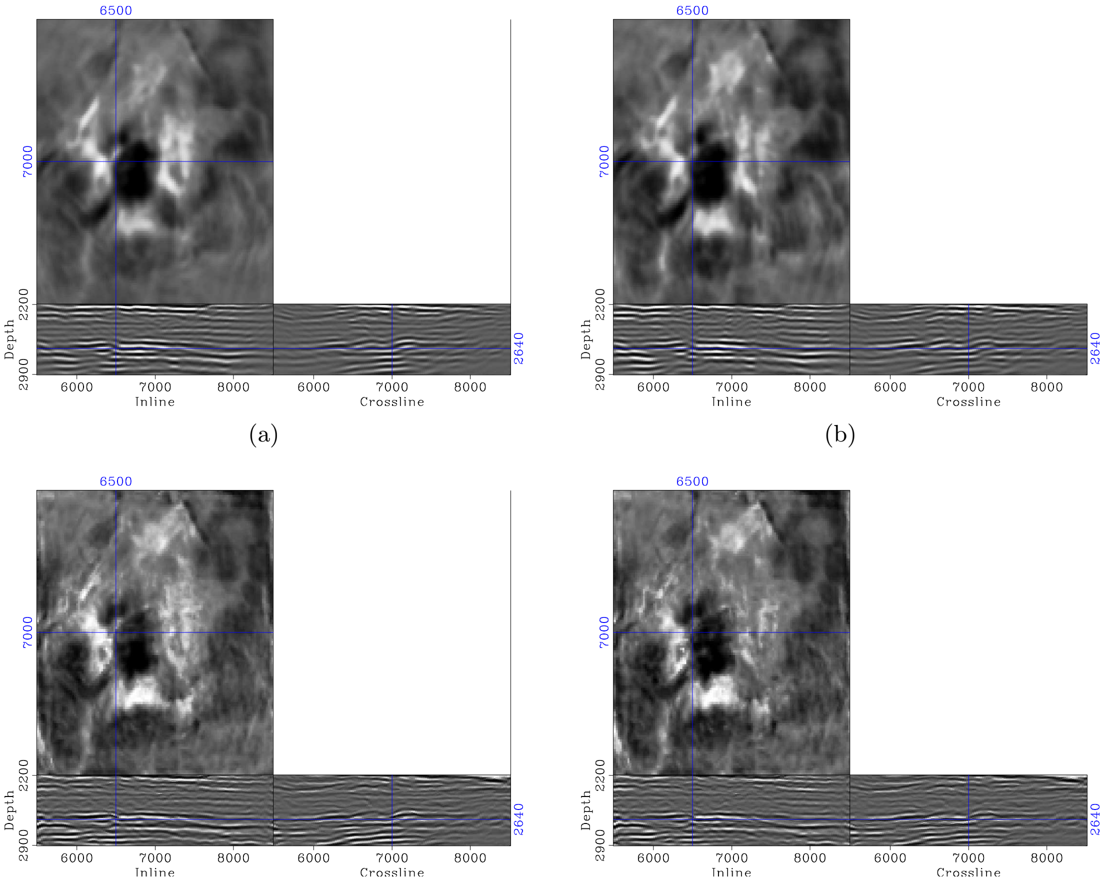
Barkved, O. I., 2004, Continuous seismic monitoring: SEG Technical Program Expanded Abstracts, **23**, 2537–2540.

Batzle, M. and Z. Wang, 1992, Seismic properties of pore fluids: Geophysics, **57**, 1396–1408.

Calvert, R., 2005, Insights and methods for 4D reservoir monitoring and characterization: SEG/EAGE DISC (Distinguished Instructor Lecture Course).

Chavent, G. and R. E. Plessix, 1999, An optimal true-amplitude least-squares prestack depth-migration operator: Geophysics, **64**, 508–515.

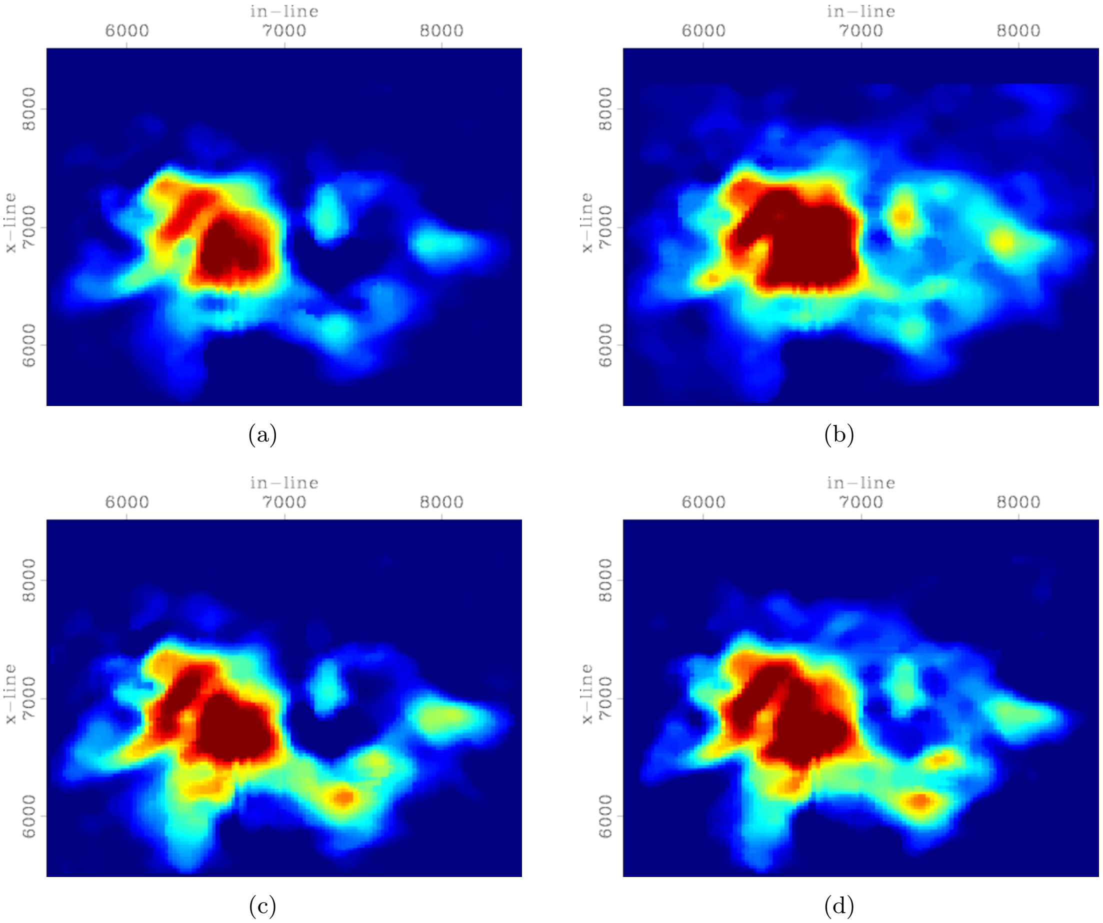
Clapp, M. L., 2005, Imaging under salt: illumination compensation by regularized inversion: PhD thesis, Stanford University.



(a) (b)

(c) (d)

Figure 8: Migrated (a) & (b), and inverted (c) & (d) monitor images for the target area. Panels (a) & (c) are obtained from the complete monitor data, whereas (b) & (d) are obtained from the incomplete (gapped) monitor data. Note that inverted images (c) & (d) show improved resolution over the migrated images (a) & (b). [**CR**] gayeni1/. fine-mod1,gap-mod1,fine-1,gap-1



(a) (b)

(c) (d)

Figure 9: Absolute time-lapse amplitudes in the reservoir obtained from migration (a) & (b), and inversion (c) & (d). Note the discrepancy in the time-lapse amplitude distribution obtained via migration of the complete (a), and incomplete (b) data. Note that this discrepancy has been removed via inversion of the same data sets (c) & (d). [**CR**] gayeni1/. hor-4-d3,hor-4-d2,hor-4-d1,hor-4-d

Ebaid, H., M. Nasser, P. Hatchell, and D. Stanley, 2009, Time-lapse seismic makes a significant business impact at Holstein: SEG, Expanded Abstracts, **28**, 3810–3814.

Hale, D., 2007, A method for estimating apparent displacement vectors from time-lapse seismic images: CWP Report-566.

Hatchell, P., R. Kawar, and A. Savitski, 2005, Integrating 4d seismic, geomechanics and reservoir simulation in the Valhall oil field: 67th EAGE Conference and Exhibition Extended abstract,.

Johnston, D., 2005, Time-lapse 4D technology: Reservoir surveillance: AAPG Search and Discovery.

Kühl, H. and M. D. Sacchi, 2003, Least-squares wave-equation migration for avp/ava inversion: Geophysics, **68**, 262–273.

Lecomte, I. and L.-J. Gelius, 1998, Have a look at the resolution of prestack depth migration for any model, survey and wavefields: SEG Technical Program Expanded Abstracts, **17**, 1112–1115.

Lefeuvre, F., Y. Kerdraon, J. Peliganga, S. Medina, P. Charrier, R. L'Houtellier, and D. Dubucq, 2003, Improved reservoir understanding through rapid and effective 4D: Girassol field, Angola, West Africa: SEG Technical Program Expanded Abstracts, **22**, 1334–1337.

Lumley, D. E., 1995, Seismic time-lapse monitoring of subsurface fluid flow: PhD thesis, Stanford University, http://sepwww.stanford.edu/public/docs/sep91/.

Mavko, G., T. Mukerji, and J. Dvorkin, 2003, The rock physics handbook: Tools for seismic analysis of porous media: Cambridge University Press.

Munns, J., 1985, The Valhall field: a geological overview: Marine and Petroleum Geology, **2**, 23 – 43.

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

Rickett, J. E. and D. E. Lumley, 2001, Cross-equalization data processing for time-lapse seismic reservoir monitoring: A case study from the Gulf of Mexico: Geophysics, **66**, 1015–1025.

Sirgue, L., O. Barkved, J. Dellinger, J. Etgen, U. Albertin, and J. Kommedal, 2010, Full waveform inversion: the next leap forward in imaging at Valhall: First Break, **28**.

Tang, Y., 2009, Target-oriented wave-equation least-squares migration/inversion with phase-encoded hessian: Geophysics, **74**, WCA95–WCA107.

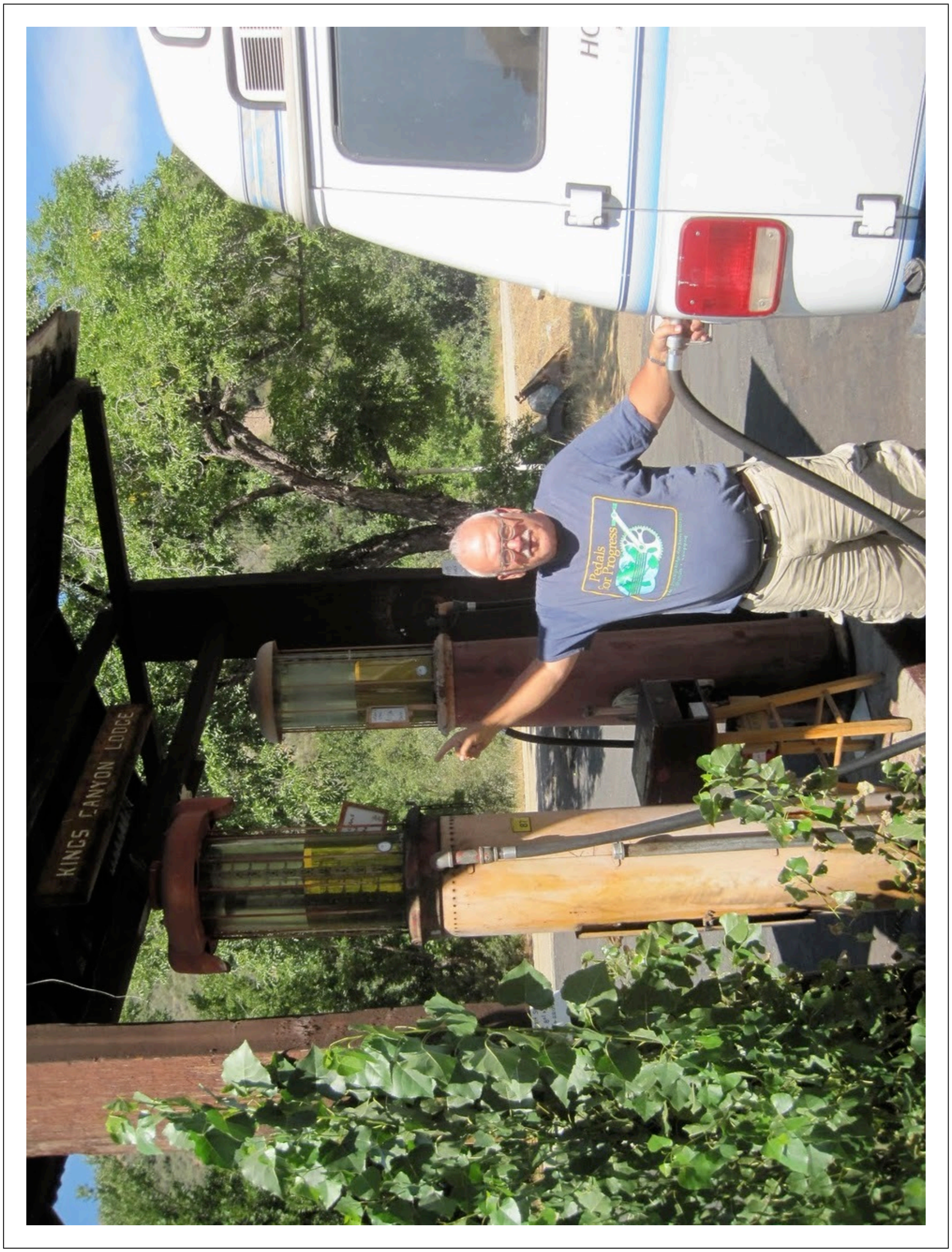
Valenciano, A. A., B. Biondi, and A. Guitton, 2006, Target-oriented wave-equation inversion: Geophysics, **71**, A35–A38.

van Gestel, J.-P., K. D. Best, O. I. Barkved, and J. H. Kommedal, 2011, Integration of the life of field seismic data with the reservoir model at the Valhall field: Geophysical Prospecting, **59**, 673–681.

van Gestel, J.-P., J. H. Kommedal, O. I. Barkved, I. Mundal, R. Bakke, and K. D. Best, 2008, Continuous seismic surveillance of Valhall field: The Leading Edge, **27**, 1616–1621.

Whitcombe, D. N., J. M. Marsh, P. J. Clifford, M. Dyce, C. J. S. McKenzie, S. Campbell, A. J. Hill, R. S. Parr, C. Pearse, T. A. Ricketts, C. P. Slater, and O. L. Barkved, 2004, The systematic application of 4D in BP's North-West Europe operations — 5 years on: SEG Technical Program Expanded Abstracts, **23**, 2251–2254.

Zou, Y., L. R. Bentley, L. R. Lines, and D. Coombe, 2006, Integration of seismic methods with reservoir simulation, Pikes Peak heavy-oil field, Saskatchewan: The Leading Edge, **25**, 764–781.


KINGS CANYON LODGE
Pedals for Progress

Stanford Exploration Project, SEP145, September 13, 2011

Elastic Born modeling in an ocean-bottom node acquisition scenario

Ohad Barak

ABSTRACT

PZ summation is a common method for separating the upgoing wavefield from the downgoing wavefield in data acquired by four-component ocean-bottom node surveys. It assumes that the vertical geophone component records mostly pressure waves. If this assumption is not satisfied, non-pressure wave energy (such as shear waves) will be introduced as pressure waves into the receiver wavefield, which may generate artifacts in the migration image. I formulate an elastic Born modeling and migration method for ocean-bottom node acquired data. I then use a synthetic example to demonstrate the effect of the introduction of non-pressure wave energy into the receiver data on the resulting image.

INTRODUCTION

In four-component ocean-bottom node (OBN) acquisition, the pressure wave is recorded by a hydrophone suspended in the water layer just above the sea floor. Additionally, three-component geophones are attached to the sea floor and record the vertical and two perpendicular horizontal particle velocities. The source is an airgun fired at the water surface. The upgoing wave can be distinguished from the downgoing wave by comparing the vertical geophone data to the hydrophone data. Whereas in the hydrophone data both the downgoing and the upgoing wave will have a positive polarity (assuming positive reflection coefficients in the subsurface), the value of the geophone data depends on the wave's vertical propagation direction. The downgoing wave will register on the vertical geophone with a negative polarity, while the upgoing wave will have a positive polarity.

This characteristic has led to the PZ summation methodology (Barr and Sanders, 1989). With a proper scaling factor, the sum of the hydrophone and the vertical geophone should result in the upgoing waves only, while their difference should yield the downgoing wave only. This differentiation between the energy propagation direction at the sea floor has several applications. One of these applications is the ability to create separate images of the upgoing waves and of the downgoing waves that are reflected off the water surface, also known as the "mirror-image" (Ronen et al., 2005; Wong et al., 2009).

The PZ summation methodology, however, assumes that the energy recorded in the hydrophone and in the vertical geophone is mostly pressure wave energy. For the hydrophone, since it is within the water column, this assumption holds. However, if significant amounts of other wave modes are recorded by the geophone, then the PZ summation will not handle them properly, and will in effect introduce them into either the upgoing or the downgoing field. The sketch in Figure 1 displays this issue for the case where PZ summation is used to extract the upgoing data. Later processing steps (such as mirror-imaging), which rely on the assumption that the separated upgoing or downgoing data contains only acoustic arrivals, may have their results affected in turn.

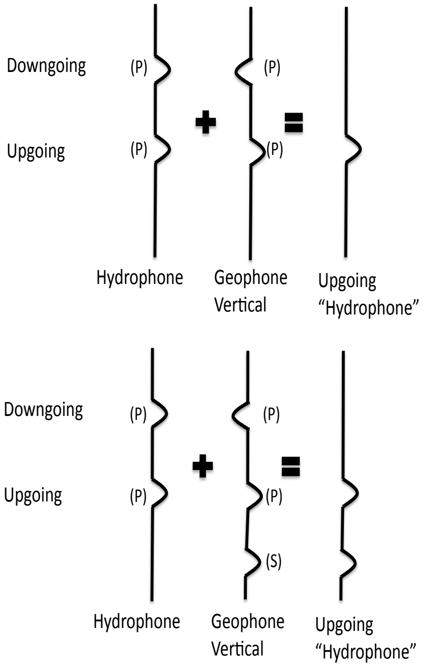


Figure 1: Simplified sketch of PZ summation. Top: For acoustic data, the summation of the hydrophone data and the vertical geophone data will result in the downgoing energy being eliminated, leaving only upgoing pressure data. Bottom: If the data contains shear waves, they will be recorded only by the geophone, and the PZ summation will mistakingly insert them into the result as upgoing pressure data. [**NR**] ohad1/. up-down-sketch

In this paper I show the effect of imperfect PZ summation, as a result of significant shear wave energy existing in the vertical geophone, on the imaging of the upgoing wavefield.

In order to create single-scattered OBN synthetic data and run RTM with such data as input, I formulated and coded an elastic Born modeler. The special case of OBN acquisition necessitated a special manipulation of the smooth and the perturbed velocity models required to carry out Born modeling. The geophone receiver data was synthesized by recording the particle velocities of the single-scattered wavefield at the sea bed, and the hydrophone data was synthesized by recording the average of the normal stresses just above the sea bed, in the water-column.

The elastic propagation code is two dimensional, and all following examples are 2D as well. The gridding method I used for the elastic propagation was the Virieux staggered grid (Virieux, 1986), in which some wavefield values are located at grid points, and some are located at half-grid points, both spatially and temporally. The actual values which are propagated are the normal and transverse stresses $(\sigma_{xx}, \sigma_{zz}, \sigma_{xz})$, and the vertical and horizontal particle velocities (v_x, v_z). The P and S wavefields are extracted from the particle velocity fields by applying either the divergence or the curl operator to them, respectively.

For imaging of the P and S-wave modes in the elastic source and receiver wavefields, I follow the vector potential imaging condition discussed in Yan and Sava (2008). As they mention, the imaging results for full elastic propagation suffer from spurious modes being created when particle velocity data are injected as a boundary condition into the receiver wavefield during elastic RTM. In other words, injection of recorded P-wave particle velocities in an elastic medium will invariably create a P-wave and an S-wave mode. These injected spurious modes will, at the imaging stage, give rise to artifacts. This problem is similar to the problem of multiple generation in the receiver wavefield when a non-smooth model is used.

There are more advanced methods for achieving both up/down separation in conjunction with P/S separation, in the data domain. Dankbaar (1985), Wapenaar et al. (1990), Amundsen (1993) and Schalkwijk et al. (2003) have all shown methods which can separate pressure and shear waves in the data, as well as separating upgoing from downgoing. However these methods require good knowledge of medium parameters at the sea bed where the geophones are located. Furthermore, it is not correct to assume that the data are composed of pressure and shear body waves only. In OBN acquisition (as in land acquisition) surface waves can contaminate the data.

PZ SUMMATION

PZ summation involves summing the pressure data recorded by the hydrophone data with the vertical particle velocity data recorded by the geophone, with some scaling factor:

$$\begin{aligned} U(z_r) &= \frac{1}{2}\left[P(z_r) - \beta V_z(z_r)\right], \\ D(z_r) &= \frac{1}{2}\left[P(z_r) + \beta V_z(z_r)\right], \end{aligned} \tag{1}$$

where P is the pressure data, V_z is the vertical particle velocity, U is the upgoing data, D is the downgoing data and z_r is the receiver depth. β is a scaling factor, which can be defined in several ways. Amundsen (1993) does the separation in the $f-k$ domain, and uses $\beta = \frac{\rho\omega}{k_z}$, where $k_z = \sqrt{\frac{\omega^2}{v^2} - k_x^2 - k_y^2}$, and ρ is the density. Alternately in the $t-x$ space, the scaling can be determined by the ratio of the direct arrival's amplitude on the hydrophone and vertical geophone components at various offsets.

As mentioned above, this method assumes that all energy is pressure wave energy, and therefore everything recorded by the hydrophone has its counterpart in the geophone data, with either positive polarity (upgoing) or negative polarity (downgoing). Shear waves, on the other hand, can only propagate upwards, since they are generated by mode conversions in the subsurface. They can have a very different polarity upon reaching the sea bed, which can change differently with offset in comparison to the P-wave polarity. Running PZ summation on data that contains significant shear wave energy (or indeed - significant amounts of anything that is not pure P body waves) introduces events into the separated data fields, which will be construed by later processing steps as either upgoing or downgoing P-wave energy.

ELASTIC BORN MODELING

The elastic isotropic wave equation in index notation reads:

$$\partial_i \sigma_{ii} + \partial_j \sigma_{ij} + f_i(\mathbf{x}, t) = \rho \partial_t v_i, \tag{2}$$

where σ_{ii} are the normal stresses, σ_{ij} are the transverse stresses, f_i is the source function in direction i, $\mathbf{x}$ is the spatial source location operating at time t, ρ is density and v_i is the particle velocity in direction i. The stresses are propagated using the stress-displacement relation:

$$\begin{aligned} \sigma_{ii} &= (\lambda + 2\mu)\partial_i v_i + \lambda \partial_j v_j, \\ \sigma_{ij} &= \mu \left(\partial_j v_i + \partial_i v_j \right), \end{aligned} \tag{3}$$

where λ and μ are the Lame elastic constants.

In the staggered time grid methodology for elastic propagation (Virieux, 1986) the stresses and particle velocities are always half a time step apart. Therefore equation 2 and equation 3 are solved in alternation during the propagation.

For elastic Born modeling, these equations must be linearized. Beylkin and Burridge (1990) show a full derivation of linearized scattering for an elastic solid. Using $r = \frac{1}{\rho}$ to denote the specific volume, we have three models for the elastic isotropic case:

$$r = r^0 + \Delta r, \quad \lambda = \lambda^0 + \Delta\lambda, \quad \mu = \mu^0 + \Delta\mu, \tag{4}$$

where λ^0, μ^0 and r^0 are the smooth models, and $\Delta\lambda$, $\Delta\mu$ and Δr are the perturbed models.

We also have the incident and scattered stress and particle velocity fields:

$$v_i^0, \quad \Delta v_i, \quad \sigma_{ij}^0, \quad \Delta\sigma_{ij} \tag{5}$$

The incident stress and particle velocity fields need to propagated with the smooth models so that they do not generate reflections. At each time step, the perturbed models must be multiplied by these incident fields, and then injected as an additive source function into the scattered fields. As a result of the staggered time grid, the injection must be done alternately into the stress and particle velocity fields. The scattered fields themselves must also be propagated with the smooth models, so that they do not generate any additional scattering.

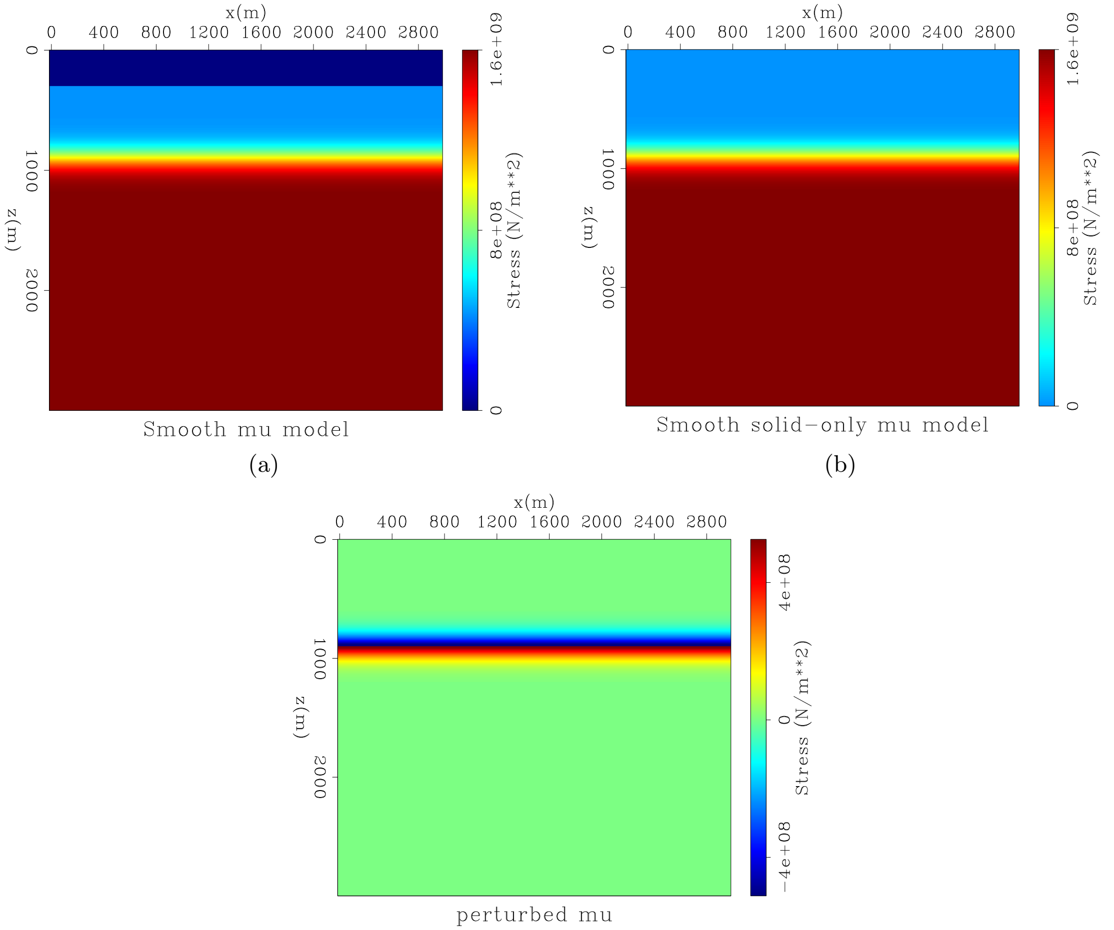
Low shear velocities at the water-solid boundary

The special case of OBN acquisition presents a unique problem for elastic Born modeling. Shear velocity in the water column is zero. In the shallow sediment, while not being very high, the shear velocity is necessarily not zero. The smoothing of the shear velocity model near the water-solid interface means that very small shear velocity values exist in the model. These in turn give rise to extremely short wavelengths, which require very fine gridding in order to spatially sample them properly and avoid dispersion effects. This issue is doubly important for OBN, since the water-solid interface is where the receivers are located, meaning that any dispersion in the modeling will have a direct effect on the synthesized data, and on the reverse propagated wavefields. I've found one possible temporary solution to this problem: use two separate smooth models, one for the incident wavefield and one for the scattered wavefield. Using Θ to denote the group of medium paramers, we have:

Incident smooth models: $\Theta_I^0 = \left[r_I^0, \quad \lambda_I^0, \quad \mu_I^0 \right]$.
Scattered smooth models: $\Theta_S^0 = \left[r_S^0, \quad \lambda_S^0, \quad \mu_S^0 \right]$.

