STANFORD EXPLORATION PROJECT

Ali Almomin, Gboyega Ayeni, Ohad Barak, Biondo Biondi, Jon Claerbout, Robert Clapp, Antoine Guitton, Chris Leader, Dave Nichols, Yaxun Tang, and Yang Zhang

Report Number 142, October 2010

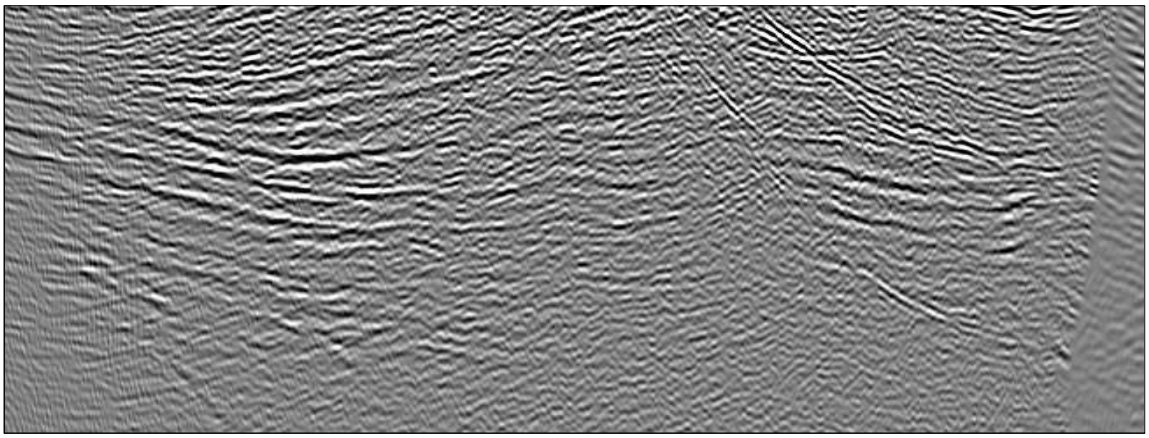


Image before updating the velocity

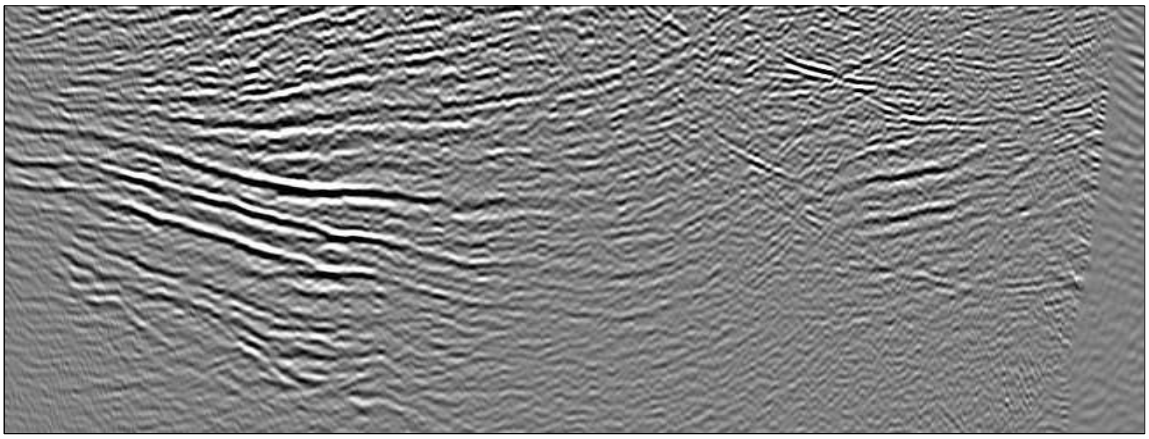


Image after updating the velocity

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/sep142
[2] http://sepwww.stanford.edu/public/docs/sepdatalib/toc_html
[3] http://sepwww.stanford.edu/research/redoc/