Perturbed models: $\Delta\Theta_S = [\Delta r_S, \quad \Delta\lambda_S, \quad \Delta\mu_S]$.

Figures 2(a)-2(c) show these three model versions for the μ parameter.



(a) (b)

(c)

Figure 2: Top left: Smooth model for incident wavefield. Note the sharp water-solid boundary. Top right: Smooth model for scattered wavefield, where the water has been replaced by the topmost layer. Bottom: The perturbed model, which is the difference between the true model without the water layer and the smooth model without the water layer. Note the lack of a perturbation at the sea bed. [**ER**] ohad1/. mu2d-smooth,mu2d-smooth-lith,mu2d-del

For the incident wavefield, the model is smoothed everywhere except for the water-solid interface (Figure 2(a)). For the scattered wavefield, the topmost sediment's model replaces the water column (Figure 2(b)). The perturbed models are the difference between the "true" scattered model (water replaced by solid) and the smooth scattered model. Therefore, the perturbed models do not include the water-solid boundary (Figure 2(c)), and as a result no reflection is generated in the scattered wavefield at the sea bed. There is, however, a reflection generated in the incident wavefield at the water bottom, which admittedly is exactly the way Born modeling is not supposed to work. The water-solid substitution also causes wrong kinematics of the scattered wavefield in the water column. Neither of these issues affect the results since all energy is absorbed at the model boundaries, eliminating any free-surface multiples. A different solution will have to be found at a later stage if I wish to incorporate free-surface multiples into the processing.

WAVE MODE SEPARATION AND IMAGING

The Helmholtz amplitude separation is based on the assumption that any isotropic vector field can be described as a combination of a scalar and vector potential fields:

$$\mathbf{u} = \nabla\Phi + \nabla \times \mathbf{\Psi}, \tag{6}$$

Where Φ is the scalar potential field and $\mathbf{\Psi}$ is the vector potential. $\mathbf{u}$ is the elastic displacement vector wavefield. The scalar potential generates pressure waves, and the vector potential generates shear waves. Therefore, the Helmholtz method of separating the P-wave amplitude from the S-wave amplitude is to apply a divergence operator and a curl operator to the displacement field:

$$P = \nabla \cdot \mathbf{u} = \nabla^2\Phi; \tag{7}$$

$$\mathbf{S} = \nabla \times \mathbf{u} = -\nabla^2\Psi. \tag{8}$$

Equations 7 and 8 work only for an isotropic medium. Dellinger and Etgen (1990) extend these operators for an anisotropic medium. Yan and Sava (2008) use these separated P and S-wave modes to formulate an imaging condition for vector potentials in an isotropic medium:

$$I_{ij}(\mathbf{x}) = \int_t \alpha_{si}\left(\mathbf{x}, t\right) \alpha_{ri}\left(\mathbf{x}, t\right) dt, \tag{9}$$

where the indices i, j denote the wave mode (P or S) of the wavefield α, and s, r denote the source and receiver fields. Therefore $I_{PP}(\mathbf{x})$ represents the cross-correlation of the source P field with the receiver P field, while $I_{PS}(\mathbf{x})$ represents the cross-correlation of the source P field with the receiver S field. In addition to the conventional PP image, the PS image can supply more information regarding medium parameters. This does, however, depend on the quality of the shear wave data, and on the accuracy of the acoustic velocity, shear velocity and density models.

MODELING AND MIGRATION RESULTS

Elastic Born forward modeling

In the following examples, I used a 3 layer 1D model of acoustic velocity, shear velocity and density. The top layer was water, and there was a full spread of receivers located at the water bottom. Figure 3 represents the models used for the incident wavefield (sharp water bottom interface). In the models used to propagate the scattered wavefield, the layer marked "sediment 1" is extended all the way to the surface. The model parameters were:

$\rho_1 = 1025\frac{kg}{m^3}$ $V_{p1} = 1500\frac{m}{s}$ $V_{s1} = 0\frac{m}{s}$

$\rho_2 = 1700\frac{kg}{m^3}$ $V_{p2} = 1700\frac{m}{s}$ $V_{s2} = 500\frac{m}{s}$

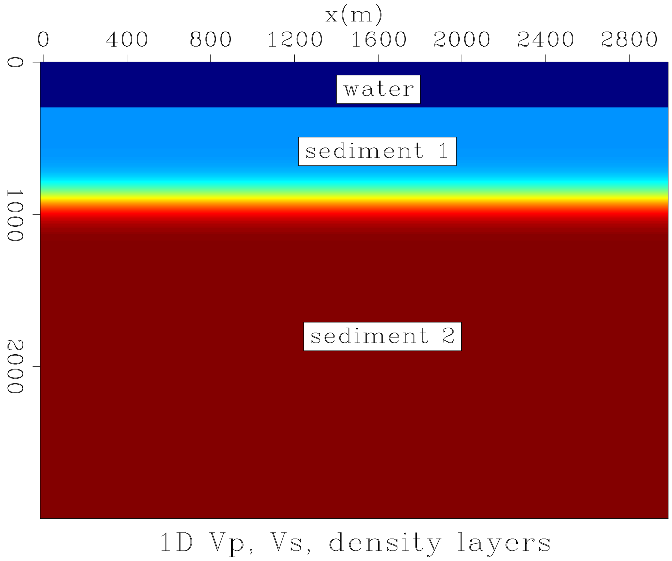


Figure 3: The smooth ρ, V_p and V_s model representation for the incident wavefield. [**ER**] ohad1/. smoothmods

$\rho_3 = 1600\frac{kg}{m^3}$ $V_{p3} = 2200\frac{m}{s}$ $V_{s3} = 1000\frac{m}{s}$

Figures 4(a)-4(l) show the incident and scattered P and S wavefields, which are a result of the application of equations 7 and 8 to the Born-modeled incident and scattered particle velocity fields. The rows are arranged by time snapshots, so at each row we see each field at the same time. The column order from left to right is incident P, scattered P, incident S, scattered S. In Figure 4(a) we can see the P reflection at the water bottom, resulting from the sharp boundary there. This reflection is absorbed by the attenuating boundaries. In Figure 4(b) and 4(f), the P reflection from the bottom reflector is visible. Figure 4(c) is the incident S-wave generated by mode conversion at the water-solid interface. Figures 4(d) and 4(h) show the scattered S-wave generated by a mode conversion at the bottom reflector. Figures 4(e) and 4(i) are the propagating incident P wavefield. In Figure 4(g), though it is the incident S field, we can still see a transmitted mode conversion from the incident P wavefield at the bottom reflector. In Figure 4(k), the S-wave has hit the bottom reflector, which is the reason that we see both a P and an S reflection in the scattered field in Figures 4(j) and 4(l).

The simulation of OBN hydrophone recording is done by saving the scattered P wavefield one model cell above the sea bottom. The horizontal and vertical geophone recording is done by saving the vertical and horizontal particle velocity fields, within the solid layer at the sea bottom. The receivers are fully spread on the sea bottom. The source is an explosive source at the sea surface.

Figure 5 shows the recorded data components, generated by the elastic Born modeling. There is no direct arrival, since the water-solid interface is not used to generate scattering. Note that the shear wave recording is virtual. There is no recevier that records only shear waves. However, using equation 8 the shear wave value can be extracted from the particle-velocity fields at the sea bed. This information is useful for analyzing which of the reflections are P and which are S. The reflection types are pointed out in the figure. The label “PP” signifies a P-wave generated at the source position, which was transmitted at the water bottom as a P-wave, and was reflected as a P-wave at the bottom reflector within the

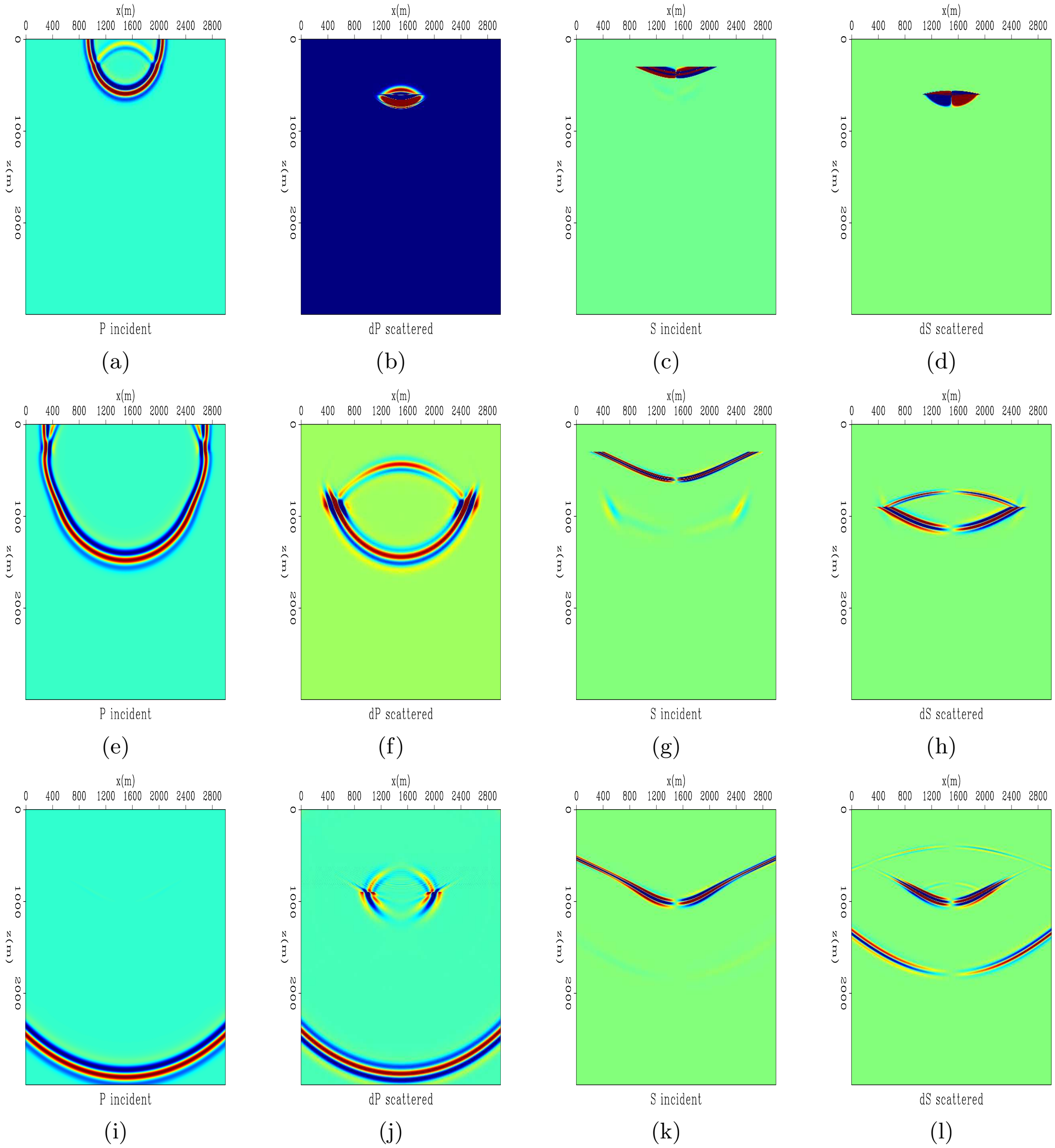


Figure 4: P and S incident and scattered wavefield snapshots at propagation times: t1=0.46s (top row), t2=0.9s (center row), t3=1.55s (bottom row). [**ER**]

ohad1/. 0Psnap1,0dPsnap1,0Ssnap1,0dSsnap1,0Psnap2,0dPsnap2,0Ssnap2,0dSsnap2,0Psnap3,0dPsnap3,0Ssn sediment (Figure 3). The label "PS" signifies a similar path, except that at the bottom reflector a mode conversion occures, resulting in an upgoing S-wave. The label "SP" signifies a P-wave generated at the source position, which was converted at the sea bottom to a transmitted S-wave, and was converted again upon reflection at the bottom reflector to an upgoing P-wave.

Observing the hydrophone data and the virtual shear wave recording, we can see that the vertical and radial geophone components record both P and S-waves. Furthermore, we can see that the arrival at $t = 1.8s$ contains two converted modes: PS and SP. They arrive at the same time, since the model is 1D. As a result of the single-scattering of the Born modeler, all data in Figure 5 are upgoing data, and therefore represent the result we expect to have from a perfect separation of the upgoing from the downgoing wavefields.

Figure 6 is the recording of data generated by acoustic Born modeling. In effect, I used the same code as for the elastic modeling, setting $V_s = 0$. This is the reason for the empty data in the virtual shear wave recording. As expected, there is only one PP arrival. Again - no direct wave is visible since the water-solid interface is not used to generate scattering. Like the previous elastic data example, this upgoing acoustic recording is what we expect from a perfect up/down separation.

PZ summation and imaging of pure and converted wave modes

PZ summation operates on the hydrophone and the vertical geophone component. Looking at figure 5, we can see that were we to use PZ summation, the PS and the SS arrivals would be introduced into the summation result. This effect is shown in the left panel of Figure 7. I scaled the geophone data to match the first reflection event of the hydrophone data (which is a PP reflection), and then took the average of the vertical geophone and hydrophone, which is equivalent to applying equation 1. In a standard processing flow, this result of PZ summation would be treated as acoustic data, and migrated with an acoustic velocity. The left panel of Figure 8 is the result of acoustic reverse time propagation of the PZ-summed hydrophone data, followed by a cross-correlation with an acoustically propagated source wavefield, and then a stacking over sources to improve the signal-to-noise ratio. The source and receiver wavefields are propagated with a smooth velocity model so that no receiver-side multiples are generated. Two artifacts are visible below the true reflector position. These artifacts come from the cross-correlation of the SS and the PS reflection events, which were brought into the acoustic data by PZ summation, with the acoustic source wavefield.

In a perfectly acoustic medium, the result of PZ summation would be just the PP reflection event, as can be seen on the right panel of Figure 7. Executing acoustic RTM with a smooth acoustic velocity model and the right panel of Figure 7 as input, we get the artifact-free image on the right panel of Figure 8.

Full elastic migration results

Conceivably, since the data in Figure 5 is strictly upgoing data, and is a result of elastic Born modeling, we would expect that running elastic RTM with this data as the receiver wavefield input should produce an accurate image of the subsurface. The actual imaging

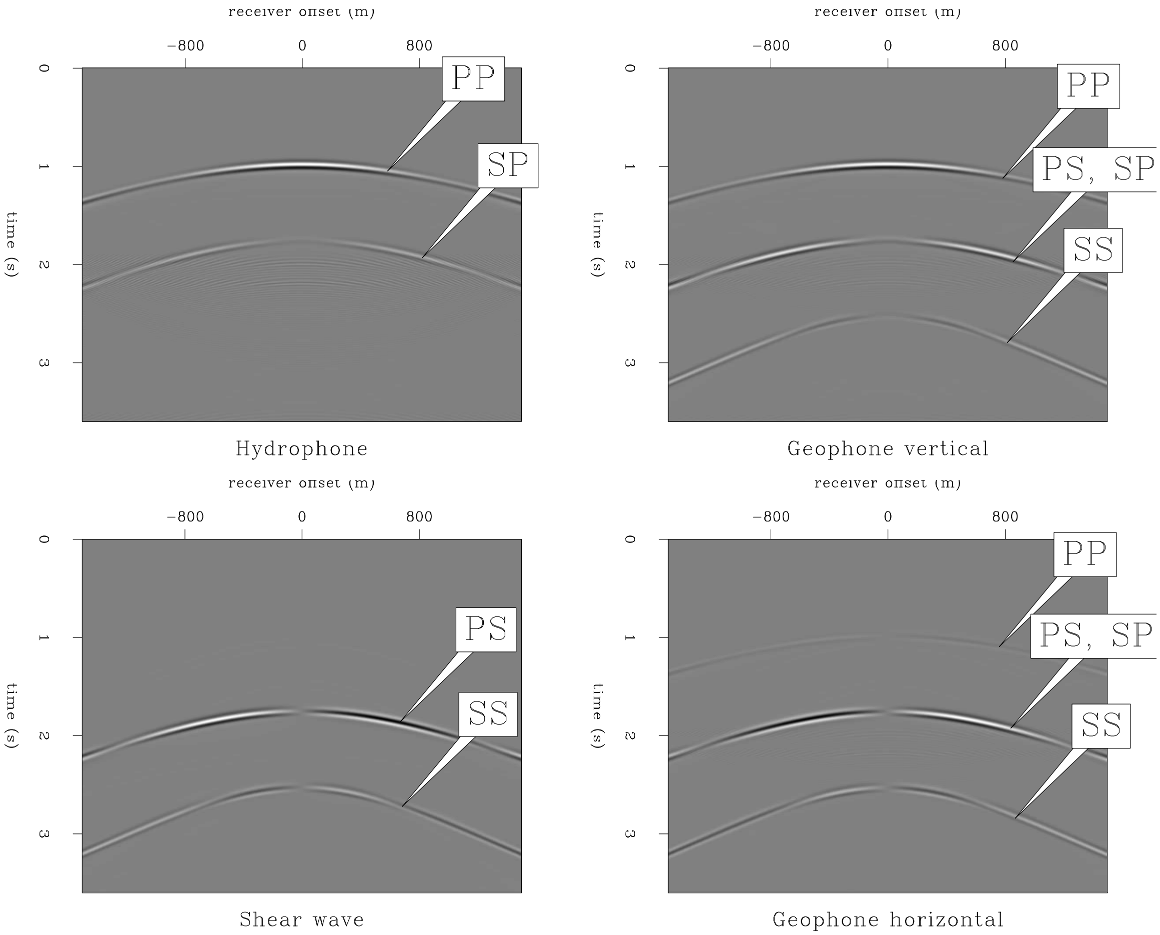


Figure 5: Synthetic OBN data generated by elastic Born modeling. Top left: Hydrophone containing only P-wave data. Top right: Vertical geophone containing P and S-wave data. Bottom left: Shear wave virtual recording. Bottom right: Horizontal geophone containing P and S-wave data. Note the lack of a direct arrival, as a result of the water-solid interface not being used to generate scattering. [**ER**] ohad1/. 0-recfigs-mute

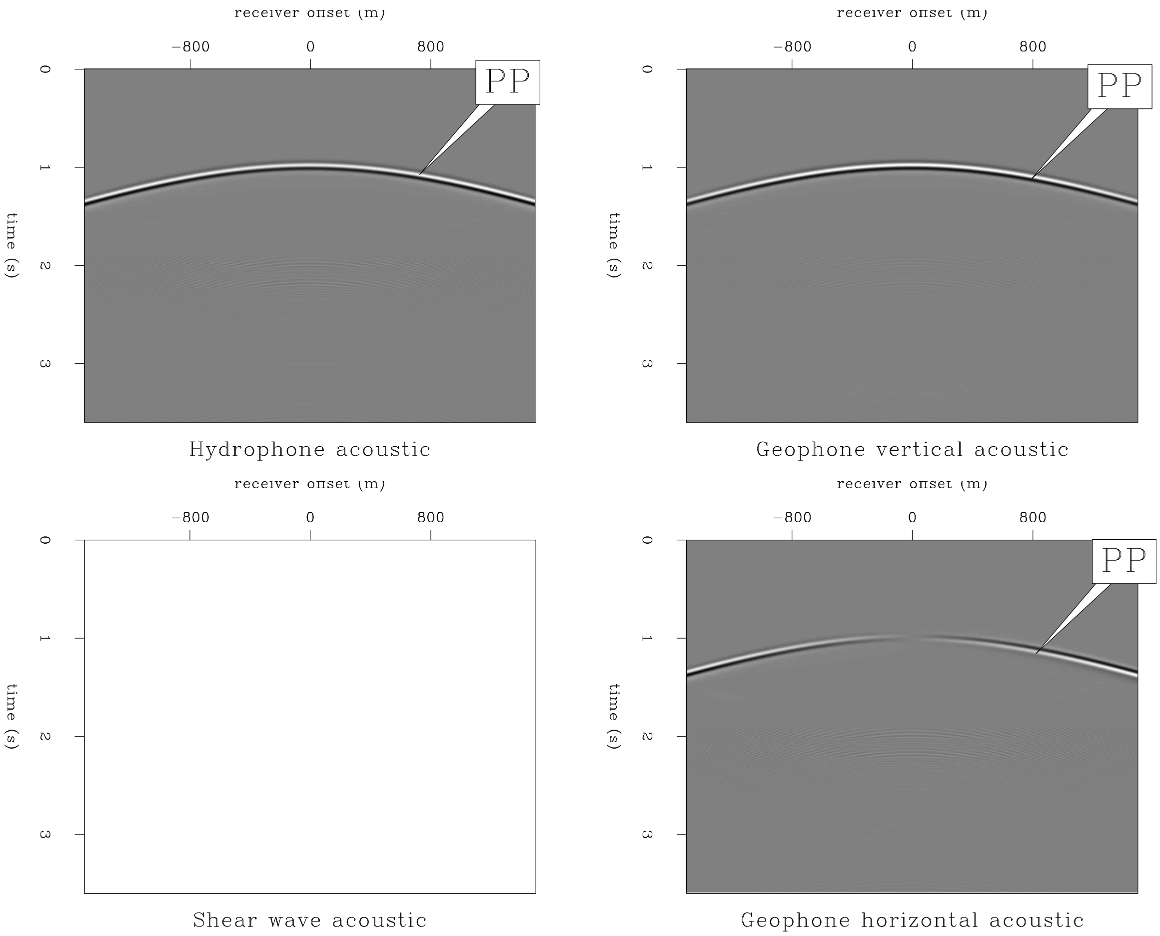


Figure 6: Synthetic OBN data generated by acoustic Born modeling. Top left: Hydrophone containing only P-wave data. Top right: Vertical geophone containing only P-wave data. Bottom left: Shear wave virtual recording. Bottom right: Horizontal geophone containing only P data. [**ER**] ohad1/. 0-recfigs-mute-acous

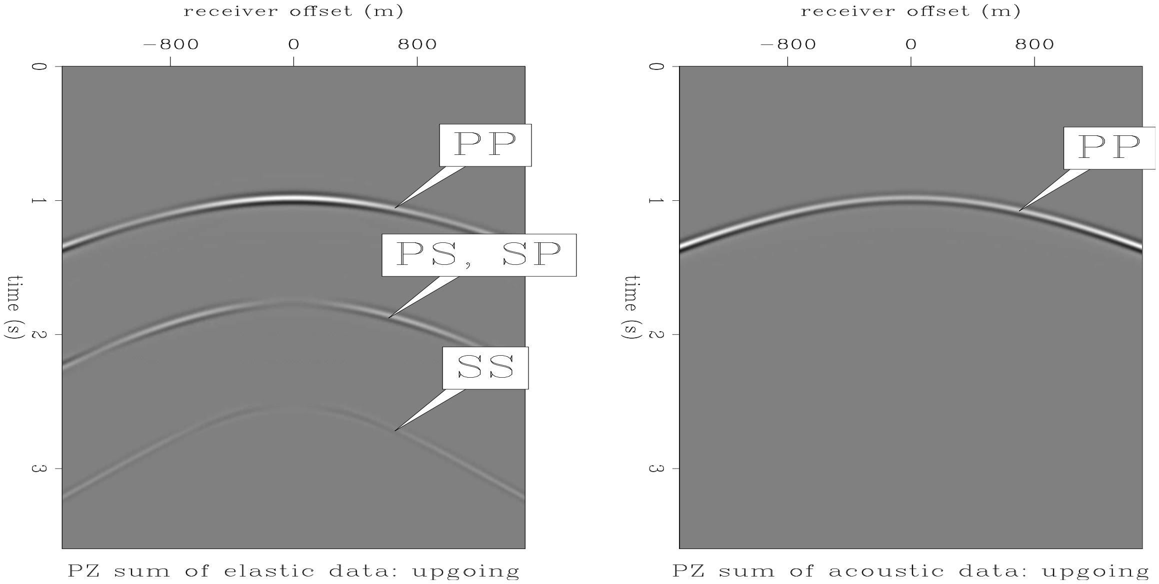


Figure 7: Result of PZ summation of elastically generated data (top) vs. acoustically generated data. The summation introduces the non-P energy into the upgoing data. [**ER**] ohad1/. 0Pr-up-acous-vs-elas

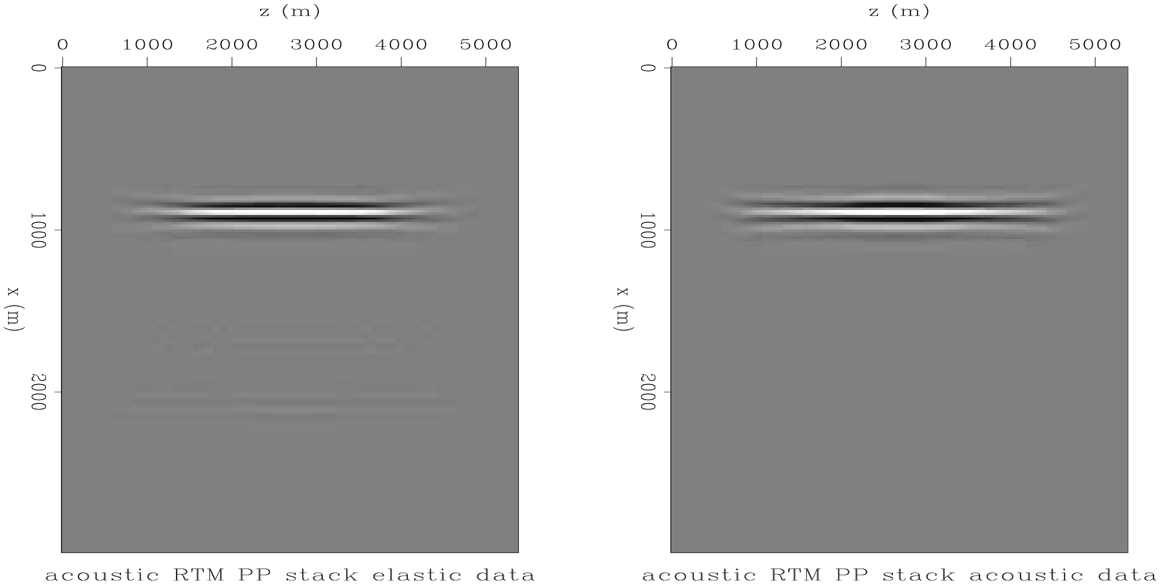


Figure 8: Comparison of the image generated by acoustic RTM of the synthetic acoustically-generated data (right) vs. acoustic RTM of the synthetic elastically-generated data (left). The images have been gained by the square of the depth and clipped at 99.7%. PZ summation is applied to the hydrophone and the vertical geophone, and the result is injected as hydrophone data only (pressure wave). Two artifacts can be seen in the migration result of the elastic data. [**ER**] ohad1/. elastic-and-acoustic-stack

results of applying the vector potential imaging condition of equation 9 to the elastic source and receiver wavefields are shown in Figure 9. Notice that each of the images has its own set of artifacts. These artifacts are the result of mode conversions occuring at the injection point of particle velocities. Whenever any particle velocity value is injected into an elastic medium, both a P and an S mode will be created, regardless of whether the recorded particle velocity was the result of a pressure or a shear wave. These modes then propagate with their respective velocities in the receiver wavefield and cross-correlate with the source wavefield at locations which are not reflector positions. They can also stack coherently over shots, as is the case here. The panels in figure 9 are not plotted at the same scale. The PS stack is an order of magnitude greater than the PP and SS stacks, and two orders of magnitude greater than the SP stack.