SEP142 — TABLE OF CONTENTS

Wave-equation Velocity Analysis

Simultaneous Source

Inversion

Deconvolution

Modeling

Target-oriented wavefield tomography: A field data example

Yaxun Tang and Biondo Biondi

ABSTRACT

We present a strategy for efficient migration velocity analysis in complex geological settings. The proposed strategy contains two main steps: simulating a new data set using an initial unfocused image and performing wavefield-based tomography using this data. We show that the new data set can be synthesized for a specific target region where velocities are inaccurate. We also show that the new data set can be much smaller than the original one due to the target-oriented modeling strategy, but it contains necessary velocity information for successful velocity analysis. These interesting features make this new data set suitable for target-oriented, fast and interactive velocity modeling building. We demonstrate the performance of our method on a selected 2-D line of a 3-D data set acquired from the Gulf of Mexico, where we update the subsalt velocity in a target-oriented fashion and obtain a subsalt image with improved continuities and signal to noise ratio.

INTRODUCTION

Velocity estimation is always a challenging task in exploration seismology. In the past decade, ray-based tomography has been widely used in practice to derive velocity models. Although ray-based methods are efficient, the infinite-frequency approximation and the caustics inherent in ray theory prevent them from accurately modeling complicated wave phenomena (Hoffmann, 2001). As seismic exploration is moving towards structurally complex areas, ray-based methods become less reliable. On the other hand, wave-equation-based tomography (Tarantola, 1984; Mora, 1989; Woodward, 1992; Pratt, 1999; Sava, 2004; Shen, 2004) uses wavefields as carriers of information. It more accurately describes the bandlimited wave phenomena, and therefore more suitable for complex geologies.

Wavefield tomography can be implemented in either data domain or image domain. In this paper, however, we mainly focus on the image-domain wavefield tomography, which is also widely known as wave-equation migration velocity analysis (Sava, 2004; Shen, 2004). It derives an optimum velocity model by driving an objective function defined in the image domain to its minimum (or maximum). Despite its advantages in modeling bandlimited wavefields, practical application of image-domain wavefield tomography is still rare and small in scale due to its huge computational cost (Biondi and Sava, 1999; Shen et al., 2005; Albertin et al., 2006). The high cost arises because of the use of more expensive wavefield modeling engines. The other reason is that it lacks flexibility and the recorded whole data set is usually used for velocity estimation.

Several methods have been proposed to make wavefield tomography more cost effective. The main idea is to reduce the size of the data used for velocity estimation. One method

is to assemble the originally recorded point-source gathers into a smaller number of areal-source gathers. But this strategy lacks flexibility, and full-domain wavefield propagation is still required at each velocity inversion iteration. Therefore, the cost reduction can not be substantial.

Biondi (2006, 2007); Guerra et al. (2009); Guerra and Biondi (2010) approach this problem in a completely different way. They synthesize a new data set based on the initial image using the concept of prestack-exploding-reflector modeling. The new data set is then used specifically for velocity analysis. The advantage of this strategy is that it can model a new data set in a target-oriented fashion, therefore the wavefield propagation can be restricted to regions with velocity inaccuracies, substantially reducing the computational cost. However, the modeling generates crosstalk when multiple image events are modeled simultaneously. This limits the number of reflectors to be modeled. Manual picking and stochastic encoding methods, such as random-phase encoding, are required to minimize the impact of the crosstalk (Guerra et al., 2009).

Another way to synthesize a target-oriented data set is through Born wavefield modeling, or demigration (Tang and Biondi, 2010). This technique has been used by Wang et al. (2005), who generate a post-stack data set and use it for efficient subsalt velocity scan. In our method (Tang and Biondi, 2010), however, we generate a prestack Born data set and use it for wavefield-based tomography. As shown by Tang and Biondi (2010), our modeling strategy is very flexible. Except for windowing out the target image from the initial image, no picking is necessary, but picking can also be introduced if it is desired. Human intervention can also be incorporated by carefully conditioning the initial image to be modeled.

Born wavefield modeling is based on the single-scattering approximation to the full wave equation. The modeled data is obtained by convolving the incident source wavefield, computed using any type of source function (e.g. plane-wave sources), with the initial image and then propagating the convolved wavefields to receiver locations, which can be located anywhere in the model. The target-oriented data set is obtained by only modeling image points within a target zone or several key reflectors that carry important velocity information. This target-oriented velocity analysis strategy is useful, because it allows us to use the most powerful velocity estimation tool to focus on improving velocities in the most challenging areas, e.g., subsalt regions, provided that velocities at other locations are sufficiently accurate, e.g., regions above the salt, where the velocities are usually very accurately determined even by ray-based tomography thanks to the relatively simple geologies.

In the next section, we briefly review the theory of Born modeling. In the subsequent sections, we apply the proposed target-oriented velocity-estimation strategy to a field data set acquired from the Gulf of Mexico.

THEORY

Our method can be formulated under the framework of seismic data mapping (SDM) (Hubral et al., 1996; Bleistein and Jaramillo, 2000), where the idea is to transform the original observed seismic data from one acquisition configuration to another with a designed mapping operator. SDM can be summarized into two main steps as illustrated in Figure 1: (1) apply the (pseudo) inverse of the designed mapping operator to the origi-

nal data set to generate a model; (2) apply the forward mapping operator to the model to generate a new data set with different acquisition configuration than the original one. This idea has been widely used in the area of seismic data interpolation and regularization. For example, in Radon-based interpolation methods (Sacchi and Ulrych, 1995; Trad et al., 2002), Radon operator is used as the mapping operator to regularize the data; the azimuth moveout (AMO) (Biondi et al., 1998) uses dip moveout (DMO) as the mapping operator to transform the data from one azimuth to another.

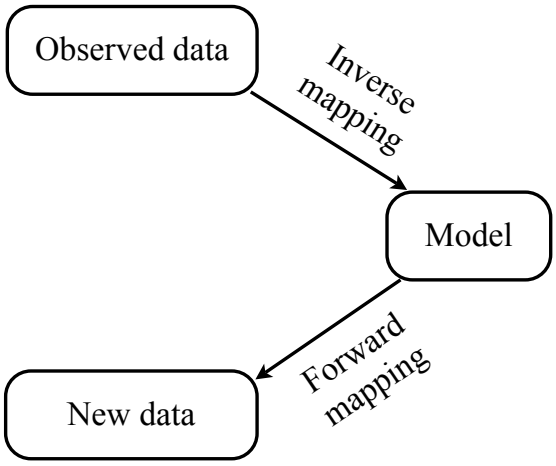


Figure 1: Flow diagrams of seismic data mapping. [**NR**] yaxun1/. sdm

In our method, we use wave-equation-based Born modeling or demigration as the mapping operator to peform data mapping. With an initial velocity model, seismic prestack images can be obtained using the pseudo inverse of the Born modeling operator as follows:

$$\mathbf{m} = \mathbf{H}_0^{\dagger}\mathbf{L}_0^{*}\mathbf{d}_{\text{obs}}, \tag{1}$$

where * and † denote adjoint and pseudo inverse, respectively; $\mathbf{m}$ is the seismic image; $\mathbf{L}_0$ is the Born modeling operator computed using initial velocity $\mathbf{v}_0$, whose adjoint $\mathbf{L}_0^*$ is the well-known depth migration operator; $\mathbf{H}_0$ is the Hessian operator (Plessix and Mulder, 2004; Valenciano, 2008; Tang, 2009); $\mathbf{d}_{\text{obs}}$ is the observed surface data.

It is important to note that the seismic image $\mathbf{m}$ has to be parameterized as a function of both spatial location and some prestack parameter, such as the subsurface offset, reflection angle, etc., in order to preserve the velocity information for later velocity analysis (Tang and Biondi, 2010). In this paper, we use the subsurface offset as our prestack parameter. The significance of the Hessian operator in equation 1 is that its pseudo inverse removes the influence of the original acquisition geometry in the least-squares sense and the resulting image is independent from the original data. However, the full Hessian $\mathbf{H}_0$ is impossible to obtain in practice due to its size and computational cost, we therefore approximate it by a diagonal matrix as follows:

$$\mathbf{H}_0 \approx \text{diag}\{\mathbf{H}_0\}. \tag{2}$$

Substituting equation 2 into equation 1 yields

$$\mathbf{m} = \text{diag}\{\mathbf{H}_0\}^{-1}\mathbf{L}_0^{*}\mathbf{d}_{\text{obs}}. \tag{3}$$

Equation 3 is also widely known as normalized or amplitude-preserving migration (Plessix and Mulder, 2004; Rickett, 2003; Tang, 2009).