An interesting observation is that although each of the images in Figure 9 has different artifacts, the one thing they have in common is the true reflector position. This feature could be used to attenuate some of the converted-mode artifacts. The one problem is that to use this criterion effectively, both the V_p and the V_s models have to be quite accurate.

DISCUSSION AND CONCLUSIONS

The primary assumption of PZ summation for OBN data is that the vertical geophone contains only pressure wave energy. I have shown that doing a PZ summation on OBN-acquired hydrophone and vertical geophone data, where the vertical geophone component contains significant amounts of non-pressure wave data, will result in an introduction of the non-pressure wave energy into the summation result and subsequently degrade the image. In the simple case I used, the only non-P energy was shear waves. Running acoustic RTM with shear wave data as input will invariably image the shear wave energy away from the true reflector position, since shear waves do not propagate with an acoustic velocity. However, since the shear waves have a different moveout from the pressure waves, the stacking procedure may be sufficient to remove any artifacts they may create in the image. With elastic RTM, the shear and the pressure waves may both be imaged at correct reflector positions, but additional artifacts appear as a result of mode-conversion at the data injection point.

The realism of the scenario I've modeled here can certainly be called into question. The shear velocity in the sea bottom tends to be a rather smooth gradient, which can range from a few tens of meters per second in the topmost unconsolidated sediment, to a few hundred meters per second as depth increases and the material becomes more consolidated. Therefore the very obvious P-to-S conversion in the incident wavefield (Figure 4(c)) is probably very weak. This means that the SS reflection in the Born-modeled data I've shown will be likewise very weak , and not contribute to image artifacts. The reason for choosing a minimum shear velocity of $500 m/s$ was purely practical - so that the modeling grid would not have to be too fine. What could conceivably contribute to artifacts are the P-to-S conversions within the solid layers, as these are much stronger shear waves and are unaccounted for by the PZ summation's "P-only" assumption.

PZ summation is a more robust method than the methods mentioned in the introduction (Dankbaar (1985); Wapenaar et al. (1990); Amundsen (1993); Schalkwijk et al. (2003)), in that it requires less subsurface parameters to operate. It cannot however tell pressure wave

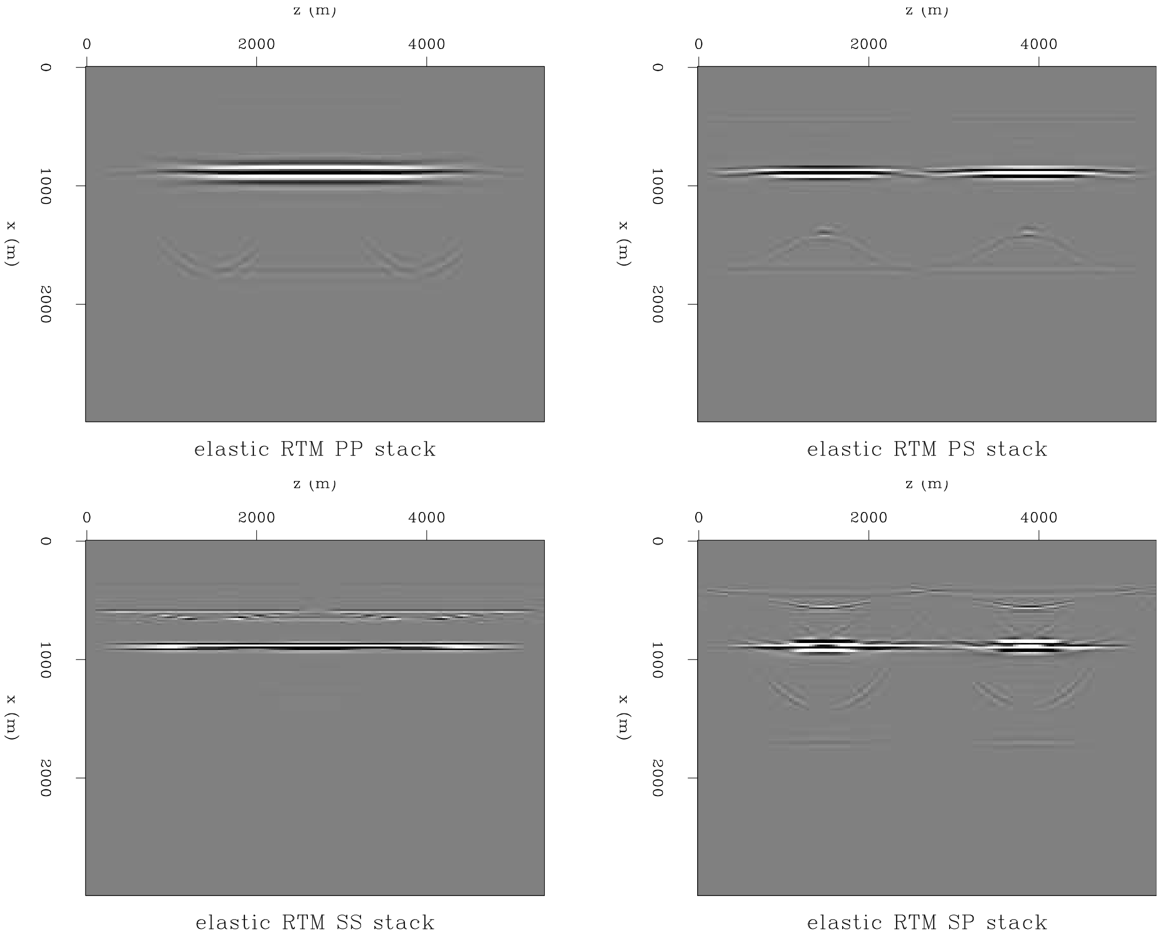


Figure 9: Images resulting from the cross-correlation of P/S source wavefield with the P/S receiver wavefield. Elastic RTM is run and the both the Born-modeled hydrophone and geophone data are injected. Top left: PP stack. Top right: PS stack. Bottom left: SS stack. Bottom right: SP stack. **NOTE** : The PS stack is an order of magnitude greater than the PP and SS stacks, and two orders of magnitude greater than the SP stack. All images have artifacts resulting from mode conversion generated at the receiver injection point. Note however that all images show the true reflector position in addition to their respective artifacts. [**ER**] ohad1/. elastic-stacks

energy from other energy, and assumes all data recorded on the hydrophone and vertical geophone must be P-wave energy.

I am currently working on a method which will be medium independent, and which will extract the pure pressure energy from the geophones. This hypothetical P-wave-only geophone data can be fed into the standard PZ summation in order to separate upgoing from downgoing wavefields in the data.

REFERENCES

Amundsen, L., 1993, Wavenumber-based filtering of marine point-source data: Geophysics, **58**, 1497–150.

Barr, F. J. and J. I. Sanders, 1989, Attenuation of water-column reverberations using pressure and velocity detectors in a water-bottom cable: SEG expanded abstracts, **8**, 653–656.

Beylkin, G. and R. Burridge, 1990, Linearized inverse scattering problems in acoustics and elasticity: Wave Motion, **12**, 15–52.

Dankbaar, J. W. M., 1985, Separation of p- and s-waves: Geophysical Prospecting, **33**, 970–986.

Dellinger, J. and J. Etgen, 1990, Wave-field separation in two-dimensional anisotropic media: Geophysics, **55**, 914–919.

Ronen, S., L. Comeaux, and X. Miao, 2005, Imaging downgoing waves from ocean bottom stations: SEG expanded abstracts, **24**, 963–966.

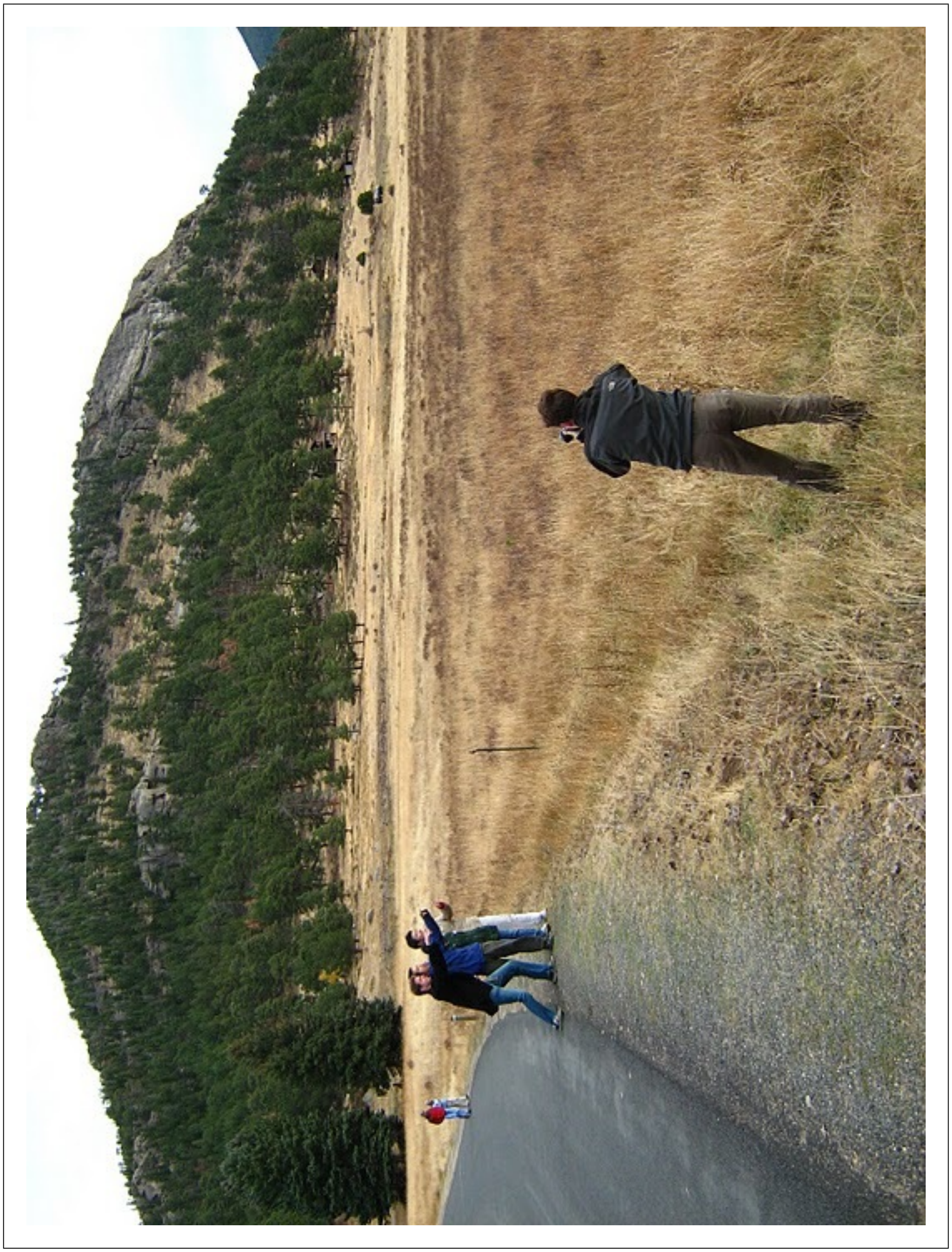
Schalkwijk, K. M., C. P. A. Wapenaar, and D. J. Verschuur, 2003, Adaptive decomposition of multicomponent ocean-bottom seismic data into downgoing and upgoing p- and s-waves: Geophysics, **68**, 1091–1102.

Virieux, J., 1986, P-sv wave propagation in heterogeneous media: Velocity-stress finite difference method: Geophysics, **51**, 889–901.

Wapenaar, C. P. A., P. Herrmann, D. J. Verschuur, and A. J. Berkhout, 1990, Decomposition of multicomponent seismic data into primary p- and s-wave responses: Geophysical Prospecting, **38**, 663–661.

Wong, M., B. L. Biondi, and S. Ronen, 2009, Inversion of up and down going signal for ocean bottom data: SEP-Report, **138**, 247–256.

Yan, J. and P. Sava, 2008, Isotropic angle-domain elastic reverse-time migration: Geophysics, **73**, S229–S239.



Stanford Exploration Project, SEP145, September 13, 2011

Krylov space solver in Fortran 2009: Beta version

Robert G. Clapp

ABSTRACT

Solving linear systems using Krylov subspace methods is an ideal candidate for object-orient programming. Iterative solver approaches use only a few different operations on vectors and operators. These operations form the basis of abstract vector and operator classes. Sophisticated solvers can then be written on top of these abstract classes separating the geophysics (operators) from the mathematics (solvers). The minimal set of object-oriented features of Fortran95 and its predecessors limited the potential separation. New inversion approaches, such as the hybrid norm, further hampered this separation when using conventional vector class descriptions. By using the object-oriented features of Fortran 2008, a separation between solvers and operators can be achieved.

INTRODUCTION

A geophysicist understands and/or approximates how a given set of earth properties (model) would create a given set of measurements (data). Geophysics is often an attempt to do the inverse: from a set of recorded data, estimate a model. When the set of measurements and/or the number of model points is large, matrix-based approaches become impractical. Iterative approaches are often the method of choice for large-scale estimation problems. Iterative solvers can become quite complex and are generally more the domain of the mathematician than the geophysicist. Ideally we would like to leverage the mathematician's expertise without having to understand all of the details of the implementation. Nichols et al. (1993); Gockenbach (1994) all implemented model estimation through an object-oriented framework, allowing this separation using C++. Schwab (1998) described a java-based approach to this problem, and Clapp (2005) described a python-based approach for large, out-of-core solvers. SEP chose instead to use Fortran 90. Unfortunately, Fortran 90 is far from a complete object-oriented language, and as a result complicated inversion problems are cumbersome to describe given its limitations. The problems encountered in implementing the hybrid norm (Claerbout, 2009; Zhang and Claerbout, 2010) is but one example of the limitations of Fortran 90 for solving inverse problems. Recently, Fortran compilers have begun to support Fortran's latest incarnation, Fortran 2008, a more complete implementation of the needed object oriented constructs. This paper is a follow up to Clapp (2010), which described how an implementation could be done in Fortran 2003, but was hampered by the immaturity of the Fortran compiler.

In this paper, I show the implementation of an abstract solver class in Fortran 2008. I begin by describing one method to implement an abstract operator-based iterative solver. I describe an abstract vector, abstract operator, and solver class. I finish by showing how a small geophysical inverse problems can be solved using these classes and concrete classes inherited from them.

OPERATOR-BASED OBJECT-ORIENTED SOLVERS

SEP (Claerbout, 1999) has traditionally taken an approach which is described as either classical, traditional, or deterministic to iterative inversion. The classical approach attempts to find the model $\mathbf{m}$ that minimizes the data misfit. Given a recorded dataset $\mathbf{d}$, and a linear operator $\mathbf{L}$, we attempt to minimize the residual vector $\mathbf{r}$ which is defined as

$$\mathbf{0} \approx \mathbf{r} = \mathbf{d} - \mathbf{L}\mathbf{m}. \tag{1}$$

In the simplest case where we are using steepest descent to solve the linear least squares inversion, we estimate $\mathbf{m}$ by mapping the initial residual (in this simple case $-\mathbf{d}$) back into the same space as the model to form a gradient vector $\mathbf{g}$ by applying the adjoint of $\mathbf{L}$. We then map the gradient vector back into data-space by applying $\mathbf{L}$ to form $\mathbf{\Delta r}$. Finally, we find the scaling factor of $\mathbf{\Delta r}$ that will make $\mathbf{r} + \mathbf{\Delta r}$ as small as possible. We then repeat this procedure until $\mathbf{r}$ is suitably small. More complex inversion approaches are normally built on this basic concept.

Vector class

The solver writer doesn't need to know anything about $\mathbf{L}$ other than how to apply it and its adjoint. In fact, the solver writer doesn't need to know much about $\mathbf{m}$ or $\mathbf{d}$. The steepest descent approach described above involves only three mathematical operations. In order to find the best scaling factor $\mathbf{\Delta r}$, we need to calculate a dot product. In order to update the model and the residual, we will need to scale $\mathbf{\Delta r}$ and add it to $\mathbf{r}$. We can define the interface for calling the forward of $\mathbf{L}$ as

```
call lop (logical add, vec m, vec d)
```

If the `vector` class has the ability to perform the add, scale, and dot product functions, we can begin to write a generic solver. Two more initialization functions are needed in the `vector` class. We need to be able to create the gradient vector before we can apply the adjoint. As a result, we need to be able to create a vector with the same number of elements as the model. Put another way, we need to *clone* the model. We also need to be able to zero this vector, or *set* the vector to some value. It is also useful to seperate the space a vector exists in and the storage mechanism. The vector space contains thing like whether we are dealing irregular or regular dataset, the number of samples, and the locations of the samples (for example the origin and sampling of the axes). The table below gives a list of the abstract vector class components.

Function	**Purpose**
`add(vector)`	Add another vector to current vector ($x = x + y$).
`scale(real)`	Scale the vector ($x = x * a$).
`scale_add(scale1,vector,scale)`	Add another vector to current vector ($x = ax + by$).
`scale(vector)`	Calculate the dot product with another vector (return $sum_i a(i) * b(i)$.
`set(real)`	Set the value of a vector ($x = v$).
`mult(real))`	Multiply a vector with another vector ($y = y * x$).
`clone(vec)`	Create another vector of the same type with the same values.
`clone_space(vec)`	Create another vector of the same type with no storage mechanism.
`check_same(vec)`	See if two of vector of the same type and exist in the same space.
`alloc()`	Create a vector from a vector space.
`info(character(len=*),integer)`	Provide user specified debugging information.

From this base class I inherit a `real` vector class and then 1-D to 7-D `real` from this class (Figure1). Further non-uniform classes would inherit form the `real` vector class while out-of-core classes would come directly from the `vector` class.

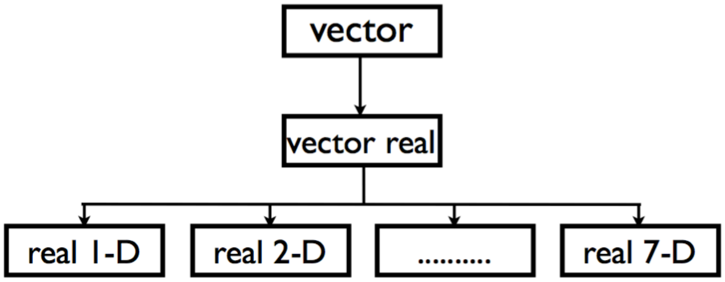


Figure 1: Inheritance class for vectors. The 1-D to 7-D real vectors are inheriting from a real vector class which is inherited from the virtual vector class. [**NR**] bob1/. inher

Operators

The base operator class contains the ability to perform a mapping from the vector-space of the operators domain, to the vector space of operator's range (the forward of the operator), and vice versa (applying the adjoint). It is beneficial for an operator to store a description of these two spaces (the reason for the clone-space function described above). This performs two functions. First, the operator can perform a sanity check to make sure that the spaces of model and data passed into the forward or adjoint function call match the space of initialized domain and range. The second reason is that inversion problems are often more complicated then the generic problem described by equation 1. For example, if **L** is actually the cascade of two operator **A** and **B**,

$$\mathbf{L} = \mathbf{A}\mathbf{B} \tag{2}$$

we need the ability to check that the domain of is **A** is equivalent to the range of **B** and we need to create a vector of that size to hold the intermediate result of applying **B** in the forward case (and **A** in the case of the adjoint).

A Fortran 2008 `type` must be declared in a module. An example of an operator declaration is seen below. In this case I am creating a causal integration operator.

```
module causal_mod
 use operator_mod  !Description of the generic operator class
 use vec_nd_mod    !The specific vector class used in this module
 implicit none
 type,extends(operator) ::causint_op  !Causint operator declaration
   contains
     procedure,pass :: op=>causint_it !Pointer to the generic forward/adj
     final:: causint_close  !clean function (not necessary in this case)
 end type
```

When leaving a functional unit that has declared a type or when deallocating a type the `final` function is called. In this case we haven't allocated any memory but for completeness, and for debugging, it is often useful to include it.

```
 subroutine causint_close(myop)
   type(causint_op) :: myop
 end subroutine
```

We need a subroutine that sets up a causal integration operator. The only information we need is the size of the vector space in which we are going to be performing causal integration on.

```
 subroutine create_causint_op(myop,v)
   class(vector) :: v   !vector space we are operating on
   class(causint_op) :: myop  !operator we are setting up
   logical :: bm=.false.  !default to having the wrong type of vector
   myop%lab=1
   select type(v)   !check the type of vector
     class is (real_1d) !make sure it is a 1-D real vector
       bm=.true.  !we have the right type of vector
   end select

   if(.not. bm) call seperr("model and vector must be real_1d")
   call myop%set_domain_range(v,v) !store the domain and range
 end subroutine
```

The only thing left is the actual operator. I am going to break it into to parts. The first part is the initialization and the overhead associated with Fortran 2008.

```
 subroutine causint_it(myop,adj,add,mod,dat)
   class(causint_op) :: myop  !causal integration object
   logical, intent(in) :: adj,add
   class(vector) :: mod,dat  !vector spaces
   real,dimension(:), pointer :: xx,yy
   real,dimension(:,:),pointer :: ar
   integer :: i,j,nm,nd
   real ::     t
   !Check to make sure the model and data vector
   !spaces match those stored in the operator declaration
   if(.not. mod%check_same(myop%domain))&
     call seperr("domains don't match")
   if(.not. dat%check_same(myop%range)) &
     call seperr("ranges don't match")
   call adj_null(adj,add,mod,dat)

   !Create a pointer to the model values
   select type(mod)
     class is (real_1d)
        xx=>mod%vals
        nm=size(mod%vals)
   end select

   !Create a pointer to the data values
   select type(dat)
     class is (real_1d)
       yy=>dat%vals
       nd=size(dat%vals)
   end select
```

The second part is standard Fortran77/90/2003/2008.

```
    t=0
   if(adj) then
     do i= nd, 1, -1
         t = t + yy(i)
        xx(i) = xx(i) + t
     end do
   else
      do i= 1, nd
          t   = t + xx(i)
          yy(i) = yy(i) + t
      end do
   end if
```

```
  end subroutine
end module
```

Combining operators

The number of different ways that an operator might need to be combined to solve a given inversion problem is infinite. Fortunately, all possible combinations can be built from four building blocks. The first is a chain operator. When the results of applying the first operator $\mathbf{L_1}$ is fed into a second operator $\mathbf{L_2}$,

$$\mathbf{d} = \mathbf{L_2 L_1 m}. \tag{3}$$

A second applies two different operators to the same vector (a column vector),

$$\begin{bmatrix} \mathbf{d_1} \\ \mathbf{d_2} \end{bmatrix} = \begin{bmatrix} \mathbf{L_1} \\ \mathbf{L_2} \end{bmatrix} \mathbf{m}. \tag{4}$$

Its corollary, a row operator, which forms a single data $\mathbf{d}$ from to models,

$$d = \begin{bmatrix} \mathbf{L_1} & \mathbf{L_2} \end{bmatrix} \begin{bmatrix} \mathbf{m_1} \\ \mathbf{m_2} \end{bmatrix}. \tag{5}$$

Finally, a diagonal operator that applies different operators to different models

$$\begin{bmatrix} \mathbf{d_1} \\ \mathbf{d_2} \end{bmatrix} = \begin{bmatrix} \mathbf{L_1} & \\ \mathbf{L_2} & \end{bmatrix} \begin{bmatrix} \mathbf{m_1} \\ \mathbf{m_2} \end{bmatrix}. \tag{6}$$

The final three all imply the creation of a new vector class that is the combination of two or more vectors. This super vector class is a storage object that calls the appropriate vector class function sequentially (except for the dot product function that must add the result of each call). As described in the next section inversion problems are often combinations of several of these combo-operator/vectors and these functions are often called recursively.

Solvers

An iterative solver operates one a problem that can be described as simply as equation 1. Translating a complicated problem into this simple form is a more complex problem. The problem is broken up into three steps: obtaining an initial residual, finding the vector that best solves the constructed inverse problem, and updating the model according to this vector. Each one of these steps involve several different potential user inputs. For simplicity, I am going to describe all potential inversion problems in terms of a regularized inversion problem with two fitting goals (each goal could be actually multiple fitting goals combined using the functions described above).