We obtain a target image $\mathbf{m}_{\text{target}}$ by applying a selecting operator $\mathbf{S}$ to the initial image as follows:

$$\mathbf{m}_{\text{target}} = \mathbf{S}\mathbf{m}, \tag{4}$$

where the selecting operator $\mathbf{S}$ can be simply a windowing operator. A new data set $\widetilde{\mathbf{d}}_{\text{obs}}$ can then be simulated as follows:

$$\widetilde{\mathbf{d}}_{\text{obs}} = \widetilde{\mathbf{L}}_0 \mathbf{m}_{\text{target}}, \tag{5}$$

where $\widetilde{\mathbf{L}}_0$ is the Born modeling operator computed using the same initial velocity $\mathbf{v}_0$, but with different acquisition configuration. The wavefield propagation can be restricted to regions with inaccurate velocities, and the modeled data can be collected at the top of the target region. The target-oriented modeling strategy makes the new data set much smaller than the original one. The new data set can be imaged using the migration operator, i.e., the adjoint of $\widetilde{\mathbf{L}}$, as follows:

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

We pose our velocity analysis problem as an optimization problem defined in the image domain, where the objective function to minimize is defined as follows:

$$J = ||\mathbf{D}\widetilde{\mathbf{m}}||^2, \tag{7}$$

where $\mathbf{D}$ is the subsurface-offset-domain differential semblance operator (DSO) (Shen, 2004; Shen and Symes, 2008), which is simply a multiplication of the subsurface offset. DSO optimizes the velocity model by penalizing energy at non-zero subsurface offset, utilizing the fact that subsurface-offset gathers should focus at zero subsurface offset if migrated using an accurate velocity model. We evaluate the gradient of equation 7 using the adjoint-state method (Shen and Symes, 2008; Sava and Vlad, 2008; Tang et al., 2008), and use gradient-based methods to optimize the velocity model.

FIELD-DATA EXAMPLES

We test our method on a field data set acquired from the Gulf of Mexico. The data were collected using a narrow azimuth towed streamer (NATS) acquisition system, and further rotated using AMO (Biondi et al., 1998) into zero azimuth. We extracted one crossline from the 3-D data set and performed a 2-D target-oriented wavefield tomography to estimate the subsalt velocity. The extracted data contains 801 shots with the minimum and maximum inline offset equal to 984 ft and 30839 ft. The frequency content ranges from 5 Hz to 35 Hz.

Figure 2 shows the initial velocity model for the extracted 2-D line. Velocities above the target (outlined by a black box) and the salt interpretation are assumed to be accurate. The goal is to invert for subsalt velocities inside the target region. The initial velocities inside the box are set to be $v(z)$. The initial image and subsurface-offset-domain common-image gathers (SODCIGs) obtained using the original data and the initial velocity model is shown in Figure 3. The amplitudes of the initial image have been balanced by the diagonal of the Hessian (Figure 4) according to equation 3 to compensate for uneven subsalt illumination

and remove the influence of the original data acquisition geometry. Note the unfocused SODCIGs due to velocity errors.

Then we use the target image (Figure 3) and the Born modeling described in the previous section to generate 31 plane-wave-source gathers at the top of the target region, where the take-off angle is from $-30°$ to $30°$. The same starting velocity model that was used for migration (Figure 2) has been used for modeling, and the new data set is collected just above the target region. We only model Born wavefields up to 25 Hz. Figure 5 shows the image migrated using the new data set and the initial velocity (Figure 2). Note the same kinematics shown in Figures 3 and 5. This suggests that the velocity information has been successfully preserved using the new data set, which is substantially smaller compared to the original one.

We minimize the objective function J (equation 7) using a nonlinear conjugate gradient solver. Figure 6 shows the inverted velocity model after 30 iterations. We then migrate the original data set using the inverted model and compare the result with that obtained using the initial velocity model. Figures 7, 8 and 9 compare the stacked section (zero-subsurface-offset image) using the initial and updated velocities. The image obtained using the inverted velocity model shows improved continuities, better focusing and higher signal to noise ratio. The angle domain common image gathers (ADCIGs) migrated using the inverted velocity mode (Figure 10(b)) are also flatter comparing to those obtained using the initial velocity model (Figure 10(a)).