The first step involves constructing the data space vectors, $\mathbf{r_d}$ and $\mathbf{r_m}$. The user might begin by specifying some initial values for these two vectors. These values then need to be updated according to the data $\mathbf{d}$ associated with the problem, a potential initial model $\mathbf{m_0}$, the operators being used $\mathbf{L_1}, \mathbf{L_2}$, and weights applied to the residual $\mathbf{W_0}, \mathbf{W_1}$.

$$\left[\begin{array}{c} \mathbf{r_d} \\ \mathbf{r_m} \end{array}\right] = \left[\begin{array}{c} \mathbf{r_d} \\ \mathbf{r_m} \end{array}\right] + \left[\begin{array}{cc} \mathbf{W_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{W_2} \end{array}\right] \left(\left[\begin{array}{c} \mathbf{d} \\ \mathbf{0} \end{array}\right] - \left[\begin{array}{c} \mathbf{L_1} \\ \mathbf{L_2} \end{array}\right] \mathbf{m_0} \right). \tag{7}$$

Once the initial residual is calculated, we iterate to find $\mathbf{x}$ through,

$$\left[\begin{array}{c} \mathbf{r_d} \\ \mathbf{r_m} \end{array}\right] = \left[\begin{array}{c} \mathbf{W_1 L_1} \\ \mathbf{W_2 L_2} \end{array}\right] \mathbf{Sx}, \tag{8}$$

where $\mathbf{S}$ is a preconditioning operator. Finally we need update our model,

$$\mathbf{m} = \mathbf{m_0} + \mathbf{Sx}. \tag{9}$$

This procedure allows a single solver to be written for a myriad of different inverse problems. It also demonstrates one of the biggest weaknesses of Fortran 90. Fortran 90 does not support function pointers. As a result, SEP has traditionally written different solvers for regularized and preconditioned problems. Combination operators could only be created by writing a function that specifically named the two operators that were to be combined. As a result, creating complex inversion problems quickly became cumbersome and prone to errors.

EXAMPLE

Converting RMS velocities to interval velocities is one of the most basic problems in reflection seismology. The Dix equation is one of the most common approaches but often leads to unrealistic velocities when dealing with the noise in the RMS velocity measurement. Clapp et al. (1998) points out that there is a linear relationship between $\mathbf{v}^{\mathbf{2}}_{\rm rms}$ and $\mathbf{v}^{\mathbf{2}}_{\rm int}$ using either a modified version of causal integration or using causal integration $\mathbf{C}$ directly and first scaling $\mathbf{v}^{\mathbf{2}}_{\rm rms}$ by sample number. With this linear relation we can now add a model styling goal, such as smooth $\mathbf{v}^{\mathbf{2}}_{\rm int}$, given us the fitting goals

$$\begin{array}{rcl} \mathbf{d} & \approx & \mathbf{Cm} \\ \mathbf{0} & \approx & \epsilon \mathbf{Dm}, \end{array} \tag{10}$$

where $\mathbf{D}$ is the derivative operator, $\mathbf{d}$ is the scaled $\mathbf{v}^{\mathbf{2}}_{\rm rms}$, and the model is $\mathbf{v}^{\mathbf{2}}_{\rm int}$. We need a weighting term which gives higher importance to good RMS picks, and equal weights all of the RMS velocities (undoing the effect of the sample number scaling), resulting in

$$\begin{array}{rcl} \mathbf{0} & \approx & \mathbf{W}(\mathbf{d} - \mathbf{Cm}) \\ \mathbf{0} & \approx & \epsilon \mathbf{Dm}. \end{array} \tag{11}$$

We can precondition the model by noting that causal integration is the inverse of the derivative except at the first sample. We know that first interval velocity value is the same as the first RMS velocity, resulting in the final setting of fitting equations,

$$\begin{array}{rcl} \mathbf{0} & \approx & \mathbf{W}(\mathbf{d} - \mathbf{CMp}) \\ \mathbf{0} & \approx & \epsilon \mathbf{p}, \end{array} \tag{12}$$

where **p** is the preconditioned variable and **M** is a masking operator that doesn't allow the first value to change. The advantage of selecting this problem is that its solution is a rather thorough test of all the necessary features of the solver. It involves a starting model, a weighting operator, and a masking operator. It requires both chaining operators and making column operator objects. All of the hard stuff is done away from the user in the solver, the user only needs to construct the required operators and initialize and run the solver.

For this example the conversion is all handled in a module. The module begins by `using` all of the modules that declare the operators and solvers, and the declaration of variables.

```
module vrms_2_int_mod !Transform from RMS to interval velocity
  use causint_mod  !Causal integration
  use weight_mod   !Weighting operator
  use mask_mod     !Masking operator
  use cg_step_mod  !Conjugate gradient
  use obj_solver_mod !Solver module
contains
  subroutine vrms2int( niter, eps, weight, vrms, vint)
    integer,            intent( in)     :: niter     ! iterations
    real,               intent( in)     :: eps       ! scaling parameter
    type(real_1d)                       :: vrms      ! rms velocity
    type(real_1d),target                       :: vint! interval velocity
    real, dimension(:), pointer         :: weight    ! data weighting
    integer                             :: st,it,nt
    logical, dimension( size( vint%vals))    :: mask
    logical, dimension(:), pointer      :: msk
    real,    dimension( size( vrms%vals))    :: wt
    real, dimension(:), pointer         :: wght
    type(prec_solver)  :: p_s   !Preconditioned solver

    type(causint_op),target :: ca_op,ca2_op   !Causal integration
    type(mask_op),target    :: m_op           !Masking operator

    type(weight_op),target :: wt_op   !Weighting operator
    type(cgstep),target :: cg   !Conjugate gradient operator
    type(real_1d),target :: dat  !Data
```

Next we need to scale the data, create the weighting vector, and masking vector.

```
vrms2int.f90        vrms2int.unrat      vrms_2_int_mod.mod
    nt = size( vrms%vals)

    call create_vec1(dat,vrms%vals)
         do it= 1, nt
           dat%vals( it) = vrms%vals( it) * vrms%vals( it) * it
           wt( it) = weight( it)*(1./it)           ! decrease weight with time
         end do
```

```
    mask = .false.;   mask( 1) = .true.            ! constrain first point
    vint%vals = 0.     ;   vint%vals(1)= dat%vals( 1)

    allocate(msk(size(mask)))
    msk=.not.mask

    allocate(wght(size(wt)))
    wght=wt
```

Finally we need to initialize the operators, setup the solver, and solve for the interval velocity squared.

```
    call create_weight_op(wt_op,vrms,wght) !Create weighting op
    call create_causint_op(ca_op,vint,"a1") !Causal op
    call create_causint_op(ca2_op,vint,"a2")!Preconditioning
    call create_mask_op(m_op,vint,msk)  !Masking operator

    p_s%lop=>ca_op;   !Mapping operator
    p_s%st=>cg;       !Step function
    p_s%pop=>ca2_op   !Preconditioning operator
    p_s%dat=>dat;     !Data
    p_s%mod=>vint     !model
    p_s%jop=>m_op;    !Masking operator
    p_s%wop=>wt_op    !Weighting operator
    p_s%eps=eps;      !Scaling factor
    p_s%p0=>vint      !Initial preconditioned variable
    call p_s%solve(niter)  !Solve for interval v^2

    call ca_op%op(.false.,.false.,vint,dat)  !Estimated RMS^2

    do it= 1, nt
       vrms%vals( it) = sqrt( dat%vals( it)/it)  !RMS velocity
    end do
    vint%vals = sqrt( vint%vals) !Interval velocity
    deallocate(wght); deallocate(msk)
 end subroutine
 end module
```

Figure 2 shows the result of running the inversion. The left panel shows the original RMS velocity and the mapped RMS velocity. The right panel shows the estimated interval velocity.

CONCLUSIONS

Iterative-based inversion maps cleanly into an object-oriented framework. Vector, operator, and solver abstract classes can be built upon to solve nearly any inversion problem. The

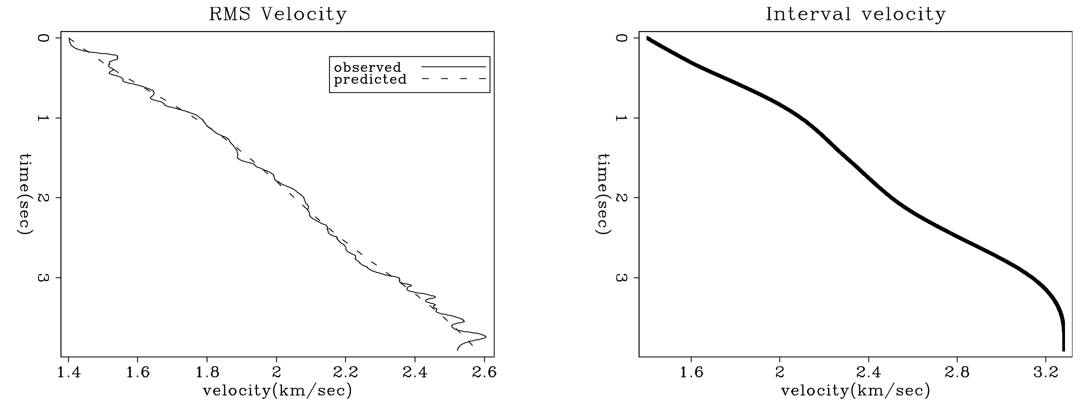


Figure 2: The left panel shows the original RMS velocity and the mapped RMS velocity. The right panel shows the estimated interval velocity. [**ER**] bob1/. stiff

Fortran 2008 standard contains all of the object-oriented features needed to write an inversion library. The resulting inverse code is more verbose than the Fortran 90 approach but the added flexibility makes this an acceptable penalty.

REFERENCES

Claerbout, J., 1999, Geophysical estimation by example: Environmental soundings image enhancement: Stanford Exploration Project.

———, 2009, Blocky models via the l1/l2 hybrid norm: SEP-Report, **139**, 1–10.

Clapp, R. G., 2005, Inversion and fault tolerant parallelization using Python: SEP-Report, **120**, 41–62.

———, 2010, Hybrid-norm and fortran 2003: Separating the physics from the solver: SEP-Report, **142**, 85–92.

Clapp, R. G., P. Sava, and J. F. Claerbout, 1998, Interval velocity estimation with a null-space: SEP-Report, **97**, 147–156.

Gockenbach, M. S., 1994, Object-oriented design for optimization and inversion software: A proposal: TRIP-Report, **1994**, 1–24.

Nichols, D., H. Urdaneta, H. I. Oh, J. Claerbout, L. Laane, M. Karrenbach, and M. Schwab, 1993, Programming geophysics in C++: SEP-Report, **79**, 313–471.

Schwab, M., 1998, Enhancement of discontinuities in seismic 3-D images using a Java estimation library: **99**.

Zhang, Y. and J. Claerbout, 2010, Least-squares imaging and deconvolution using the hb norm conjugate-direction solver: SEP-Report, **140**, 129–142.

Stanford Exploration Project, SEP145, September 13, 2011

Data examples of logarithm Fourier-domain bidirectional deconvolution

Qiang Fu, Yi Shen and Jon Claerbout

ABSTRACT

Time-domain bidirectional deconvolution methods show great promise for overcoming the minimum-phase assumption in blind deconvolution of signals containing a mixed-phase wavelet, such as seismic data. However, time-domain bidirectional methods usually suffer from slow convergence (Slalom method) or the starting model (Symmetric method). Claerbout proposed a logarithm Fourier-domain method to perform bidirectional deconvolution. In this paper, we test the new logarithm Fourier-domain method on both synthetic data and field data. The results demonstrate that the new method is more stable than previous methods and that it produces better results.

INTRODUCTION

Usually, a seismic data trace d can be defined as a convolution of a wavelet w with a reflectivity series r. This can be written as $d = r * w$, where $*$ denotes convolution. Blind deconvolution seeks to estimate the wavelet and reflectivity series using only information contained in the data. Traditionally, seismic blind deconvolution has two assumptions, namely whiteness and minimum phase. The whiteness assumption supposes that the reflectivity series r is a flat spectrum, while the minimum-phase assumption supposes that the wavelet w is causal and has a stable inverse. Recently, some new methods have been proposed to avoid or correct these two assumptions in seismic blind deconvolution.

In Zhang and Claerbout (2010a), the authors proposed to use a hyperbolic penalty function introduced in Claerbout (2009) instead of the conventional L2 norm penalty function to solve the blind deconvolution problem. With this method, a sparseness assumption replacese of the traditional whiteness assumption in the deconvolution problem. Furthermore, Zhang and Claerbout (2010b) proposed a new method called “bidirectional deconvolution” in order to overcome the minimum-phase assumption. Bidirectional deconvolution assumes that any misxed-phase wavelet can be decomposed into a convolution of two parts: $w = w_a * w_b$, where w_a is a minimum-phase wavelet and w_b is a maximum-phase wavelet. To solve this problem, we estimate two deconvolution filters, a and b, which are the inverses of wavelets w_a and w_b, respectively. Since Zhang and Claerbout (2010b) solve the two deconvolution filters a and b alternately, we call this method the slalom method. Shen et al. (2011a) proposed another method to solve the same problem. They use a linearized approximation to solve the two deconvolution filters simultaneously. We call this method the symmetric method. Fu et al. (2011) proposed a way to choose an initial solution to overcome the local-minima problem caused by the high nonlinearity of blind deconvolution. Shen et al. (2011b) discuss an important aspect of any inversion problem, preconditioning and how it improves bidirectional deconvolution.

All of the forementioned methods solve the problem in the time domain. Claerbout et al. (2011) proposed a method to solve the problem in the Fourier domain. We will show in a later section that this new method converges faster and is less sensitive to the starting point or preconditioner than the above-mentioned time-domain methods.

METHODOLOGY

Claerbout et al. (2011) show the complete derivation of the method. Here, we describe only the major steps of this method. As with any iterative method, we have two issues to solve in one iteration: the update direction and the step length of the update. Below, we disscuss how we can solve these two issues in the logarithm Fourier-domain method.

As we discussed in the previous section, we can decompose the arbitrary data d into three parts: the reflectivity series r, the minimum phase wavelet w_a and the maximum phase wavelet w_b:

$$d = r * (w_a * w_b). \tag{1}$$

We wish to solve for the deconvolution filters a and b, which should be the inverses of wavelets w_a and w_b:

$$\left\{ \begin{array}{l} w_a * a = \delta(n) \\ w_b * b = \delta(n) \end{array} \right. . \tag{2}$$

From equation 2, we know that a is minimum phase and b is maximum phase. If we know the deconvolution filters a and b, we can get reflectivity series r as follows:

$$r = d * a * b. \tag{3}$$

Next we transform our problem into the Fourier domain. We use capital letters to denote variables in the Fourier domain:

$$R = DAB. \tag{4}$$

We use U to denote the logarithm of the product of A and B:

$$U = \log(AB). \tag{5}$$

Our problem then becomes

$$R = De^U, \tag{6}$$

where U has become our new unknown in bidirectional deconvolution, and we want to update it in each iteration. After some derivation (Claerbout et al., 2011), we get, in the time domain,

$$\left\{ \begin{array}{l} \Delta u = r^{\circledast}\mathrm{Hyp}'(r) \\ \Delta r = r * \Delta u \end{array} \right. , \tag{7}$$

where $^{\circledast}$ means cross-correlation and $\mathrm{Hyp}(r_i) = \sqrt{r_i^2 + R_0^2} - R_0$ is the hyperbolic penalty function.

By Newton's method (using the only first 2 terms of the Taylor expansion), we can calculate the step length α:

$$\alpha = \frac{\sum_i \text{Hyp}'(r_i)\Delta r_i}{\sum_i \text{Hyp}''(r_i)\Delta r_i^2}. \tag{8}$$

Because we use Newton's method, this step length α calculated above is not the final value. To obtain the final step length at each iteration, we need another iteration (nested or second-order iteration):

```
$\alpha_j = 0$
Iterate($j$)
    $\alpha_j = \frac{\sum_i \text{Hyp}'(r_i)\Delta r_i}{\sum_i \text{Hyp}''(r_i)\Delta r_i^2}$
    $\alpha_{final} = \alpha_{final} + \alpha_j$
    $r = r + \alpha_j \Delta r$
    $u = u + \alpha_j \Delta u$
```

Given the update directions (both for the unknown u and for the residual r) and the step length α of the update, we have everything we need for each iteration. We can iterate to convergence.

Trial and error on step length α

Because the method above can lead to over-shooting on the step length α, it may lead to a blow-up problem. The Newton method requires a convex function, but for some field data sets that condition may not be met. To overcome this, we use trial and error to avoid step length α being too large. If the hyperbolic penalty function on new $r = r + \alpha\Delta r$ is greater than on original r, the step length α is too large, and we overshoot; in that case, we reduce the step length α by half until the stability condition is met.

```
$\alpha_j = 0$
Iterate($j$)
    $\alpha_j = \frac{\sum_i \text{Hyp}'(r_i)\Delta r_i}{\sum_i \text{Hyp}''(r_i)\Delta r_i^2}$
    Iterate
        If   $\text{Hyp}(r + \alpha_j \Delta r) <= \text{Hyp}(r)$   Then   Break
        $\alpha_j = \alpha_j / 2$
    $\alpha_{final} = \alpha_{final} + \alpha_j$
    $r = r + \alpha_j \Delta r$
    $u = u + \alpha_j \Delta u$
```

EXAMPLES

In this section we will test the logarithm Fourier-domain method with both synthetic and field applications. As prior experience shows (Shen et al., 2011b), preconditioning is a critical part of the seismic blind deconvolution problem. Here, we use the Burg Prediction Error Filter (PEF) as the preconditioner for all tests. Unless otherwise specified, we will use one filter for all traces.

In the implementation of the logarithm Fourier-domain method, we use the steepest descent method to search the solution, and for the time domain symmetric method, we use the conjugate direction method. Because of this difference, the convergence speed or the number of iterations required for convergence may not be compared directly.

Synthetic 1D example

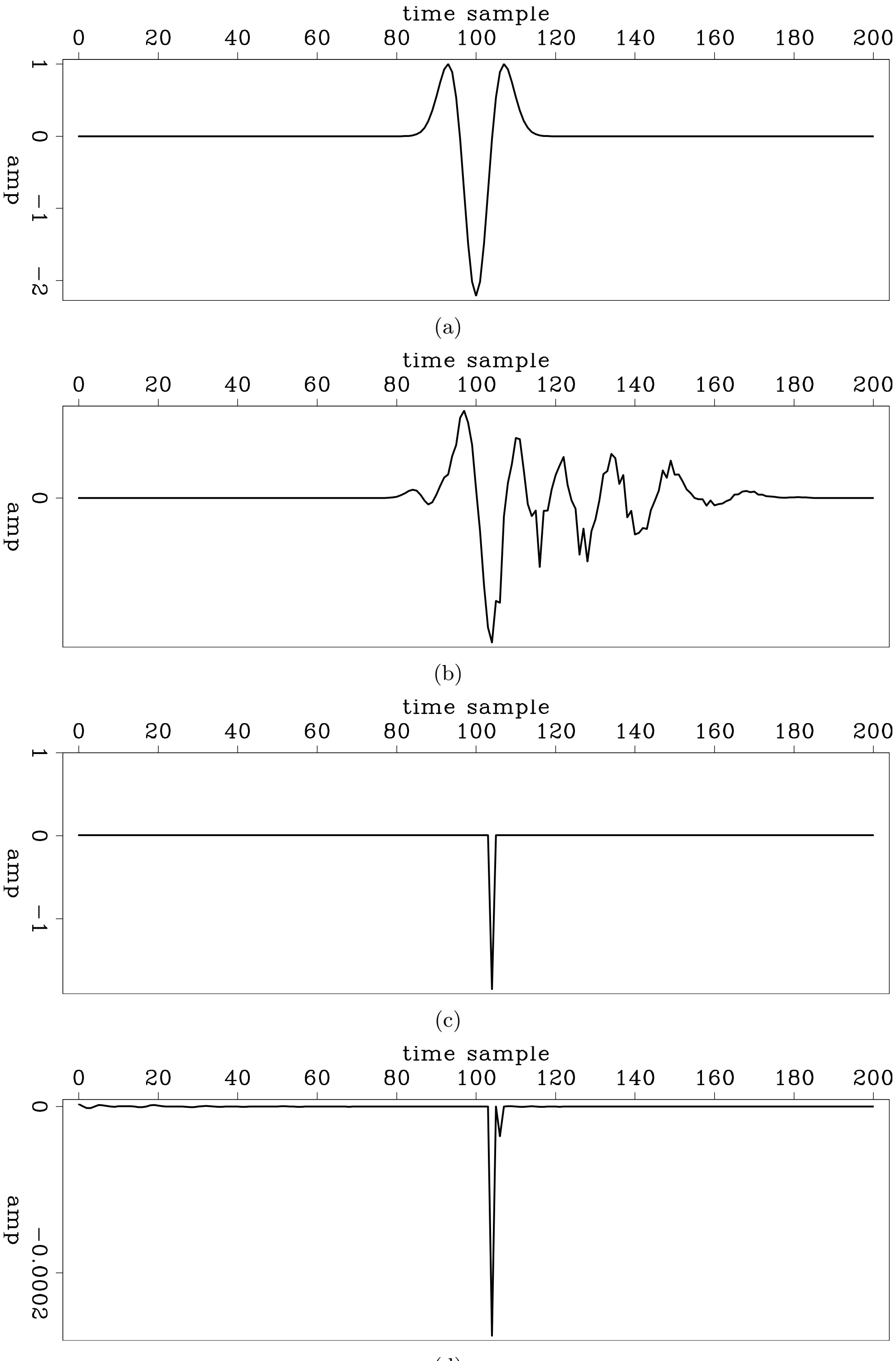
First, we demonstrate the advantage of the logarithm Fourier-domain method over the time-domain method on a very simple synthetic data example. The input is a Ricker wavelet generated by an approximate approach that applies a second-order derivative to binomial coefficients (Fu et al., 2011). Figure 1(a) shows a synthetic Ricker wavelet, and 1(b) shows the Ricker wavelet after Burg PEF preconditioning. Next we use the symmetric method (Shen et al., 2011a) to perform time-domain bidirectional deconvolution. Figures 1(c) and 1(d) compare the results of bidirectional deconvolution using the two methods.

For this simple 1D synthetic example, we use 0.1 (which is 5% of the peak amplitute of the input) as the threshold of the hyperbolic penalty function for the logarithm Fourier-domain method and use 95% quantile of the data residual as the threshold for the time-domain method. Using the logarithm Fourier-domain method, we turn the Ricker wavelet into a spike output after about 50 iterations. Using the time-domain symmetric method, even after 30,000 iterations, we get a major spike followed by a minor spike, plus a few additional jitters at the begining of the trace. At the time of this publication, we do not fully understand why the symmetric method, which utilizes a conjugate direction solver, is significantly slower than the logarithmic method, which utilizes a steepest descent algorithm.One possible explanation, which has not been tested, is that the deconvolution filters derived from the time domain symmetric method are not strictly minimum and maximum phase wavelets.

Another important observation from this synthetic test case is the output location of the bidirectional deconvolution. If we check the wiggles carefully, we can measure that the major peaks of the two deconvolution results in figures 1(c) and 1(d) (both of which are at time sample 104) are not the same as the location of the major peak of the input data figure 1(a) (which is at the time sample 100). Instead, they are located at the major peak location of the preconditioning result in figure 1(b). This inspired us to realize that the output spike location of the deconvolution is determined by the preconditioner, and that we can change the preconditioner to move the spike of the deconvolution result to the location desired. In another paper (Shen et al., 2011b), we discuss this interesting topic in more detail.

Field data common-offset gather example

Now we test the logarithm Fourier-domain bidirectional deconvolution with a 2D marine common offset gather data set that is commonly used to test the time-domain bidirectional deconvolution methods. This 2D marine common-offset gather is very popular in papers



(a)

(b)

(c)

(d)

Figure 1: (a) The synthetic Ricker wavelet; (b) The Ricker wavelet after Burg PEF preconditioning; (c) The bidirectional deconvolution result of the logarithm Fourier-domain method after 50 iterations; (d) The bidirectional deconvolution result of the time-domain symmetric method after 30,000 iterations. [**ER**] qiang/. ricker,ricker-decon,ricker-log,ricker-linear

discussing blind deconvolution using the hyperbolic penalty function. In Zhang and Claerbout (2010a), Zhang and Claerbout (2010b), Fu et al. (2011), Shen et al. (2011a) and Shen et al. (2011b) the authors tested their methods and theories with this data set as a field data example. Hence this is a good choice for comparing this new method with previous ones.

Figure 2 shows the 2D marine common offset gather. Figure 3 shows the common-offset section after Burg PEF preconditioning. Figures 4 and 5 compare the results of two methods of bidirectional deconvolution.

Figures 6(a) and 6(b) show the comparison of the estimated wavelets from the two methods. Recall that the estimated wavelet is the inverse deconvolution filter. We get the inverse filter by inverting the frequency spectrum of the filter in the Fourier domain, so the wavelet waveform is periodic. This means the jitter we see at the end of the wavelet is the anti-causal part of the filter.

From figures 4 and 5, both methods provide good quality results. However we think the logarithm Fourier-domain method (figure 4) is a little better. First, there are fewer precursors in the the logarithm Fourier domain result than in the time domain result. And the precursor of the logarithm Fourier-domain result does not have a sharp boundary as the precursor of the time domain result. The sharp boundary of the time domain result is caused by the anti-causal deconvolution filter b, which is 20 sample long, and a sharp boundary on the beginning of the anti-causal deconvolution filter b. Because the logarithm Fourier-domain uses a periodic deconvolution filter, the anti-causal part of the deconvolution filter has no beginning. Moreover, within the salt body (in the vicinity of 2.4 s to 2.6 s), the Fourier-domain result looks cleaner. We find within the salt body, there are fewer low frequency events (which can be seen as the white stripe in figure 5 within 2.4 s to 2.6 s). We do not expect to see any features within the salt body, therefore all events we see in this area in the raw data are the air-gun bubbles. This indicates that the Fourier-domain deconvolution handles the air-gun bubbles better than the time-domain method. However, excluding these differences, the results are similar. We can also find that, except for the tail parts, the estimated wavelets (figures 6(a) and 6(b)) are similar. At the tail of the wavelet which, because of periodicity, correspond to the anticausal part of the logarithm Fourier domain wavelet the wavelet estimated by the time domain symmetric method has less jitter. On the contrary, the logarithm Fourier-domain wavelet appears to have a small anti-causal air-gun bubble.

Here, we have a paradox. From the comparison of deconvolution result (figures 4 and 5), we conclude that in terms of quality, the the logarithm Fourier-domain method is better. But from the wavelet comparison, we can reach an oppsite conclusion. This is because the deconvolution filter in the logarithm Fourier-domain method is periodic and the length of anti-causal part has not an extra constraint, whereas the anti-causal filter in the time domain symmetric method is constrained by the anti-causal filter length parameter and tends to have less anti-causal jitter in the final estimated wavelet.

CONCLUSIONS

We test the logarithm Fourier-domain method using both synthetic and field data sets. The results show that this Fourier-domain method has advantages over previous time-domain

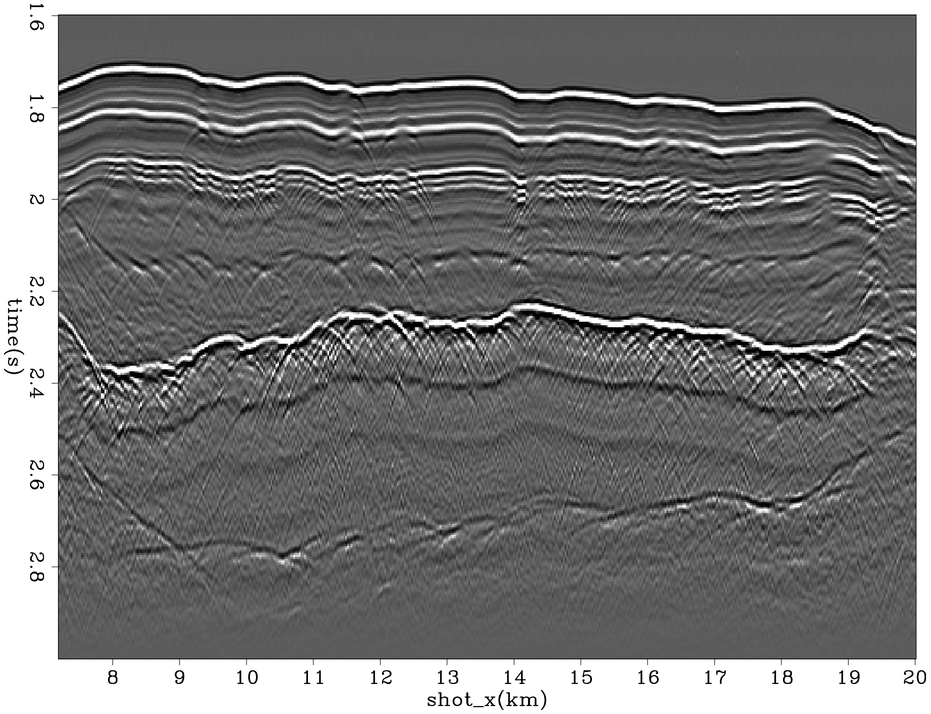


Figure 2: A common-offset section of a marine survey. [**ER**] qiang/. cof-data

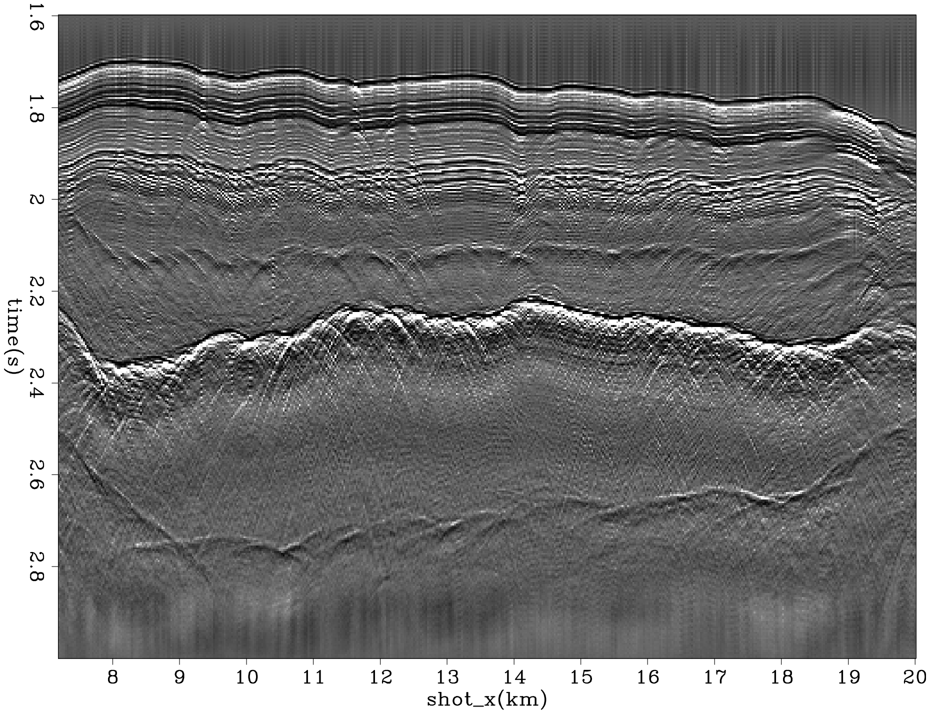


Figure 3: The common-offset section after applying the Burg PEF preconditioner. [**ER**] qiang/. cof-data-decon

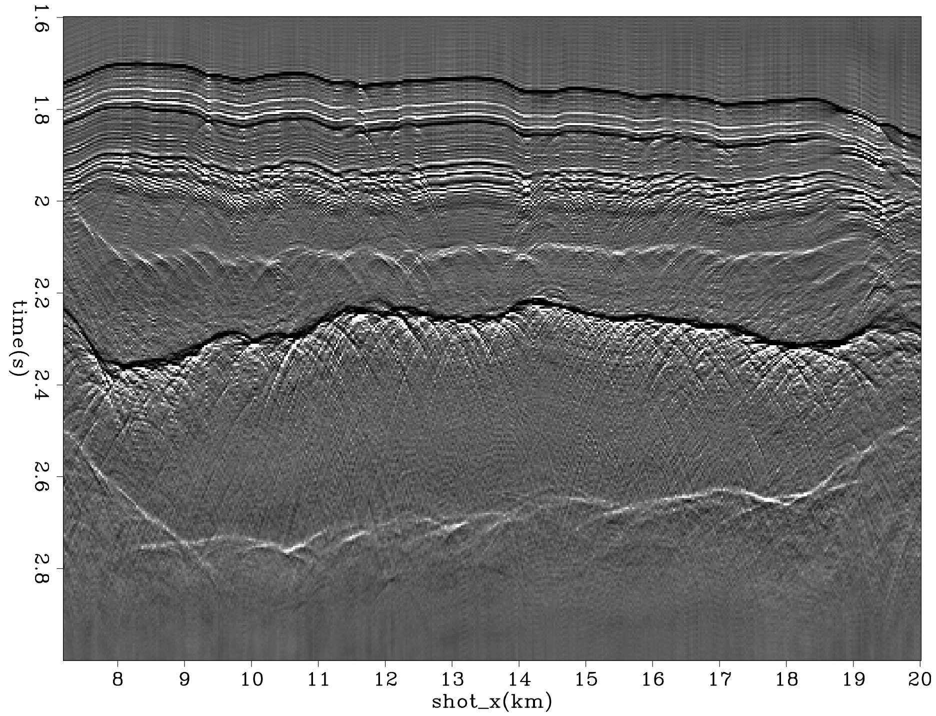


Figure 4: Logarithm Fourier-domain bidirectional deconvolution result. [**ER**] qiang/. cof-data-log

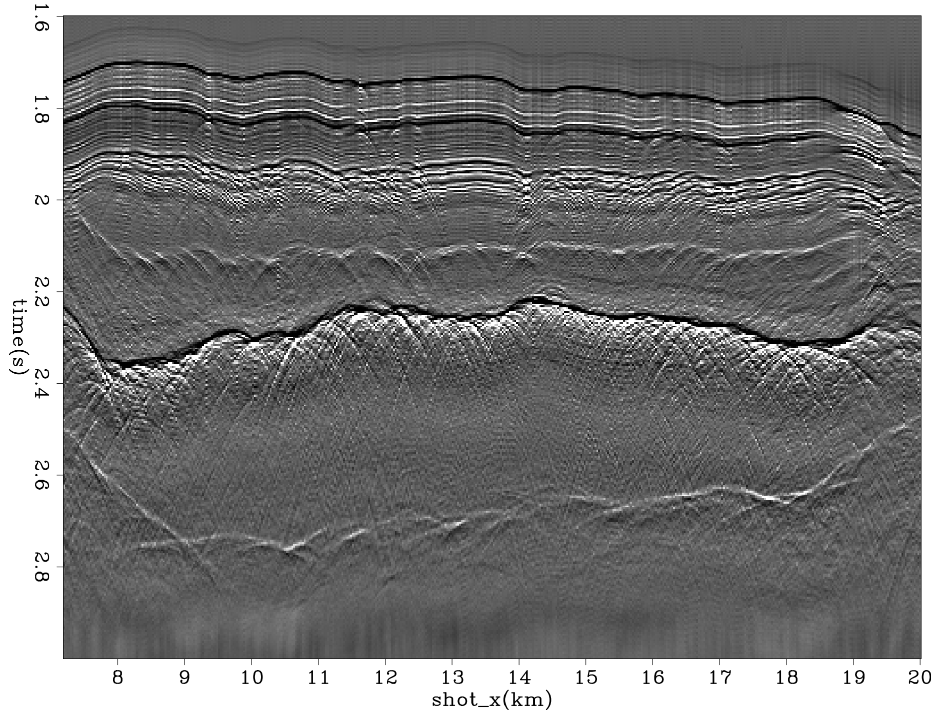
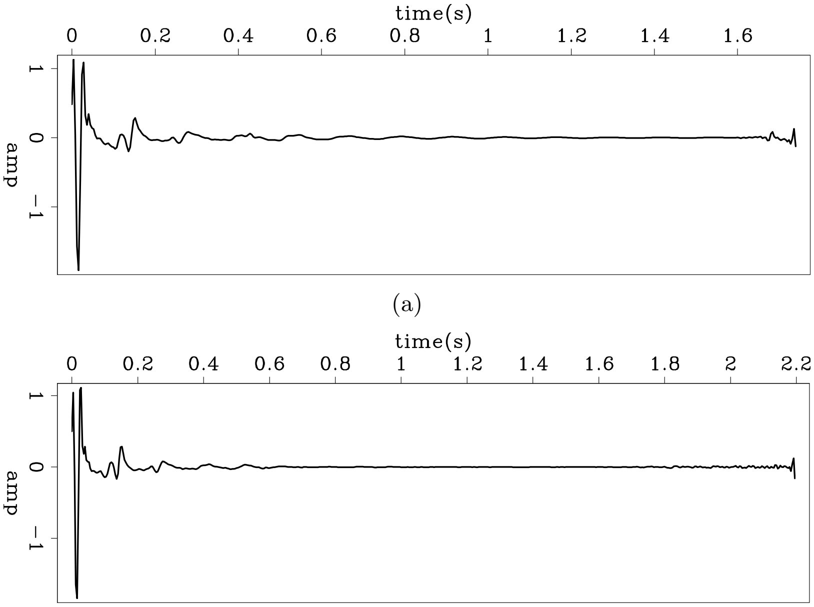


Figure 5: Time-domain (symmetric) bidirectional deconvolution result. [**ER**] qiang/. cof-data-linear



(a)

(b)

Figure 6: Estimated wavelet from (a) logarithm Fourier-domain and (b) time-domain (symmetric) bidirectional deconvolution. The estimation wavelets are the inverse deconvolution filters, calculated by inverting the frequency spectrum of the filter in the Fourier domain. Therefore the wavelet waveform is periodic, and the jitter we find at the end of the wavelet is the anti-causal part of the filter. qiang/. wavelet-cof-data-linear,wavelet-cof-data-log

methods, for both the convergence speed and the quality of the result.

FUTURE WORK

As we mentioned, the 2D marine common offset gather data we used in previous example section has been widely used in judge the result of hyperbolic penalty function based bidirectional deconvolution methods. Because this is the only data set tested so far, conclusions drawn from it are limited. Therefore, we intend to test our methods on other data sets. Next, we discuss some of our preliminary results.

Preliminary Field data pre-stack shot gathers deconvolution result

The common-offset data set we used in previous example section is extracted from a pre-stack survey line. Figure 7 shows the pre-stack shot gathers for the whole survey line. In order to see clearly the deep feature, we use a gain function of t^3 on figure 7. This gain function is only applied here for display purposes here and figure 8 shows two shot gathers in this survey without any gain applied. We also need gain function for deconvolution procedure to ensure the early time and late portion of data have the same weight to contribute to the solution. But We did not figure out what is a suitable gain function of this data set for deconvolution, so in the later test case, we use no gain function in deconvolution procedure.

We find that the previous common-offset gather windowed the data not only in space but also in time. We want to get the data within the same time window as the previous common offset gather. But now we are working on the prestack gathers, and simply cutting a horizontal time window will lose the far-offset part of the deep event, due to the moveout. We do not preform a Normal Move Out correction (NMO) to correct the moveout, because we do not want the stretch caused by NMO to damage the shot waveform. Instead, we perform a constant time shift on each trace and try to flat one target event only. We use a major event in the desired time window, which is the reflection from the top of the salt body, as a target event to calculate the shift time. The shift time fucntion is

$$\tau(x) = t - \sqrt{t_0^2 - x^2/v_0^2}, \tag{9}$$

where x is the offset, v_0 is a constant RMS velocity, t is the travel time at x offset of the target event and t_0 is the travel time at zero offset of the target event. The only unknow in equation 9 is v_0. We can estimated v_0 by picking several points on the target event and do a regression. The Figure 9(a) shows the time-shift function to flatten the shot gather. We apply this time shift and then window the data set by time window 1.6 s to 3 s, figures 9(b) and (c) show two gathers after this window.

After we get the two windowed shot gathers, we apply the Burg PEF as the preconditioner and then perform the logarithm Fourier-domain bidirectional deconvolution on the two windowed shot gathers. Figure 10 shows the preconditioner results, and figure 11 shows the deconvolution results for the two shot gathers. The two shot gathers are processed independently, which means we use different Burg PEFs and different deconvolution filters on the two shot gathers.

The major event (in the vicinity of 2.2 s), which is the top of the salt body, has a phase shift with increasing offset. In figure 11, the near-offset part of this event is black, and then it turns white after an offset of about 1500 meters. The head wave starts at the same offset of 1500 meters. This is not coincidence, but occurs because the reflection has a 90° phase shift after the critical angle.

Figures 12(a) and 12(b) show the estimated wavelets of these two shot gathers. We see that these are quite similar, indicating that the shot waveforms do not change much between these two shots.

We also tried deconvolution of multi-shot gathers with one filter. Figures 13 and 14 show deconvolution results for 39 and 451 shot gathers, respectively, and figures 15(a), 15(b) and 15(c) show a comparison among estimated wavelets of single shot gathers and multi-shot gathers. The wavelets are similar, and from panels (a) to (c) we can see that using more shot gathers reduces jitter. These results tell us that the shot waveforms do not change significantly from shot to shot. This is consistent with our observations in the previous analysis of two shot gathers.

we notice in the figure 15(c), there is low frequency component which looks like a white stripe within the salt body. We do not have the similar thing in our common offset result (figure 4). We expect to get better deconvolution quality for pre-stack gathers due to we have more data, but the results show us the oppsite conclusion. That is a indicate the wavelet is various with offset changing. We need a way to use various wavelet for different

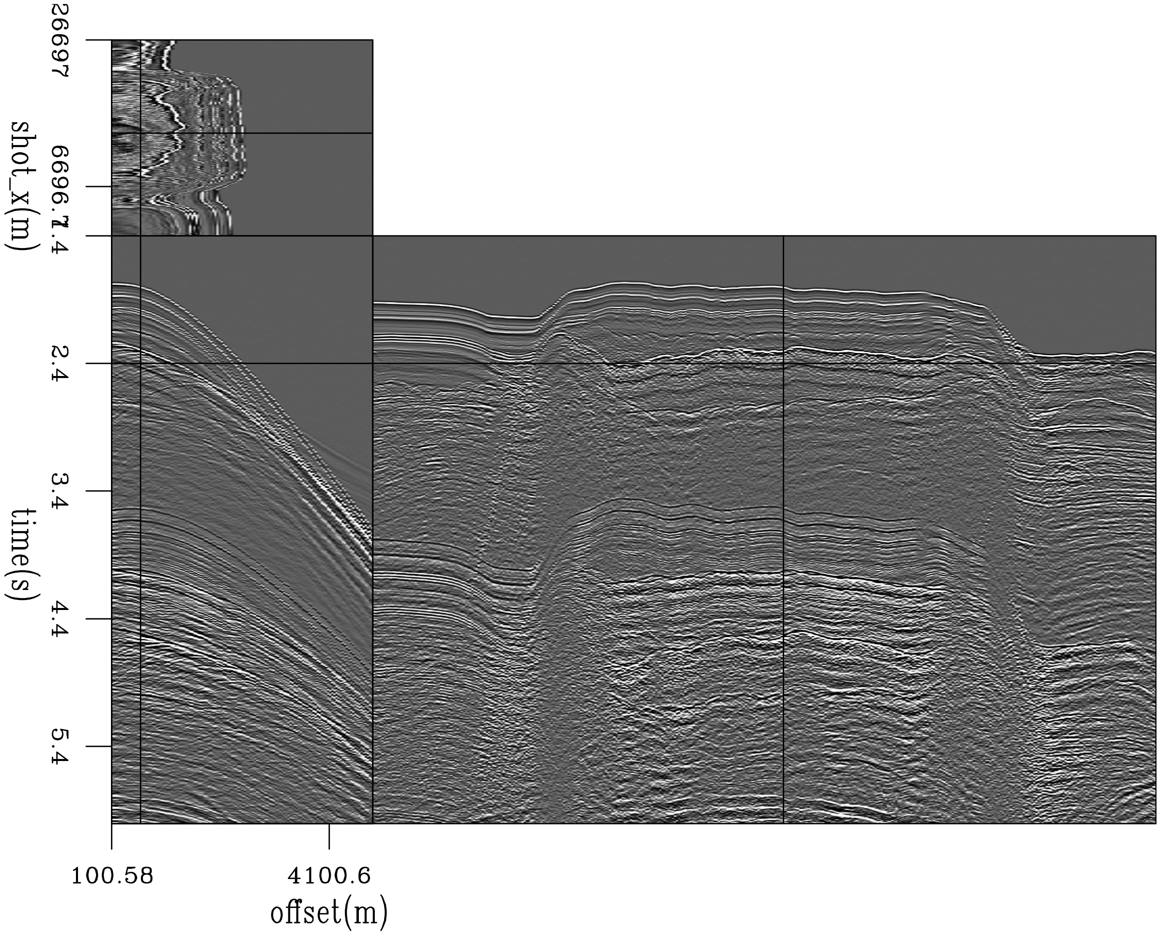
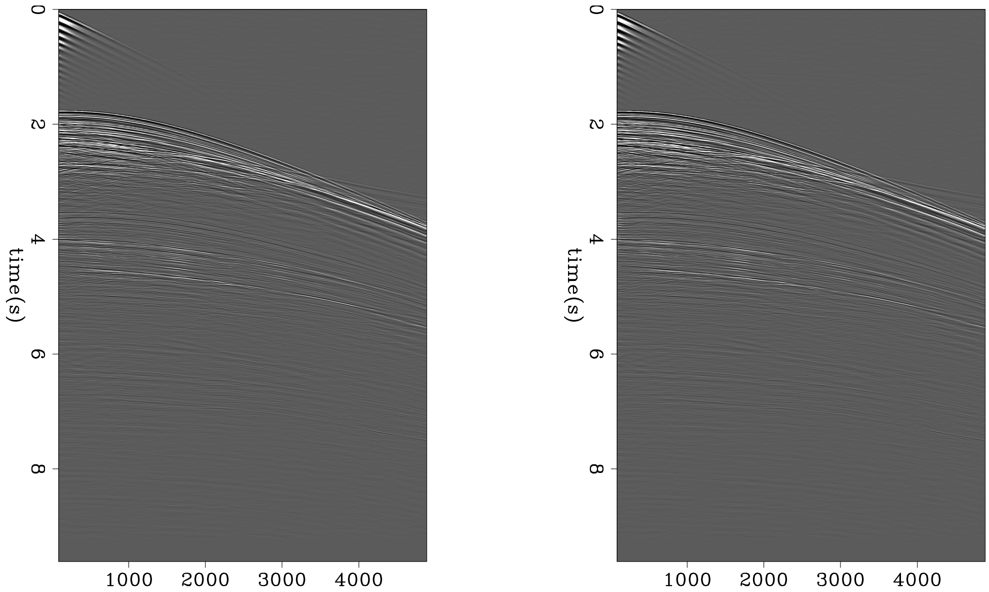


Figure 7: Whole marine pre-stack survey line. [**ER**] qiang/. pre-cube



(a) (b)

Figure 8: Two shot gathers. [**ER**] qiang/. 2-gathers

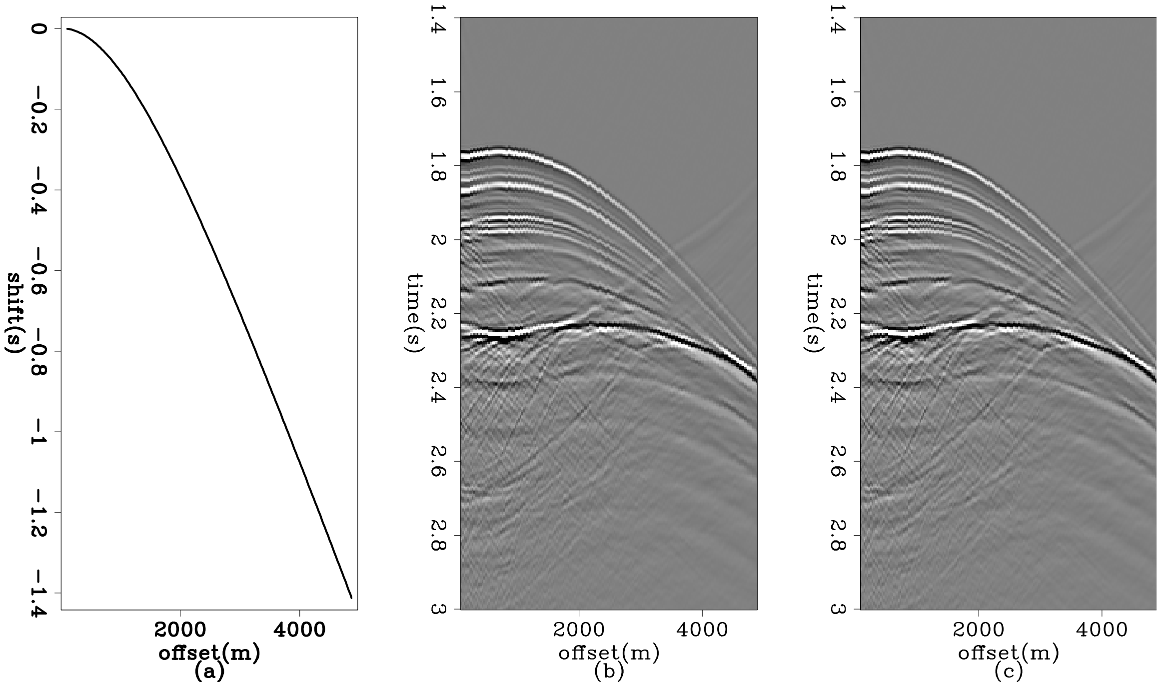


Figure 9: (a)The time shift function to flatten the gather; (b) and (c) two shifted shot gathers. [**ER**] qiang/. shifted-2-gathers

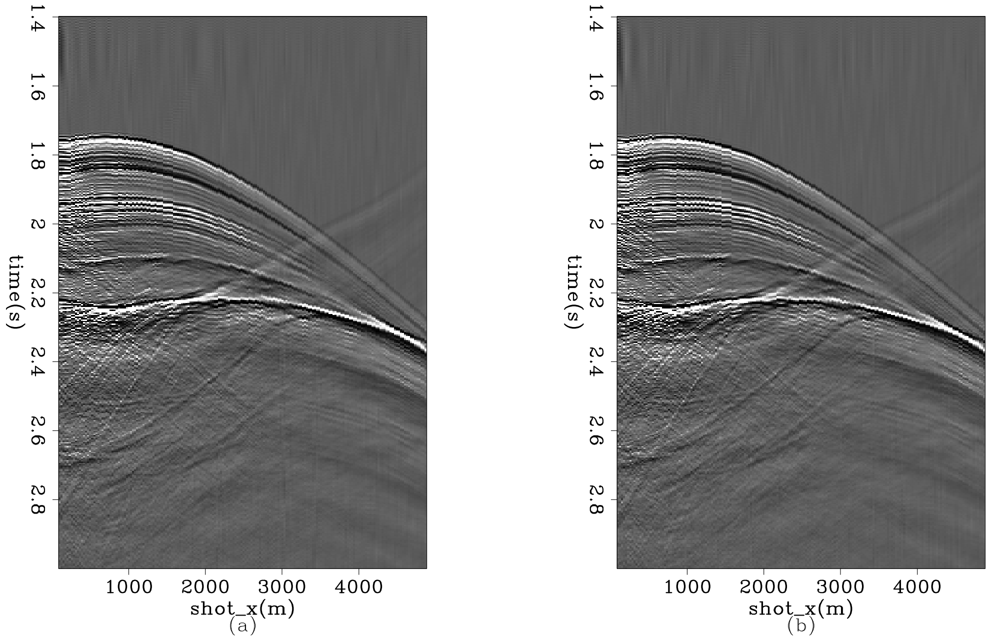
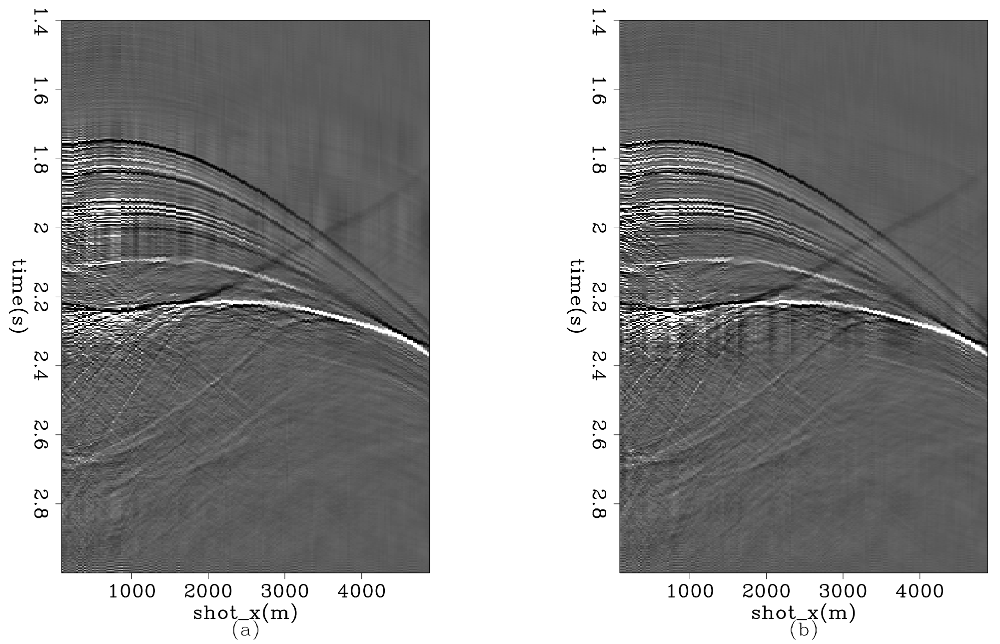
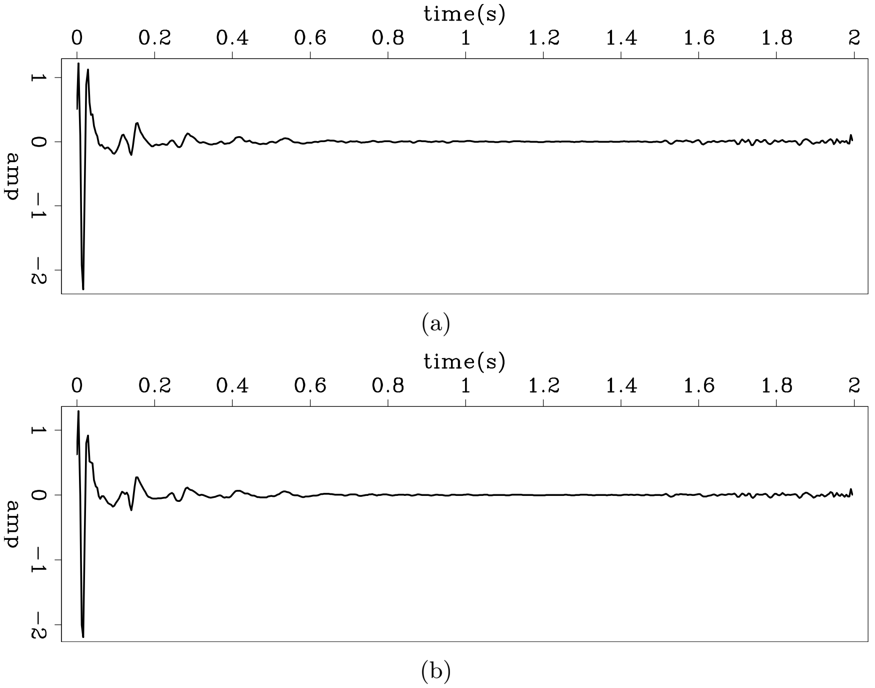


Figure 10: Two shot gathers after applying Burg PEF preconditioning. Different PEF filters are estimated and applied separately. [**ER**] qiang/. shifted-2-decon



(a) (b)

Figure 11: Two shot gathers after logarithm Fourier domain bidirectional deconvolution. The deconvolution is applied separately for each gather and within the same shot gather we use one costant deconvolution filter for all traces [**ER**] qiang/. shifted-2-log-bidecon



(a)

(b)

Figure 12: The estimated wavelets by logarithm Fourier-domain bidirectional deconvolution. (a) Shot gather 14000; (b) shot gather 14028. We notice that the in the Ricker part, (b) is less symmetric than (a), but we do not know exactly what cause this difference. [**ER**] qiang/. wavelet-shot-00,wavelet-shot-28

offset range to handle this problem. But we can not simply divided the shot gathers into several vertical stripes because the wavelet is function of both offset and travel time (and maybe tavelling angle). We need more work to find out the pattern to separate shot gather into portions in which one portion has a relative constant wavelet. Furthermore, we also need to figure out the suitable gain function as the weighting for bidirectional deconvolution before we do more test on this pre-stack data set.