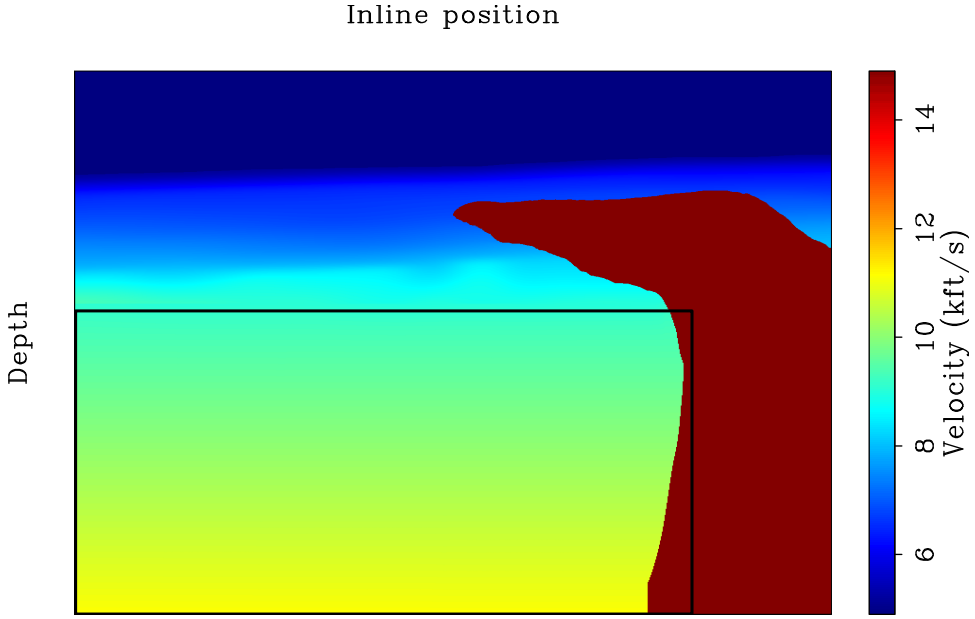


Figure 2: The initial velocity model for the selected 2-D line. The black box outlined area is the target region for velocity analysis. Velocities outside the region are assumed to be accurate. [**NR**] yaxun1/. wemva-bpgom2d-bvel

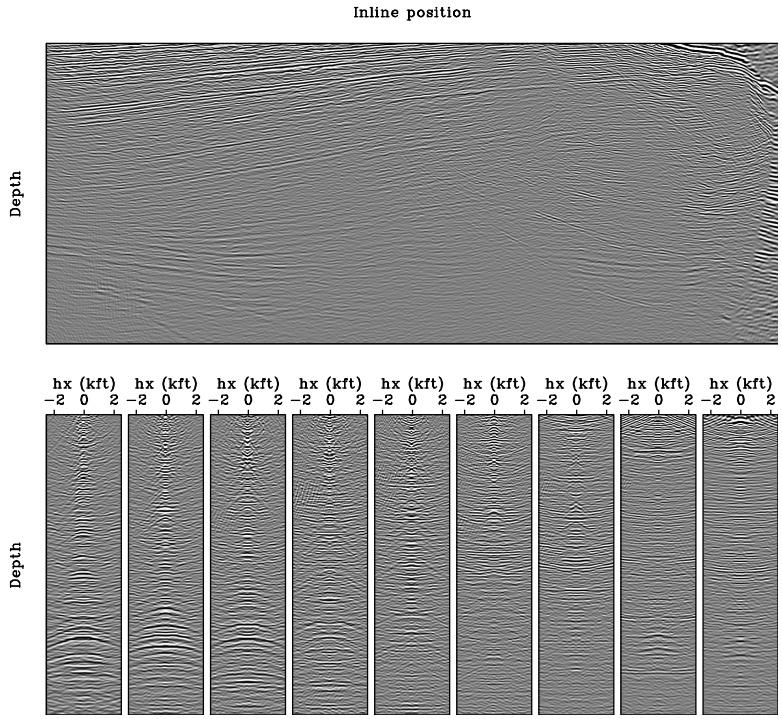


Figure 3: Initial target image and gathers obtained using the original data and the initial velocity shown in Figure 2. Top panel shows the stacked image (zero subsurface offset); bottom panel shows the SODCIGs for different horizontal locations. [**NR**] yaxun1/. wemva-bpgom2d-bimg-target-odcig