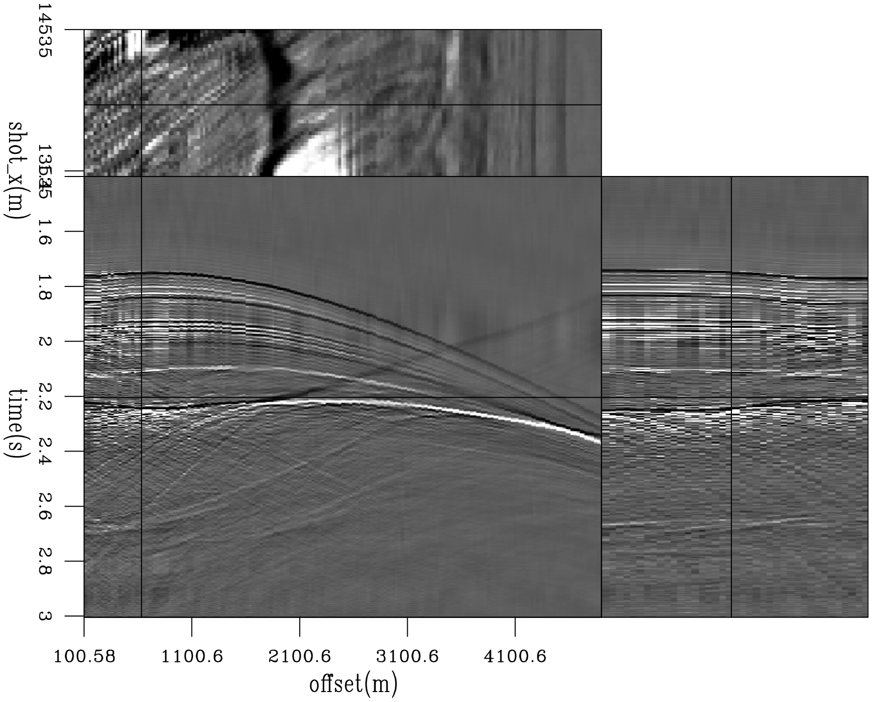


Figure 13: Logarithm Fourier-domain bidirectional deconvolution result on 39 shot gathers. [**ER**] qiang/. 39-bidecon-cube

ACKNOWLEDGMENTS

We would like to thank Yang Zhang, Shuki Ronen, Robert Clapp, Dave Nichols and Antoine Guitton for helpful discussions of our research.

REFERENCES

Claerbout, J., 2009, Blocky models via the l1/l2 hybrid norm: SEP-Report, **139**, 1–10.

Claerbout, J., Q. Fu, and Y. Shen, 2011, A log spectral approach to bidirectional deconvolution: SEP-Report, **143**, 297–300.

Fu, Q., Y. Shen, and J. Claerbout, 2011, An approximation of the inverse ricker wavelet as an initial guess for bidirectional deconvolution: SEP-Report, **143**, 283–296.

Shen, Y., Q. Fu, and J. Claerbout, 2011a, A new algorithm for bidirectional deconvolution: SEP-Report, **143**, 271–282.

———, 2011b, Preconditioning a non-linear problem and its application to bidirectional deconvolution: SEP-Report, **145**, 117–130.

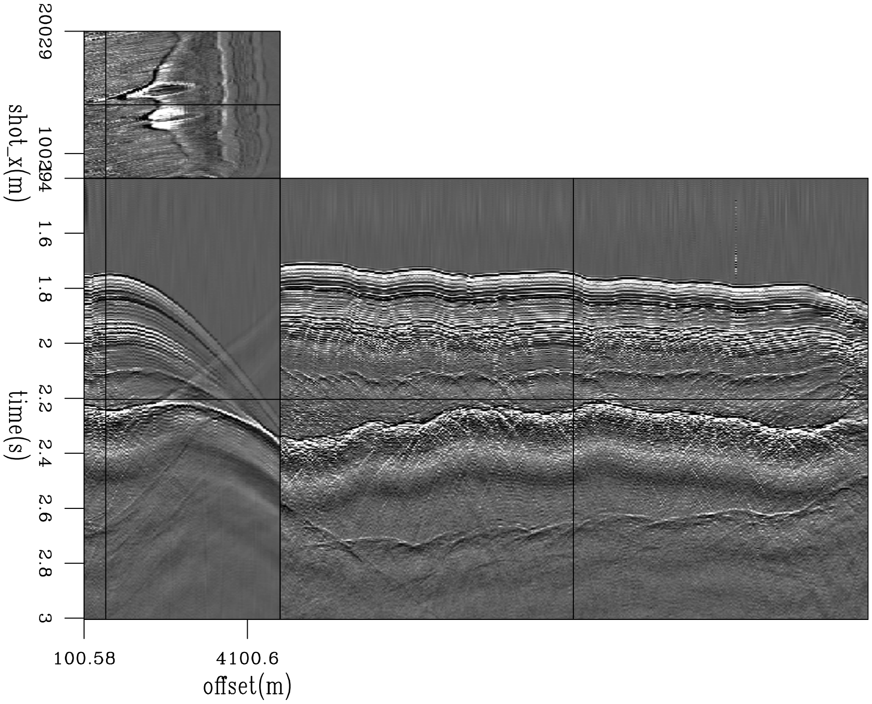
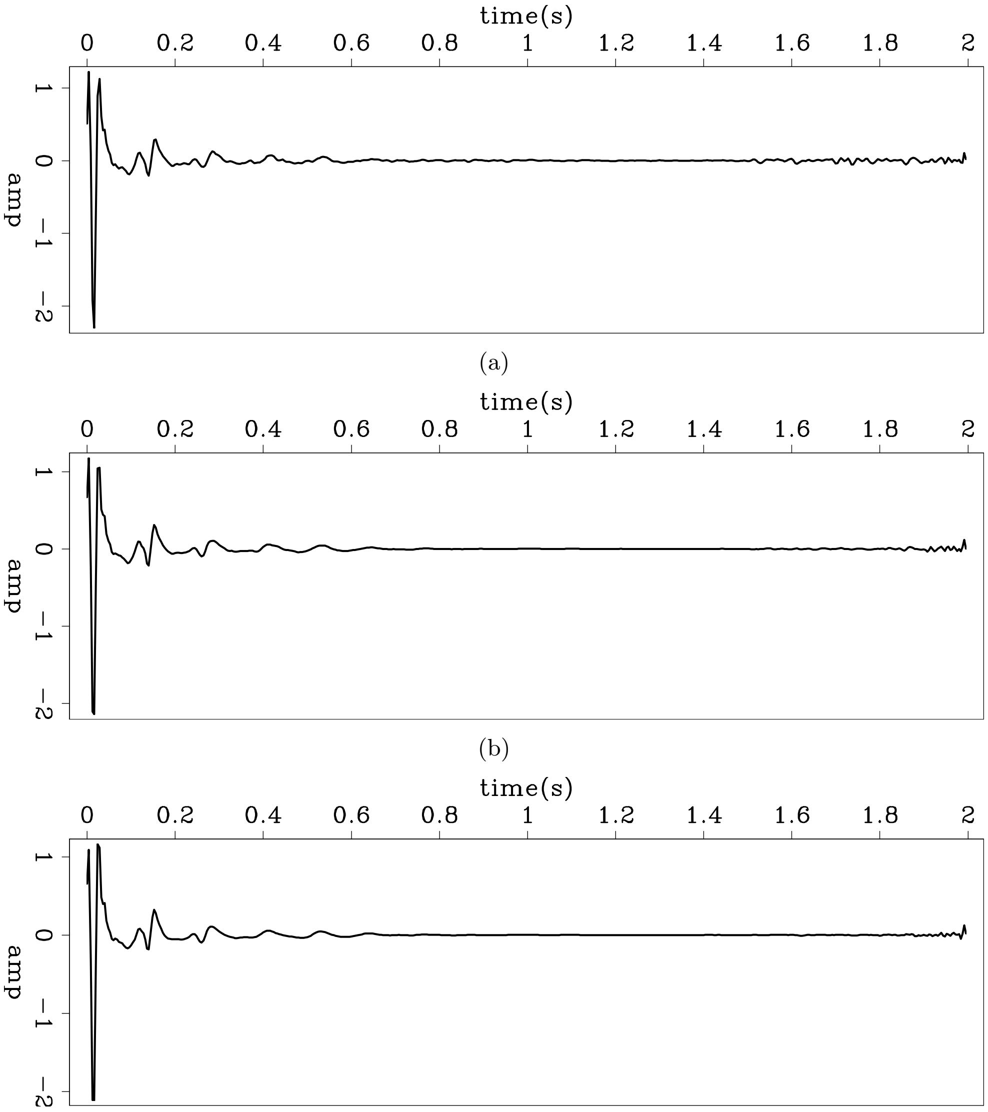


Figure 14: Logarithm Fourier-domain bidirectional deconvolution result on 451 shot gathers. [**CR**] qiang/. 451-bidecon-cube

Zhang, Y. and J. Claerbout, 2010a, Least-squares imaging and deconvolution using the hb norm conjugate-direction solver: SEP-Report, **140**, 129–142.

———, 2010b, A new bidirectional deconvolution method that overcomes the minimum phase assumption: SEP-Report, **142**, 93–104.



(a)

(b)

(c)

Figure 15: The wavelets estimated from (a) one shot gather (same as 12(a)); (b) 39 shot gathers (from shot location of 13500 meters to 14500 meters); (c) 451 shot gathers (from shot location of 8000 meters to 20000 meters). [**ER**] qiang/. wavelet-1-shot,wavelet-39-shots,wavelet-451-shots

Preconditioning a non-linear problem and its application to bidirectional deconvolution

Yi Shen, Qiang Fu and Jon Claerbout

ABSTRACT

Non-linear optimization problems suffer from local minima. When we use gradient-based iterative solvers on these problems, we often find the final solution to be highly dependent on the initial guess. Here we introduce preconditioning and show how it helps resolve these issues in our current problem—bidirectional deconvolution. Using three data examples, we show that results with preconditioning are more spiky than results without preconditioning. Additionally, field data results with preconditioning have fewer precursors, cleaner salt bodies, more symmetric wavelets, and faster convergence than those without preconditioning. In addition to the field data, we illustrate the theory and application of two methods of preconditioning: prediction-error filter (PEF) preconditioning and gapped anti-causal leaky integration followed by PEF (GALI-PEF) preconditioning. Unlike PEF preconditioning, GALI-PEF preconditioning helps constrain the spike to the central wavelet, or allows us to shift it to another position in the wavelet by manipulating the length of the gap.

INTRODUCTION

Least-squares data fitting leads to multivariate linear equations and consequently more theories and techniques than any one person can master in a lifetime. In that field, we are always on well-traveled paths. Problems with non-linear physics are another story: “My program worked great until I increased the model size a little bit.”

Nonlinear optimization problems have many unexpected traps—local minima, as shown in Figure 1. Problems with nonlinear physics require a deeper understanding of the setting than do linear ones. Luckily, there exist helpful techniques that are universally applicable. The first key is to realize that linear equations can be solved with any starting guess, whereas with nonlinear relationships, a sensible starting solution is essential.

Preconditioning is a well established technique used in linear regressions with prior information to hasten convergence. Preconditioning usually begins with regularization and then steers the iterative descent along the path set out by a prior model. However, it does not determine the final result.

The word “gradient” sounds like something fixed in the geometry of the application. Nothing could be further from the truth. Every application offers us a choice of coordinate systems and ways to parameterize the model, and changing the model representation changes the gradient. For example, we could be seeking the earth density as a function of location. We could establish the problem as just that, density as a function of location. On the other hand, we could establish the problem as finding the spatial derivative of the density. The two formulations really seek the same thing, but operators, unknowns, and gradients differ.

Each component of a gradient is independent of the other components and may be scaled arbitrarily as long as its polarity is unchanged. That means that any gradient can be multiplied by any diagonal matrix containing all positive numbers. Additionally, we show in the theory section below that a gradient may be multiplied by any positive definite matrix. That matrix happens to be the model covariance $\mathbf{B}^T\mathbf{B}$, which in local terminology is the inverse of the model styling goal times its adjoint. We may choose any positive definite matrix to modify the gradient. We may even change that matrix from one iteration to the next. What is important is that the matrix is positive definite. At early stages of descent, it is helpful to make the gradient large where confidence is high, and small where it is not. With linear regressions this has no effect on the solution. With nonlinear physics, it steers the solution away from unwelcome local minima.

In image estimation there generally are locations in physical space and in Fourier space in which we have little interest, where we have little expectation that our data contains useful information or that the model will be findable. We need (in nonlinear cases) to be certain such regions do not disturb our descent, especially in early iterations. Therefore, we should view our gradient both in the model space and in the data space, then choose an appropriate diagonal weighting and filter. Given a filter $\mathbf{F}$ and weight $\mathbf{W}$, we apply either $\mathbf{FW}$ or $\mathbf{WF}$ to the gradient. We then apply the matrix transpose, yielding either $(\mathbf{FW})^T(\mathbf{FW})$ or $(\mathbf{WF})^T(\mathbf{WF})$. This procedure destroys no information in the data, but merely selects what aspects of the data are used first. As the final solution is approached, the gradient diminishes; and the down-weighted regions eventually emerge in the gradient, because they are the only things left. Closer to the ultimate solution, it is far less dangerous to have down-weighted regions affecting the solution.

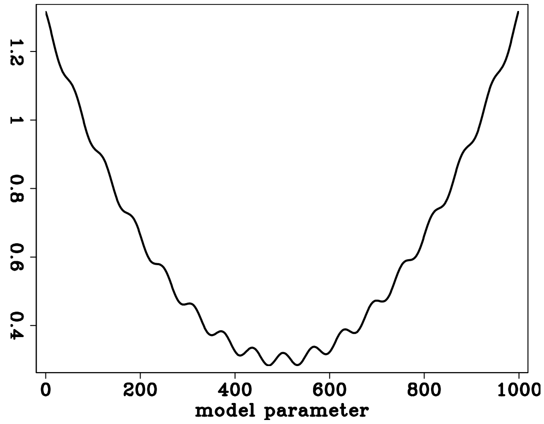


Figure 1: Multiple local minima in the penalty function. [**ER**] yishen/. wigglypar

THEORY

Preconditioning offers smart directions

We start from fitting goals

$$\begin{aligned} \mathbf{0} &\approx \mathbf{Fm} - \mathbf{d} \\ \mathbf{0} &\approx \mathbf{Am} \end{aligned} \tag{1}$$

and change variables from $\mathbf{m}$ to $\mathbf{p}$ using $\mathbf{m} = \mathbf{A}^{-1}\mathbf{p}$:

$$\begin{array}{l} \mathbf{0} \approx \mathbf{r}_d = \mathbf{F}\mathbf{m} - \mathbf{d} = \mathbf{F}\mathbf{A}^{-1}\mathbf{p} - \mathbf{d} \\ \mathbf{0} \approx \mathbf{r}_m = \mathbf{A}\mathbf{m} = \mathbf{I}\mathbf{p} \qquad\qquad\quad . \end{array} \tag{2}$$

Without preconditioning, we have the search direction

$$\Delta\mathbf{m}_{\text{bad}} = \left[\ \mathbf{F}^{\mathrm{T}} \quad \mathbf{A}^{\mathrm{T}}\ \right] \left[\begin{array}{c} \mathbf{r}_d \\ \mathbf{r}_m \end{array} \right], \tag{3}$$

and with preconditioning, we have the search direction

$$\Delta\mathbf{p}_{\text{good}} = \left[\ (\mathbf{F}\mathbf{A}^{-1})^{\mathrm{T}} \quad \mathbf{I}\ \right] \left[\begin{array}{c} \mathbf{r}_d \\ \mathbf{r}_m \end{array} \right]. \tag{4}$$

The essential feature of preconditioning is not that we perform the iterative optimization in terms of the variable $\mathbf{p}$, but that we use a search direction that is a gradient with respect to $\mathbf{p}^{\mathrm{T}}$, not $\mathbf{m}^{\mathrm{T}}$. Using $\mathbf{A}\mathbf{m} = \mathbf{p}$ we have $\mathbf{A}\Delta\mathbf{m} = \Delta\mathbf{p}$. This enables us to define a good search direction in model space:

$$\Delta\mathbf{m}_{\text{good}} = \mathbf{A}^{-1}\Delta\mathbf{p}_{\text{good}} = \mathbf{A}^{-1}(\mathbf{A}^{-1})^{\mathrm{T}}\mathbf{F}^{\mathrm{T}}\mathbf{r}_d + \mathbf{A}^{-1}\mathbf{r}_m. \tag{5}$$

We define the gradient by $\mathbf{g} = \mathbf{F}^{\mathrm{T}}\mathbf{r}_d$ and notice that $\mathbf{r}_m = \mathbf{p}$.

$$\Delta\mathbf{m}_{\text{good}} = \mathbf{A}^{-1}(\mathbf{A}^{-1})^{\mathrm{T}}\mathbf{g} + \mathbf{m}. \tag{6}$$

The search direction (6) shows a positive-definite operator scaling the gradient. All components of any gradient vector are independent of each other and independently point to a direction for descent. Obviously, each can be scaled by any positive number. Now we have shown that we can also scale a gradient vector by a positive definite matrix and still expect the conjugate-direction algorithm to descend, as always, to the "exact" answer in a finite number of steps. This is because modifying the search direction with $\mathbf{A}^{-1}(\mathbf{A}^{-1})^{\mathrm{T}}$ is equivalent to solving a conjugate-gradient problem in $\mathbf{p}$.

Application to Bidirectional Deconvolution

Bidirectional deconvolution (Zhang and Claerbout, 2010; Shen et al., 2011; Claerbout et al., 2011) is a non-linear problem, which has a low convergence rate and unstable result when the starting solution is not close to the true answer. In this section, we apply preconditioning to this problem to obtain a fast and stable result by utilizing prior knowledge. The deconvolution problem is defined as follows:

$$d * a * b^r = \tilde{r}, \tag{7}$$

where d is the data, a and b are the unknown causal filters, and the superscript r denotes the time reverse of filter b. The hybrid norm is applied to $\tilde{r}$, and the reflectivity model is simply $\tilde{r}$ plus a time shift.

We notice that there is only model regularization in this deconvolution problem. Now we change our model from a and b to $\tilde{a}$ and $\tilde{b}$ using $a = p_a * \tilde{a}$ and $b = p_b * \tilde{b}$:

$$d * p_a * p_b^r * \tilde{a} * \tilde{b}^r \approx 0. \tag{8}$$

Thus, we focus on estimating $\tilde{a}$ and $\tilde{b}$ instead of a and b. By applying the prior knowledge in the preconditioners p_a and p_b, we can avoid unwelcome local minima.

GALI-PEF versus PEF preconditioning

In the previous subsections, we showed theoretically that prior knowledge from preconditioners p_a and p_b leads bidirectional deconvolution to the global minimum in the nonlinear problem. We have various choices of preconditioners to indicate different prior knowledge. Here we present two kinds of preconditioning, prediction-error filter(PEF) preconditioning and gapped anti-causal leaky integration followed by PEF (GALI-PEF) preconditioning.

The PEF, whose output is white, is widely used for deconvolution in standard industry practice. The expectation of whiteness in deconvolution encourages us to use PEF as our preconditioner. Thus we choose PEF as the preconditioner p_a and a spike as the preconditioner p_b in PEF preconditioning. Recall that a PEF is a causal filter with a causal inverse. Theoretically, this property adds confidence that deconvolution with a PEF might retrieve the correct phase spectrum as well as the correct amplitude spectrum. However, the wavelet we aim to estimate is not always causal — can be mixed-phase. In most field data —such as band-limited marine seismic data or land response of an accelerometer —the wavelet is similar to a Ricker wavelet. It is dangerous to deal only with the causal part of the data by using PEF, because it may mislead the bidirectional deconvolution to an incorrect phase spectrum and into an unwelcome local minimum.

Therefore, utilizing the prior knowledge of the anti-causal part of the data becomes necessary. A finite representation of the Ricker wavelet is the negative of the second finite difference of some binomial coefficients. In Z-transform representation, this is

$$[(1-1/z)(1-z)][(1+1/z)^N(1+z)^N], \tag{9}$$

where N is the order of the binomial coefficient. In real cases, such as the marine data example, there is a time gap between the first ghost and first arrival; thus the numerical representation of the wavelet becomes

$$[(1-(\rho/z)^g)(1-(\rho z)^g)][(1+1/z)^N(1+z)^N], \tag{10}$$

where g is an integer which indicates the length of the gap, and ρ is a real number which reduces the energy in a trace and deals with the situation where the gap is not an integer. With this numerical representation of the wavelet, we can divide the data by $[(1-(\rho/z)^g)]$ to estimate the anti-causal part of the wavelet. The inverse of $[(1-(\rho/z)^g)]$ is gapped anti-causal leaky integration, which is used as preconditioner p_b. After convolving the data with p_b, we apply a PEF to the convolution result and use this estimated PEF as preconditioner p_a. We hope this GALI-PEF preconditioning leads the bi-directional deconvolution to the correct phase spectrum and makes the result fall into the global minimum.

NUMERICAL EXAMPLE

Bidirectional deconvolution with and without preconditioning

We considered three bidirectional deconvolution methods (Zhang and Claerbout (2010), Shen et al. (2011) and Claerbout et al. (2011)). Of these three methods, the method proposed by Shen et al. (2011) most needs preconditioning. We therefore test our preconditioning on this method to illustrate the effectiveness and limitation of preconditioning.

To illustrate the capabilities of preconditioning, we analyze the results obtained by inverting a zero-phase wavelet. This wavelet is created by convolving the minimum-phase with its own time-reversed wavelet. Figures 2, 3 and 4 show the zero-phase wavelet and its bidirectional deconvolution proposed by Shen et al. (2011), without and with PEF preconditioning. The results show that the wavelet is not completely compressed into a spike without preconditioning, but preconditioning does yield a spike. These results indicate that preconditioning steers the non-linear problem away from unwelcome local minima. However, we can still see slight ringing around the spike in the preconditioned result, indicating that PEF preconditioning does not fully guide the result to the global minimum. This suggests we should introduce more prior knowledge into the preconditioning.

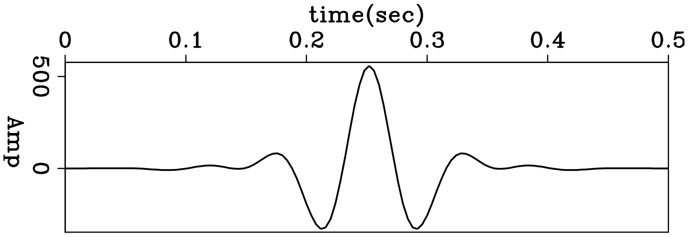


Figure 2: Zero-phase wavelet as the input to the bidirectional deconvolution in Figure 3 and 4. [**ER**] yishen/. data2

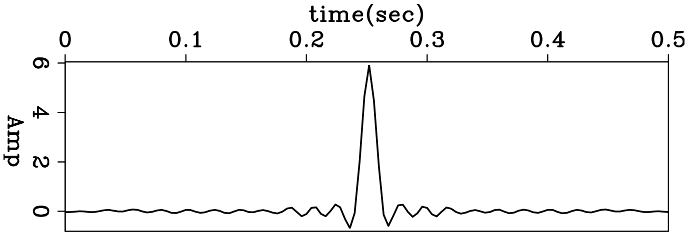


Figure 3: Deconvolution result without preconditioning. The wavelet is not completely compressed into a spike. [**ER**] yishen/. mod2new

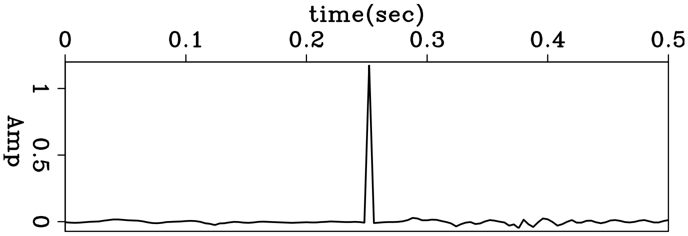
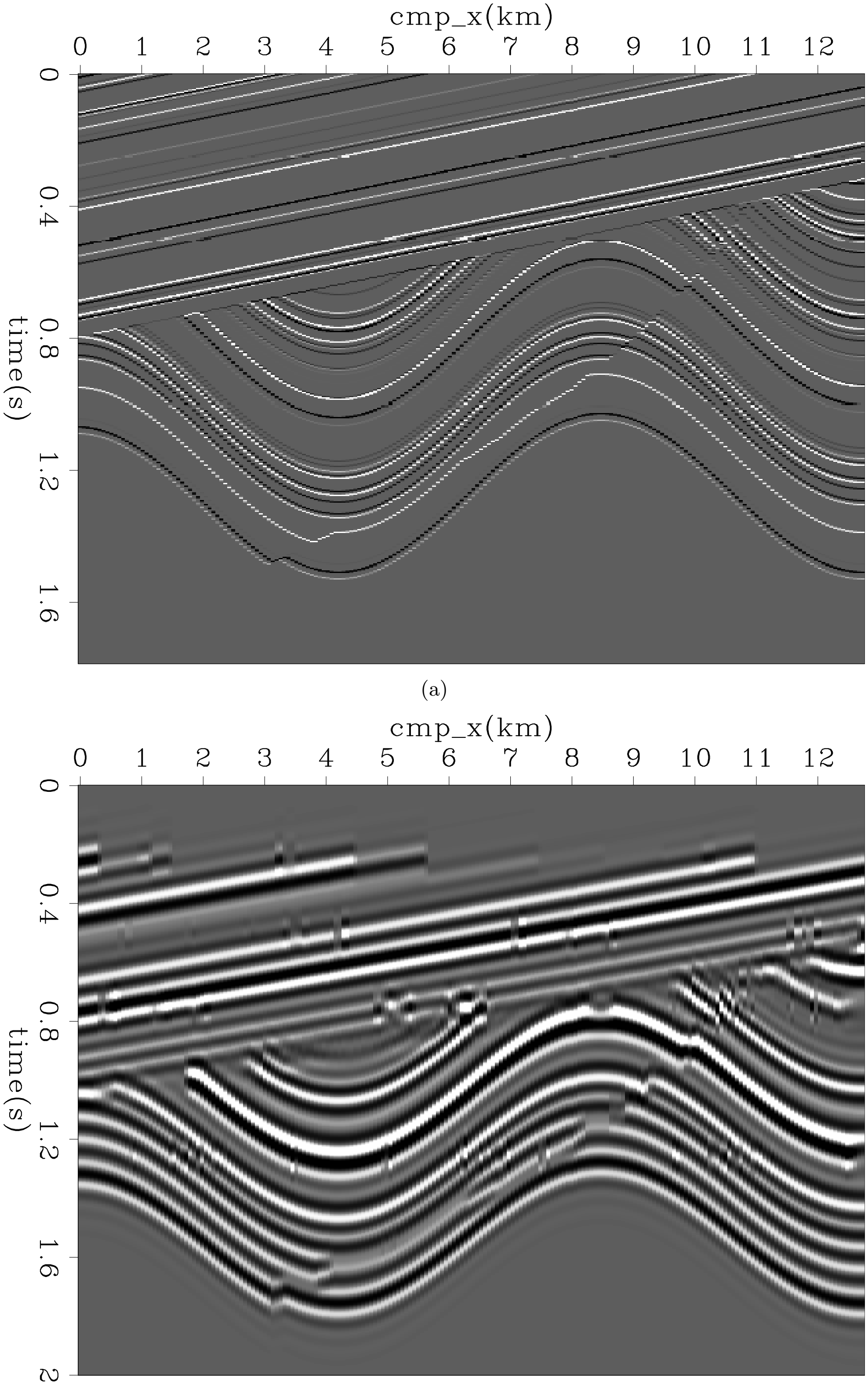


Figure 4: Deconvolution result with PEF preconditioning. The wavelet is compressed into a spike. [**ER**] yishen/. mod2newpre

After deconvolving the simple 1D case, we test preconditioning on more complicated 2D synthetic data. Figure 5(a) shows the starting reflectivity model. Figure 5(b) shows the data generated by convolving the reflectivity model with the zero-phase wavelet in the previous section. All traces in the data share the same wavelet during modeling and deconvolution.

Figures 6(a) and 6(b) show the bidirectional deconvolution proposed by Shen et al.



(a)

(b)

Figure 5: (a) The 2D synthetic reflectivity model; (b) the synthetic data generated using the zero-phase wavelet. [**ER**] yishen/. syn3,data3

(2011) without and with PEF preconditioning. The deconvolution model with PEF preconditioning is more spiky than the one without preconditioning, but it still retains some slight ringing around the events. Recall that results in the 1D example show similar properties,, because the same wavelet is used to generate the data in the two examples.

The last example is a common-offest section of marine field data. Figure 7 shows the input data. Figures 8(a) and 8(b) show the bidirectional deconvolution proposed by Shen et al. (2011) without and with PEF preconditioning. Both methods perform well to retrieve the sparse reflectivity in this field data. However, the preconditioned result has fewer precursors and cleaner events than the one without preconditioning. Another important difference is that around 2.4 seconds, there is an unknown event appearing in Figure 8(a), but it disappears in Figure 8(b). Thus we get a cleaner salt body when we apply preconditioning to this set of field data. The cause of the unknown event is still unidentified, but we have one possible explanation for this event. In this dataset, every trace looks identical, but with a time shift. There are two parallel events between 1.7 sec and 1.8 sec which have almost the same distance for all common midpoints. This phenomenon is unusual and may cause the unknown event, because the distance between the salt top and the unknown event is the same as that between the two parallel events. We hope the unknown event will disappear if we use another data set with more variable traces.

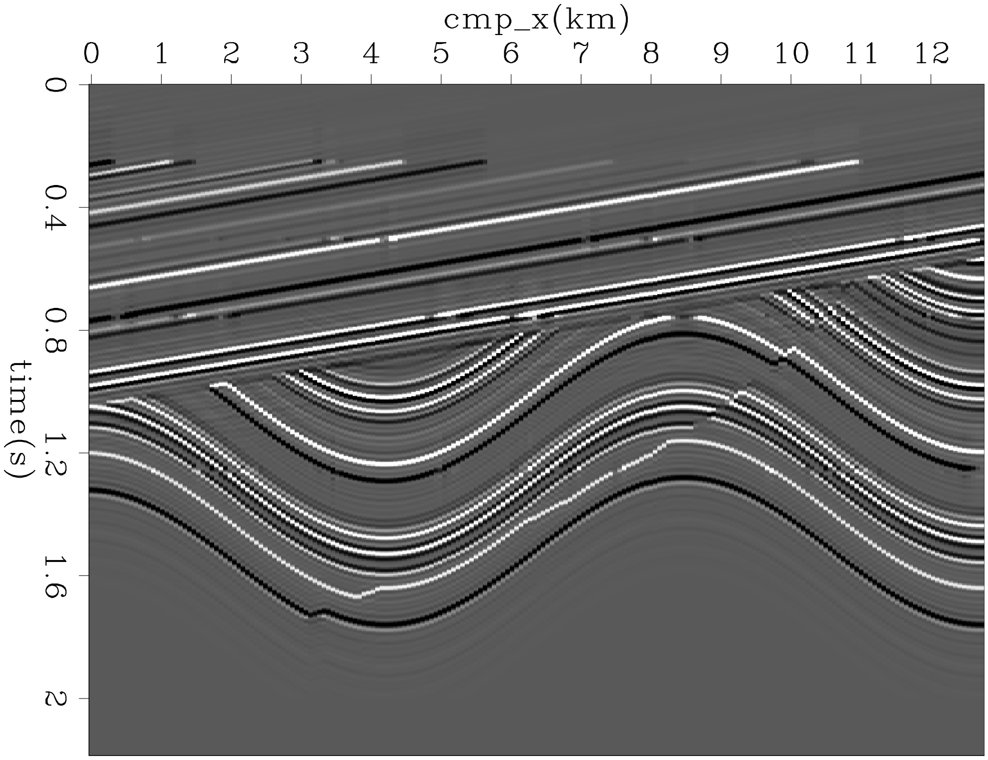
Figures 9(a) and 9(b) show the shot wavelet estimated without and with PEF preconditioning. We notice that both results estimate the bubbles and the double ghost, which can be seen in the data. However, the estimated wavelet with preconditioning is more symmetric than the one without preconditioning. This symmetric quality meets our expectation that the wavelet we invert should look like a Ricker wavelet.

PEF versus GALI-PEF preconditioning

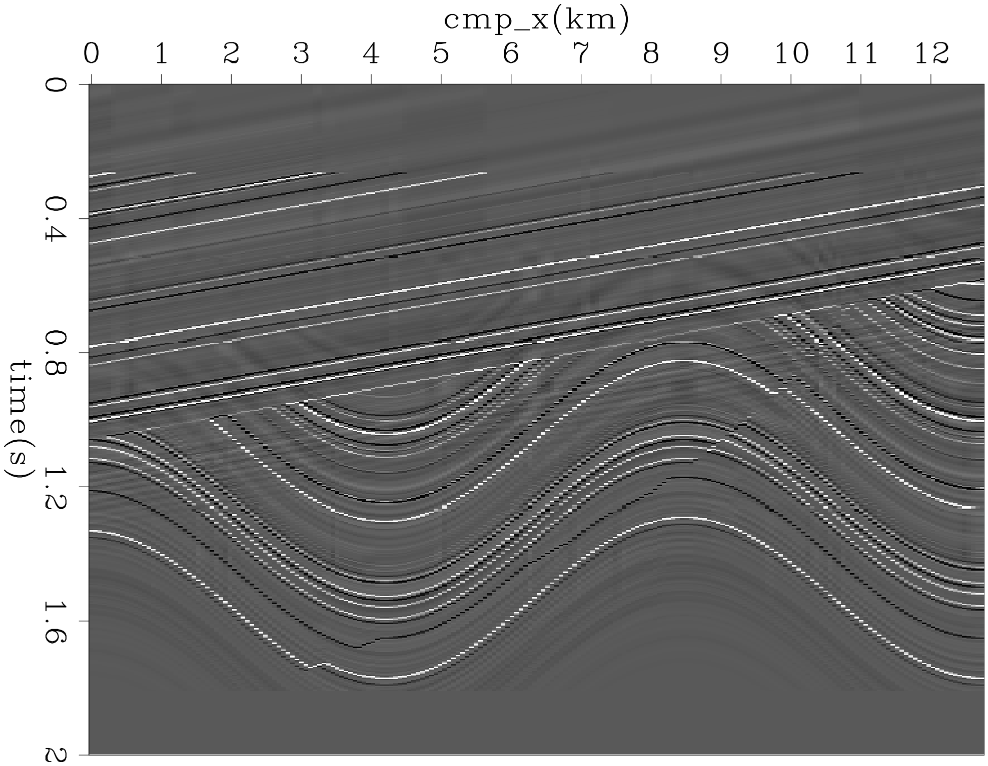
In this subsection, we test the PEF preconditioning and GALI-PEF preconditioning on bidirectional deconvolution. Fu et al. (2011) shows that the method proposed by Claerbout et al. (2011) produces the most stable result among the three bidirectional deconvolution methods considered above. Therefore, we use Claerbout et al. (2011) to compare these two preconditionings to make the comparison reliable.

We use the same field data shown in the previous subsection for this example. First, we convolve the data with PEF and GALI-PEF preconditioning respectively, as shown in Figure 10. Then we apply bidirectional deconvolution to the convolution results, as is displayed in Figure 11. We may draw the following conclusions from the comparison results.

GALI-PEF preconditioning helps constrain the spike to the central wavelet. As the data shows, the events in Figure 7 look like a Ricker wavelet, with two weak side lobes and one strong middle lobe. We expect the preconditioned spike to coincide with the strong middle lobe. Because PEF is a causal filter with causal inverse, it shifts the output toward the first lobe of the Ricker wavelet. Thus the polarity of the output is the same as the first lobe of the Ricker. From panel (b) in Figure 10, the strong event(the water bottom) is black. This polarity, as well as its output location, is the same as that of the first lobe of the mixed-phase wavelet, around 1.8 seconds in Figure 10. Focusing on the first lobe in preconditioning leads to same effect in the bidirectional deconvolution. Panel (b) in Figure 11 shows exactly the same outcome: the output is in the same location and has the



(a)



(b)

Figure 6: Given the 2D synthetic data in Figure 5(b), (a) reflectivity model retrieved without preconditioning; (b) reflectivity model retrieved with PEF preconditioning. [**ER**] yishen/. mod3new,mod3newpre

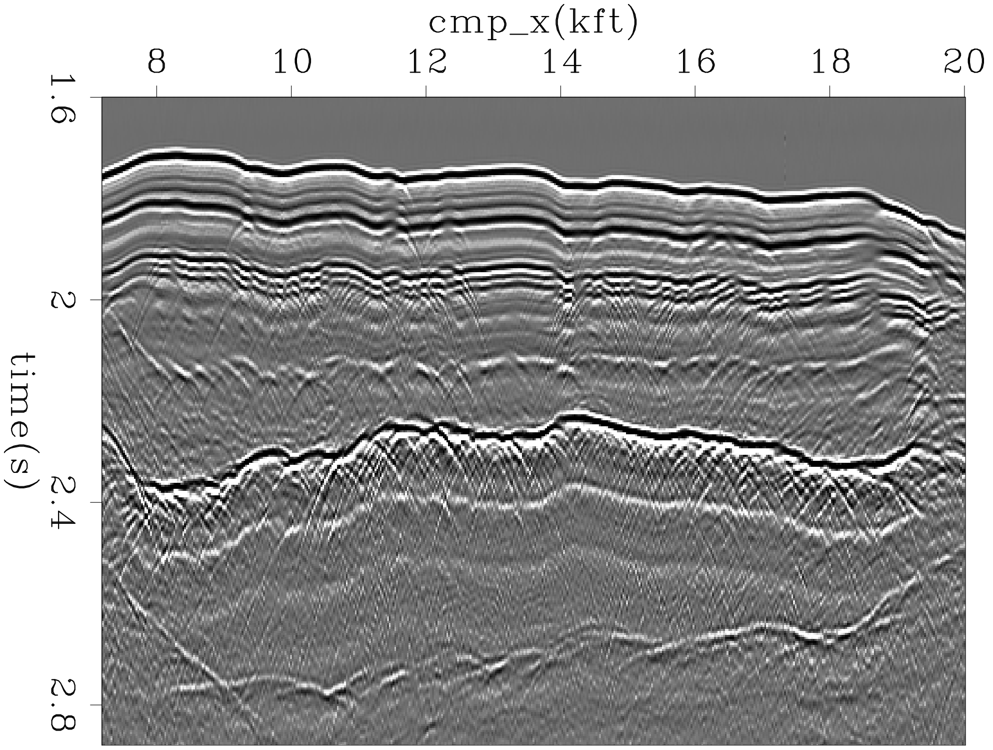


Figure 7: Input Common Offset data. [**ER**] yishen/. data4

same polarity as the first lobe of the Ricker wavelet.

On the other hand, GALI-PEF preconditioning helps shift the time of output. Panel (c) in Figure 10 shows that the event is produced in the same location and polarity as the middle of the Ricker wavelet. The same is true of the bidirectional deconvolution results. To take the water bottom for example, the event appears white in both GALI-PEF preconditioning and its bidirectional deconvolution result, which is the same polarity as the middle lobe of the wavelet. This centered spike is the usual goal of GALI-PEF preconditioning, but by manipulating the length of the gap, we can shift the spike to any desired location. In this case, the gap between the first ghost and first arrival is roughly 10-15 ms. If the gap in GALI-PEF preconditioning is longer than this separation, the output will move towards the second side lobe of the wavelet, and *vice versa*.

Unfortunately, however, GALI-PEF preconditioning does not improve the result compared to PEF preconditioning. Both the PEF and GALI-PEF preconditioning results are almost the same except for reversed polarity and a time shift. In addition, the precursors in Figure 10(c) are strong, because of the anti-causal integration. From another perspective, although the GALI-PEF preconditioner produces a noisier, more resonant section than does PEF, that section illustrates the polarity more clearly than does PEF. Also, the interval between every two adjacent precursors illustrates the gap between first ghost and first arrival.

Both preconditioning methods speed convergence. The convergence rates with and without preconditioning are shown in Figure 12. The average mismatch here is mea-

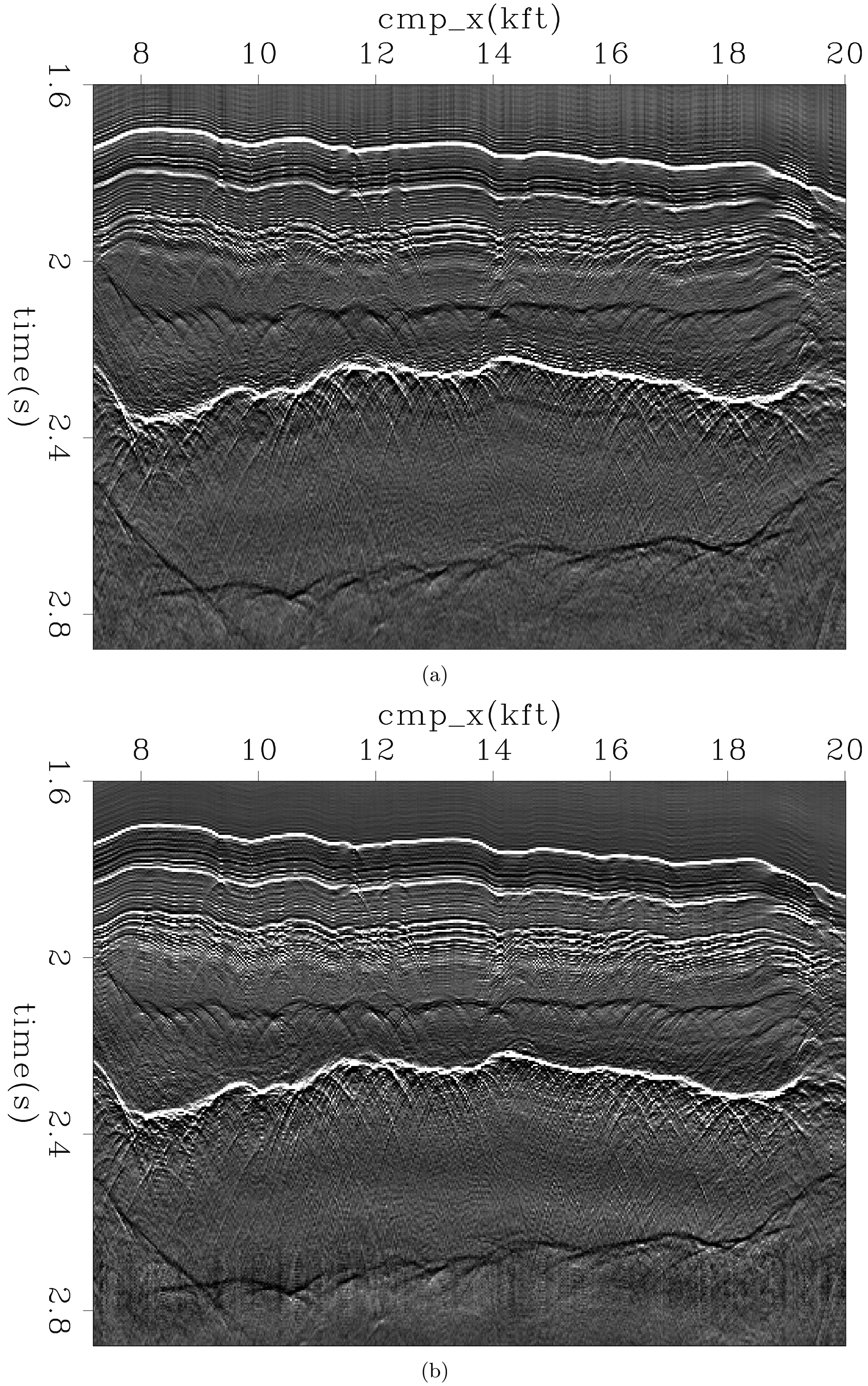
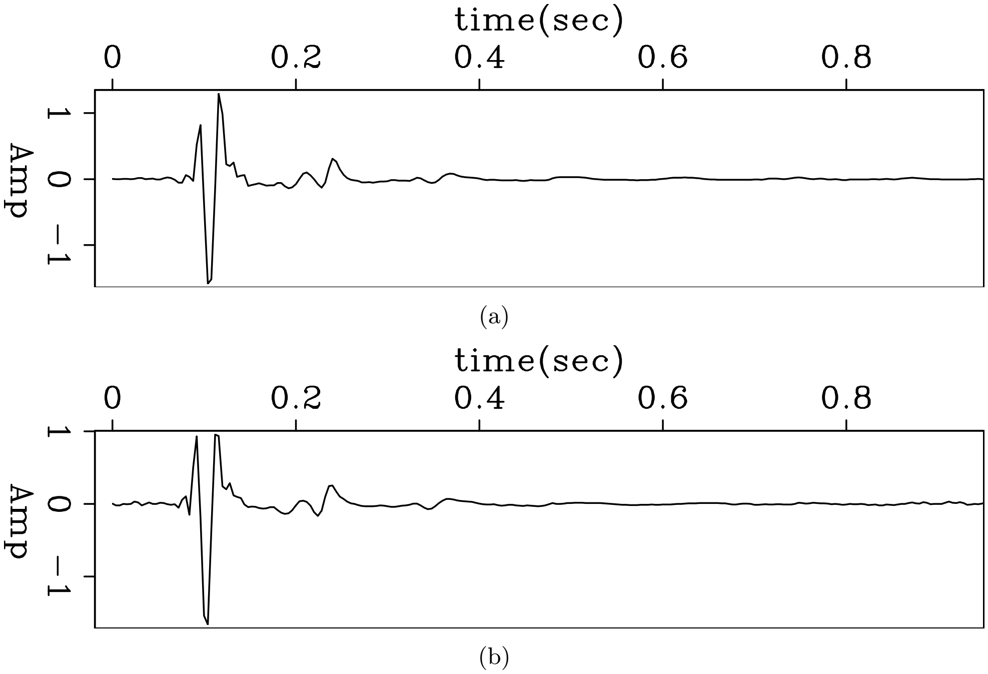


Figure 8: Given the common offset data in Figure 7, (a) reflectivity model retrieved without preconditioning; (b) reflectivity model retrieved with PEF preconditioning. [**ER**] yishen/. mod4new,mod4newpre



(a)

(b)

Figure 9: Given the common offset data in Figure 7, (a) shot wavelet estimated without preconditioning; (b) shot wavelet estimated with PEF preconditioning. [**ER**] yishen/. wavelet4new,wavelet4newpre

sured by using a hybrid penalty function (Claerbout, 2010):

$$\frac{\bar{r}}{R} = r(\bar{H}) = \sqrt{\left(\frac{1}{N}\sum_{i=1}^{N}\sqrt{1+\frac{r_i^2}{R^2}}\right)^2 - 1} \tag{11}$$

where $H(r) = \sqrt{R^2 + r^2} - R$, and R is the threshold. This expression of the misfit is dimensionless and reflects the speed of convergence. Note that the three convergence curves in Figure 12 originate from different points, because the average residual without preconditioning is calculated directly from the raw data, whereas the ones with the two preconditioning methods are calculated from the data transformed by PEF and GALI-PEF preconditionings respectively. Thus, we only consider the relative trend, not the absolute value, of the curves. We notice that the convergence rates drop somewhat with preconditioning, because both PEF and GALI-PEF already help reduce the average mismatch. However, convergence is reached soon after 30 iterations with the help of preconditioning, whereas without preconditioning convergence takes more than 55 iterations. Therefore, preconditioning does reduce the computational cost.

Both methods of preconditioning improve bidirectional deconvolution. The logarithm bidirectional deconvolution proposed by Claerbout et al. (2011), which estimates the filters in the Fourier domain, is more stable than the one proposed by Shen et al. (2011). Thus the result depends less on preconditioning in the logarithm method. However, we still notice that both methods of preconditioning improve the results by reducing precursors. In addition, the unknown event around 2.4 seconds in panel (a) of Figure 11 becomes weaker in the results with preconditioning, especially in bidirectional deconvolution with PEF preconditioning.

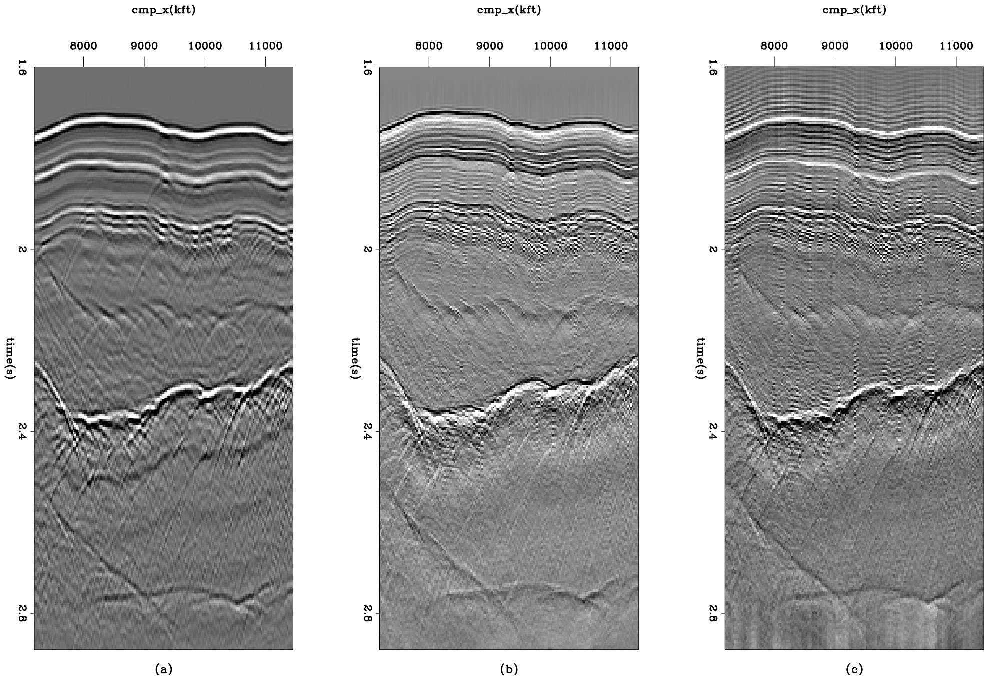
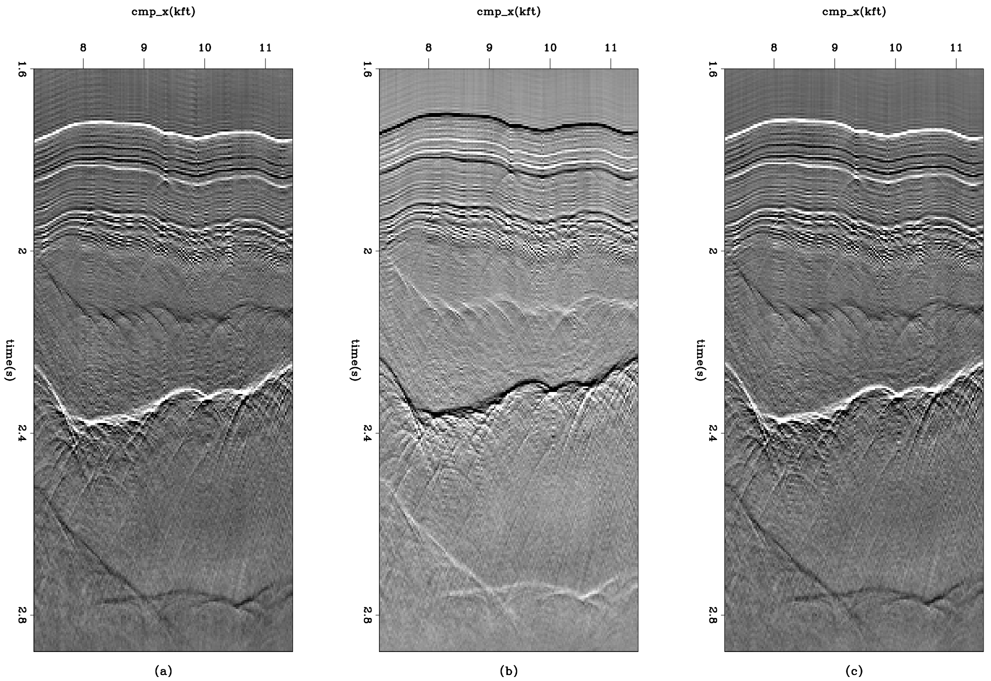


Figure 10: Given the common offset data in Figure 7, (a) 1/3 of original data; (b) data transformed by PEF preconditioning; (c) data transformed by GALI-PEF preconditioning. These three panels are the inputs to the bidirectional deconvolution output in Figure 11. [**ER**] yishen/. dataside

CONCLUSION

In this paper, we illustrate the importance of preconditioning in non-linear problems, and we apply preconditioning to bidirectional deconvolution. The results of three data examples show that wavelets are more spiky in the results with preconditioning than in those without preconditioning. However, the results with preconditioning in the 1D and 2D synthetic sections show slight ringing around the spike, which may encourage us to use more prior knowledge in the preconditioning. For field data, the results with preconditioning have fewer precursors, a cleaner salt body, and a more symmetric wavelet than those without preconditioning. This proves that preconditioning can guide the gradient along sensible pathways, thus avoiding potential local minima, making the results more reliable, and speeding convergence.

In addition, we introduce two methods of preconditioning —PEF and GALI-PEF—and apply them to the field data. Both approaches improve the bidirectional deconvolution result and improve the convergence speed. But unlike PEF preconditioning, GALI-PEF preconditioning helps constrain the spike to the center of the wavelet (or other positions in the wavelet if we change the length of gap). However, we have tested these two methods on only one set of field data. More experiments on other datasets are needed to illustrate the effectiveness and limitations of these two methods of preconditioning in our future work.



(a) (b) (c)

Figure 11: Given the common offset data in Figure 7, (a)bidirectional deconvolution without preconditioning; (b) bidirectional deconvolution with PEF preconditioning (c) bidirectional deconvolution with GALI-PEF preconditioning. [**ER**] yishen/. modelside

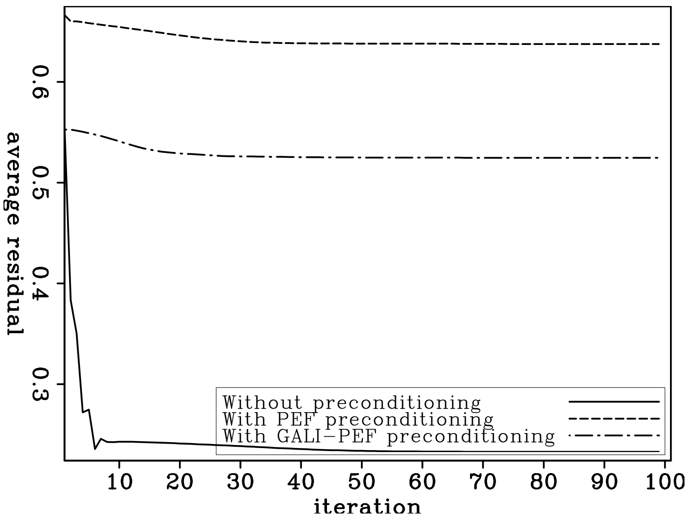


Figure 12: Convergence rate of the results in Figure 11. Both preconditioning methods speed convergence. [**ER**] yishen/. resavg

ACKNOWLEDGMENTS

The authors thank Shuki Ronen for his idea of applying a gap in anti-causal leaky integration, and we thank Dave Nichols, Robert Clapp, Yang Zhang, Antoine Guitton for fruitful discussions.

REFERENCES

Claerbout, J., 2010, Image estimation by example.

Claerbout, J., Q. Fu, and Y. Shen, 2011, A log spectral approach to bidirectional deconvolution: SEP-Report, **143**, 295–298.

Fu, Q., Y. Shen, and J. Claerbout, 2011, Data examples of logarithm fourier-domain bidirectional deconvolution: SEP-Report, **145**, 101–116.

Shen, Y., Q. Fu, and J. Claerbout, 2011, A new algorithm for bidirectional deconvolution: SEP-Report, **143**, 271–281.

Zhang, Y. and J. Claerbout, 2010, A new bidirectional deconvolution method that overcomes the minimum phase assumption: SEP-Report, **142**, 93–103.

SEP PHONE DIRECTORY

Name	Phone	Login Name
Almomin, Ali	723-0463	ali
Ayeni, Gboyega	723-6006	gayeni
Barak, Ohad	723-9282	ohad
Berryman, James	–	berryman
Biondi, Biondo	723-1319	biondo
Claerbout, Jon	723-3717	jon
Clapp, Bob	725-1334	bob
de Ridder, Sjoerd	723-1250	sjoerd
Fu, Qiang	723-1250	qiang
Halpert, Adam	723-6006	adam
Lau, Diane	723-1703	diane
Leader, Christopher	723-0463	chrisl
Li, Elita	723-9282	myfusan
Shen, Xukai	723-0463	xukai
Shen, Yi	723-6006	yishen
Wong, Mandy	723-9282	mandyman
Zhang, Yang	723-1250	yang

Dept fax number: (650) 725-7344

E-MAIL

Our Internet address is "*sep.stanford.edu*"; i.e., send Jon electronic mail with the address "*jon@sep.stanford.edu*".

WORLD-WIDE WEB SERVER INFORMATION

Sponsors who have provided us with their domain names are not prompted for a password when they access from work. If you are a sponsor, and would like to access our restricted area away from work, visit our website and attempt to download the material. You will then fill out a form, and we will send the username/password to your e-mail address at a sponsor company.

STEERING COMMITTEE MEMBERS, 2011-2012

Name	**Company**	**Tel #**	**E-Mail**
Raymond Abma	BP	(281)366-4604	abmar1@bp.com
Biondo Biondi	SEP	(650)723-1319	biondo@sep.stanford.edu
Robert Bloor	ION/GX Technology	(281)781-1141	robert.bloor@iongeo.com
Jon Claerbout	SEP	(650)723-3717	jon@sep.stanford.edu
Thomas Dickens (Co-chair, 2nd year)	ExxonMobil	(713)431-6011	tom.a.dickens@exxonmobil.com
Yi Luo	Saudi Aramco	–	yi.luo@aramco.com
Alejandro Valenciano (Co-chair, 1st year)	PGS	–	alejandro.valenciano@pgs.com

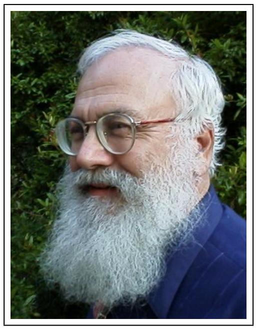
Research Personnel

Ali Almomin graduated from Texas A&M University in 2007 with a BS in Geophysics. Then, he joined Saudi Aramco and worked in several exploration and research departments with a focus on 3D seismic processing and near surface imaging. He joined Stanford Exploration Project in 2009 to pursue a PhD in Geophysics and is currently working on seismic tomography. He is a member of SEG, EAGE, and SPE.

Gboyega Ayeni received his B.Sc. in Applied Geophysics from Obafemi Awolowo University, Nigeria in 2004. He was a Shell sholar at University of Leeds, where he obtained an M.Sc with Distinction in Exploration Geophysics. Gboyega joined SEP in September 2006 to work towards his Ph.D in Geophysics. He is a member of SEG, EAGE, AGU, SPE and AAPG.

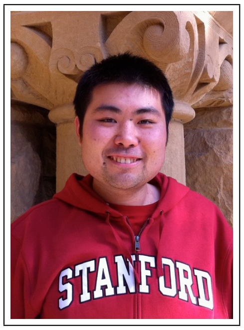
Ohad Barak received a B.Sc. (2006) and an M.Sc. (2009) in Geophysics from Tel-Aviv University. In 2008 he joined Paradigm Geophysical and worked there as a developer. He joined SEP in 2009 and is currently pursuing a Ph.D. in geophysics at Stanford University, and a longer biography.

Biondo L. Biondi is professor of Geophysics at Stanford University. Biondo graduated from Politecnico di Milano in 1984 and received an M.S. (1988) and a Ph.D. (1990) in geophysics from Stanford. He is co-director of the Stanford Exploration Project and of the Stanford Center for Computational Earth and Environmental Science. In 2004 the Society of Exploration Geophysicists (SEG) has honored Biondo with the Reginald Fessenden Award. Biondo recently published a book, 3-D Seismic Imaging, that is the first text book to introduce the theory of seismic imaging from the 3-D perspective. The book is published by SEG in the Investigations in Geophysics series. During 2007 gave a one-day short course in 28 cities around the world as the SEG/EAGE Distinguished Short Course Instructor (DISC) . He is a member of AGU, EAGE, SEG and SIAM.

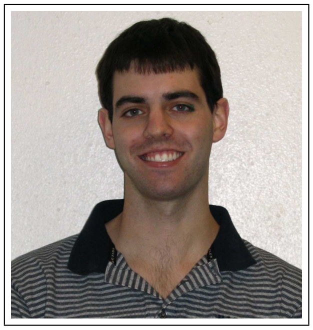


Jon F. Claerbout (M.I.T., B.S. physics, 1960; M.S. 1963; Ph.D. geophysics, 1967), professor at Stanford University, 1967. Emeritus 2008. Best Presentation Award from the Society of Exploration Geophysicists (SEG) for his paper, *Extrapolation of Wave Fields.* Honorary member and SEG Fessenden Award "in recognition of his outstanding and original pioneering work in seismic wave analysis." Founded the Stanford Exploration Project (SEP) in 1973. Elected Fellow of the American Geophysical Union. Authored three published books and five internet books. Elected to the National Academy of Engineering. Maurice Ewing Medal, SEG's highest award. Honorary Member of the European Assn. of Geoscientists & Engineers (EAGE). EAGE's highest recognition, the Erasmus Award.

Robert Clapp received his B.Sc. (Hons.) in Geophysical Engineering from Colorado School of Mines in May 1993. He joined SEP in September 1993, received his Masters in June 1995, and his Ph.D. in December 2000. He is a member of the SEG and AGU.



Qiang Fu received his Bachelors degree in Computer Science from Beijing Institute of technology, China in 2003. From 2007 to 2010 he worked for Saudi Aramco in Saudi Arabia. He joined SEP in the fall of 2010 and is currently working toward a M.S. in Geophysics at Stanford University. He is a student member of the SEG.



Adam Halpert graduated from Rice University in May 2007 with a B.Sc. in Earth Science and a B.A. in Policy Studies. He joined SEP in the summer of 2007, and is currently working toward a Ph.D. in Geophysics. He is a student member of the SEG and AGU.

Yi Shen got her Bachelors degree in Acoustics, from Dept. of Electronic Science and Engineering, Nanjing University, China in 2010. She joined SEP in the fall of 2010, and is currently working toward a Ph.D. in Geophysics at Stanford University. She is a student member of the SEG.

Yaxun Tang received his B.Sc. (Jul. 2003) and M.Sc. (Aug. 2005) in Geophysics from School of Ocean and Earth Science, Tongji University, Shanghai. He joined SEP in 2005 and is currently working towards a Ph.D. in Geophysics at Stanford University. He is a student member of SEG and EAGE.

SEP ARTICLES PUBLISHED OR IN PRESS

Almomin A., 2011, Correlation-based wave-equation migration velocity analysis: SEG Technical Program Expanded Abstracts (Accepted).

Ayeni, G., Y. Tang, and B. Biondi, 2009, Joint preconditioned least-squares inversion of simultaneous source time-lapse seismic data sets: SEG Technical Program Expanded Abstracts, **28**, 3914–3918.

Ayeni, G. and Nasser, M., 2009, Optimized local matching of time-lapse seismic data: A case study from the Gulf of Mexico, SEG Technical Program Expanded Abstracts, **28**, 3939–3943.

Ayeni, G., and Biondi, B., 2009, Joint target-oriented wave-equation inversion of multiple time-lapse seismic data sets, SEG Technical Program Expanded Abstracts, **28**, 3825–3829.

Ayeni, G., Tang, Y., and Biondi, B., 2010, Efficient seismic monitoring of hydrocarbon reservoirs using multiple shooting vessels. In Proceedings of the 2010 Offshore Technology Conference.

Ayeni, G., and Biondi, B., 2010, Continuous reservoir monitoring with asynchronous simultaneous-source seismic data, 72nd Conference & Technical Exbition, EAGE, Extended Abstracts.

Ayeni, G., 2010, Seismic reservoir monitoring with permanent encoded seismic arrays: SEG Technical Program Expanded Abstracts **29**, 4221–4226.

Ayeni G., and Biondi, B., 2010, Target-oriented joint least-squares migration/inversion of time-lapse seismic data sets: Geophysics, **75**, no. 3, R61–R75.

Ayeni, G., 2011, Cyclic 1D matching of time-lapse seismic data sets: A case study of the Norne Field, SEG Technical Program Expanded Abstracts (Submitted).

Ayeni, G., and Biondi, B., 2011, Wave-equation inversion of time-lapse seismic data sets, SEG Technical Program Expanded Abstracts (Submitted).

Ayeni, G., Almomin, A. and Nichols, D., 2011, On the separation of simultaneous-source data by inversion, SEG Technical Program Expanded Abstracts (Submitted).

Guerra, C. and Biondi, B., 2011, Fast 3D velocity updates using the pre-stack exploding reflector model. SEG Technical Program Expanded Abstracts **29**, 4375–4379.

Guerra, C. and Biondi, B., 2010, Fast 3D velocity updates using the pre-stack exploding reflector model. Geophysics (Accepted).

Biondi, B., 2010, Velocity estimation by image focusing analysis: Geophysics, **75**, pp. 49–60.

Biondi, B., 2010, Velocity estimation by image focusing analysis: SEG Technical Program Expanded Abstracts **29**, 4066–4070.

de Ridder, S., Green's function retrieval by iterated coda correlations, Eos, Trans. AGU, 90(52), Fall Meet. Suppl., Abstract S21A–1699.

de Ridder, S., B. Biondi, G. Papanicolaou, 2009, Kinematics of iterative interferometry in a passive seismic experiment: SEG Technical Program Expanded Abstracts **28**, 1622–1626.

de Ridder, S., N. Crook, S. S. Haines and S. T. Ide, 2010, Seismic Investigation of Underground Coal Fires; A Feasibility Study at the Southern Ute Nation Coal Fire Site, Symposium on the Application of Geophysics to Engineering and Environmental Problems **23**, 630–638.

de Ridder and B. Biondi, 2010, Low-frequency passive seismic interferometry for land data: SEG Technical Program Expanded Abstracts **29**, 4041–4046.

Dellinger, J., J. Yu and S. de Ridder, 2010, Virtual-source interferometry of 4C OBC data at Valhall without a low-cut recording filter: SEG low-frequency workshop.

de Ridder, S. and J. Dellinger, 2011, Imaging of Near-Surface Structures at Valhall using Ambient-Seismic-Noise Eikonal Tomography: EAGE Third Passive Seismic Workshop, Athens, Greece, PSP13.

de Ridder, S. and J. Dellinger, 2011, Ambient seismic noise eikonal tomography for near-surface imaging at Valhall. The Leading Edge, **30**, no. 5, pp. 936–942.

de Ridder, S., 2011, Ambient seismic noise tomography at Valhall. SEG Technical Program Expanded Abstracts **30**, 1597–1601.

Halpert, A., Clapp, R.G., and B. Biondi, 2009, Seismic image segmentation with multiple attributes: SEG Technical Program Expanded Abstracts **28**, 3700–3704.

Halpert, A., Clapp, R.G., and B. Biondo, 2010, Speeding up seismic image segmentation: SEG Technical Program Expanded Abstracts **29**, 1276–1280.

Halpert, A., Clapp, R.G., and B. Biondo, 2011, Interpreter guidance for automated seismic image segmentation: 73rd Conference & Exhibition, EAGE, Extended Abstracts (Accepted).

Leader, C., Shen, X., and Clapp, R., 2011, Memory efficient reverse time migration in 3D: Proceedings of Nvidia GPU Technology Conference 2011/2012 (Submitted).

Li, Y., Y. Zhang and J. Claerbout, 2010, Geophysical applications of a novel and robust L1 solver: SEG Technical Program Expanded Abstracts **29**, 3519–3523.

Li, Y., and Biondi, B., 2011, Migration velocity analysis for anisotropic models: SEG Technical Program Expanded Abstracts **30** (Submitted).

Li, Y., Y. Zhang and J. Claerbout, 2011, Hyperbolic estimation of sparse and blocky models: Geophysics (Submitted).

Li, Y., Nicoles, D., Osypov, K., and Bachrach, R., 2011,Anisotropic tomography with rock physics constraints: 73rd Conference & Technical Exbition, EAGE, Extended Abstracts.

Shen, X., 2010, Near-surface velocity estimation by weighted early-arrival waveform inversion: SEG Technical Program Expanded Abstracts **29** (Accepted).

Zhang, Y., Claerbout, J., Guitton, A., 2011, A new bidirectional deconvolution that overcomes the minimum phase assumption: 73rd Conference & Technical Exbition, EAGE, Extended Abstracts.

Tang, Y. and B. Biondi, 2009, Least-squares migration/inversion of blended data: SEG Technical Program Expanded Abstracts **28**, 2859–2863.

Tang, Y., 2009, Target-oriented wave-equation least-squares migration/inversion with phase-encoded Hessian: Geophysics, **74**, WCA95–WCA107.

Tang, Y. and S. Lee, 2010, Efficient Gauss-Newton Hessian for full waveform inversion: 72nd Conference & Technical Exbition, EAGE, Extended Abstracts.

Tang, Y. and B. Biondi, 2010, Target-oriented wavefield tomography using demigrated Born data: SEG Technical Program Expanded Abstracts **29**, 4280–4285.

Tang, Y. and S. Lee, 2010, Preconditioning full waveform inversion with phase-encoded Hessian: SEG Technical Program Expanded Abstracts **29**, 1034–1038.

Tang, Y. and B. Biondi, 2011, Target-oriented wavefield tomography using synthesized Born data: Geophysics (Accepted).

Tang, Y. and B. Biondi, 2011, Subsalt velocity analysis by target-oriented wavefield tomography: A 3-D field-data example: SEG Technical Program Expanded Abstracts (Submitted).

Vyas, M. and Y. Tang, 2010, On the gradient of wave-equation migration velocity analysis: 72nd Conference & Technical Exbition, EAGE, Extended Abstracts.

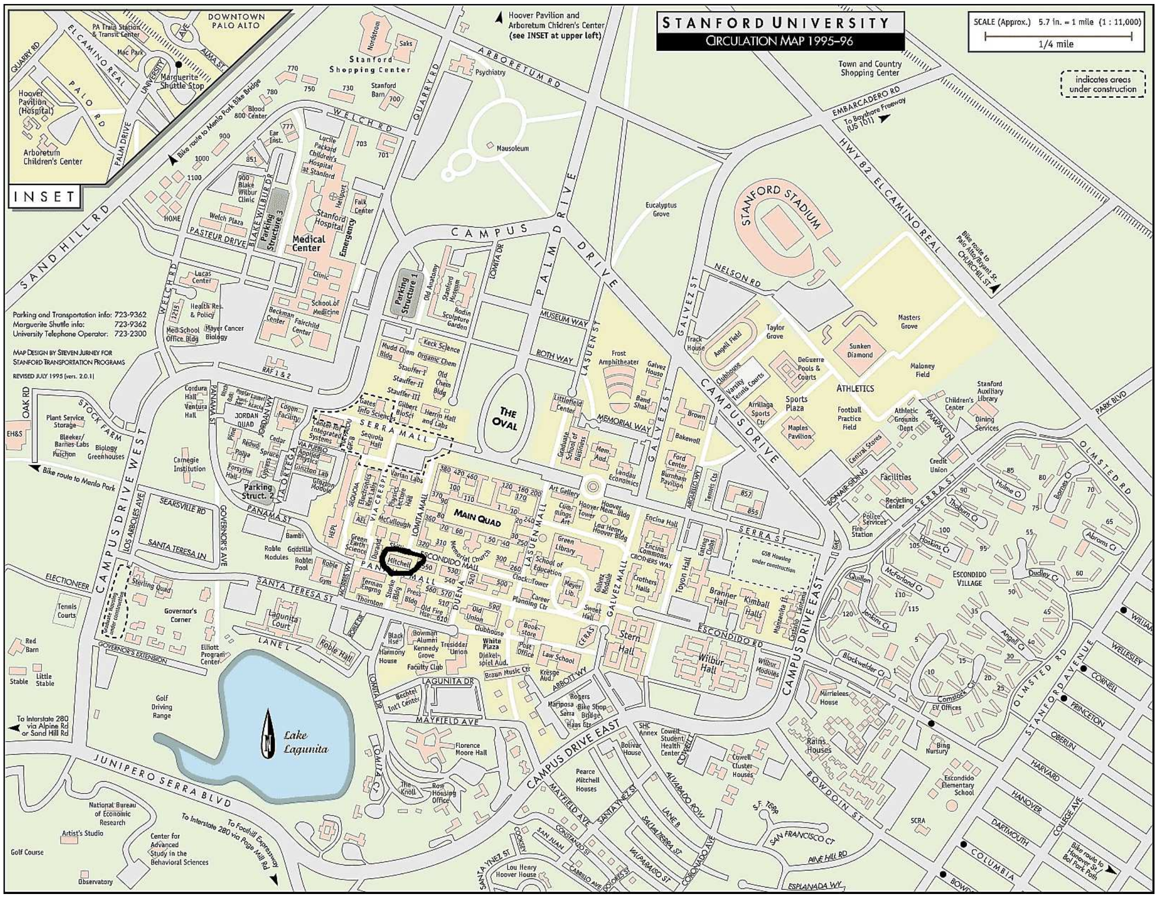
Vyas, M. and Y. Tang, 2010, Gradients for wave-equation migration velocity analysis: SEG Technical Program Expanded Abstracts **29**, 4077–4081.

Wong, M., Biondi, B. L., and Ronen, S., 2010, Joint inversion of up- and down-going signal for ocean bottom data: SEG Technical Program Expanded Abstracts **29**, 2752–2756.

Wong, M. and S. Ronen and B. Biondi , 2010, Least-squares reverse time migration/inversion for ocean-bottom data: A case study (Submitted).

Shen, X., 2010, Near-surface velocity estimation by weighted early-arrival waveform inversion: SEG Technical Program Expanded Abstracts **29**, 1975–1979.

Shen, X. and R. Clapp, 2011, Random boundary condition for low-frequency wave propagation: SEG Technical Program Expanded Abstracts (Submitted).


STANFORD UNIVERSITY
CIRCULATION MAP 1995–96
SCALE (Approx.) 5.7 in. = 1 mile (1 : 11,000)
1/4 mile
indicates areas under construction
Hoover Pavilion and Arboretum Children's Center (see INSET at upper left)
INSET
DOWNTOWN PALO ALTO
PA Train Station & Transit Center
Marguerite Shuttle Stop
Hoover Pavilion (Hospital)
Arboretum Children's Center
EL CAMINO REAL
PALO RD
QUARRY RD
PALM DRIVE
ALMA ST
UNIVERSITY AVE
Parking and Transportation info: 723-9362
Marguerite Shuttle info: 723-9362
University Telephone Operator: 723-2300
MAP DESIGN BY STEVEN JURNEY FOR STANFORD TRANSPORTATION PROGRAMS
REVISED JULY 1995 (vers. 2.0.1)
Stanford Shopping Center
Town and Country Shopping Center
ARBORETUM RD
EMBARCADERO RD
To Bayshore Freeway (US 101)
HWY 82 EL CAMINO REAL
SAND HILL RD
CAMPUS DRIVE
PALM DRIVE
Medical Center
Stanford Hospital
Lucile Packard Children's Hospital at Stanford
Parking Structure 1
Parking Structure 3
Parking Struct. 2
Mausoleum
Eucalyptus Grove
STANFORD STADIUM
NELSON RD
Angell Field
Taylor Grove
Masters Grove
Sunken Diamond
DeGuerre Pools & Courts
Sports Plaza
Maples Pavilion
ATHLETICS
Maloney Field
Football Practice Field
Stanford Auxiliary Library
Children's Center
THE OVAL
Frost Amphitheater
MUSEUM WAY
ROTH WAY
SERRA MALL
MAIN QUAD
Hoover Tower
Memorial Church
Green Library
Mitchell
ESCONDIDO MALL
LOMITA MALL
LASUEN MALL
GALVEZ MALL
SERRA ST
ESCONDIDO RD
Stern Hall
Wilbur Hall
Branner Hall
Kimball Halls
Toyon Hall
Tresidder Union
White Plaza
Bookstore
Post Office
Law School
Old Union
Clock Tower
ESCONDIDO VILLAGE
CAMPUS DRIVE EAST
CAMPUS DRIVE WEST
SANTA TERESA ST
Governor's Corner
Lagunita Court
Roble Hall
Lake Lagunita
Golf Driving Range
JUNIPERO SERRA BLVD
MAYFIELD AVE
Florence Moore Hall
Cowell Student Health Center
Escondido Elementary School
Rains Houses
Red Barn
Stable
Little Stable
To Interstate 280 via Alpine Rd or Sand Hill Rd
Bike route to Menlo Park
Bike route to Menlo Park Bike Bridge
National Bureau of Economic Research
Artist's Studio
Golf Course
Center for Advanced Study in the Behavioral Sciences
Observatory
To Interstate 280 via Page Mill Rd
To Foothill Expressway
Lou Henry Hoover House
OLMSTED RD
STANFORD AVENUE
BOWDOIN ST
COLLEGE AVE
COLUMBIA
DARTMOUTH
HANOVER
HARVARD
OBERLIN
PRINCETON
CORNELL
WELLESLEY
WILLIAMS
PARK BLVD
Bike route to Palo Alto/Bryant St
Bike route to Hanover St / Bol Park Path