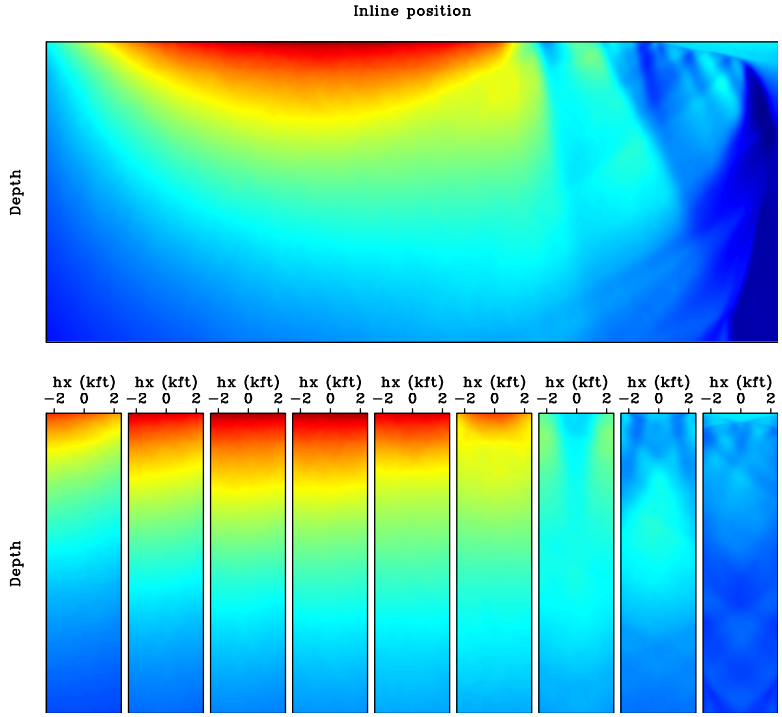


Figure 4: The diagonal of Hessian for the target region. View descriptions are the same as in Figure 3. [**NR**] yaxun1/. wemva-bpgom2d-bhes-target-odcig

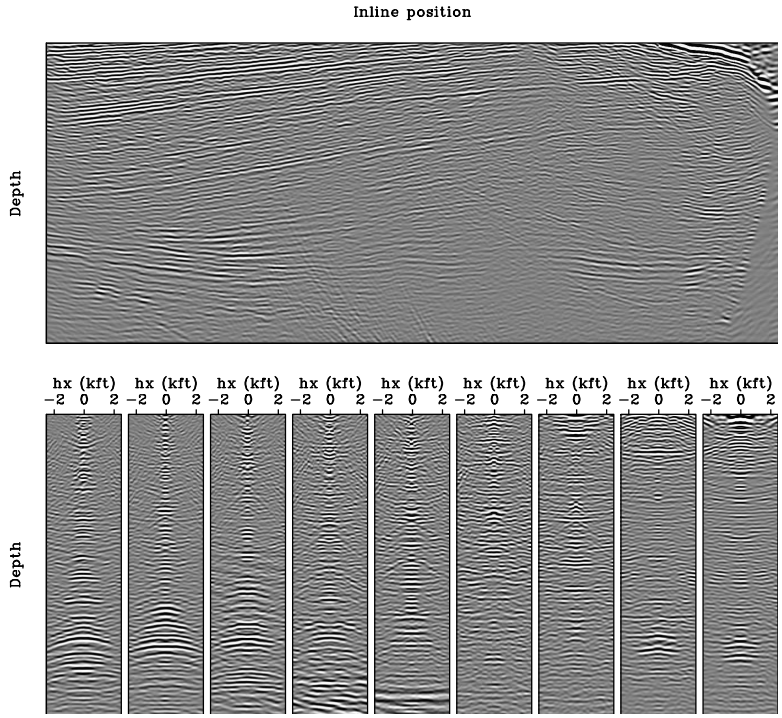


Figure 5: Image and gathers obtained using the new data set and the initial velocity (Figure 2). View descriptions are the same as in Figure 3. [**NR**] yaxun1/. wemva-bpgom2d-bimg-odcig-born-planes

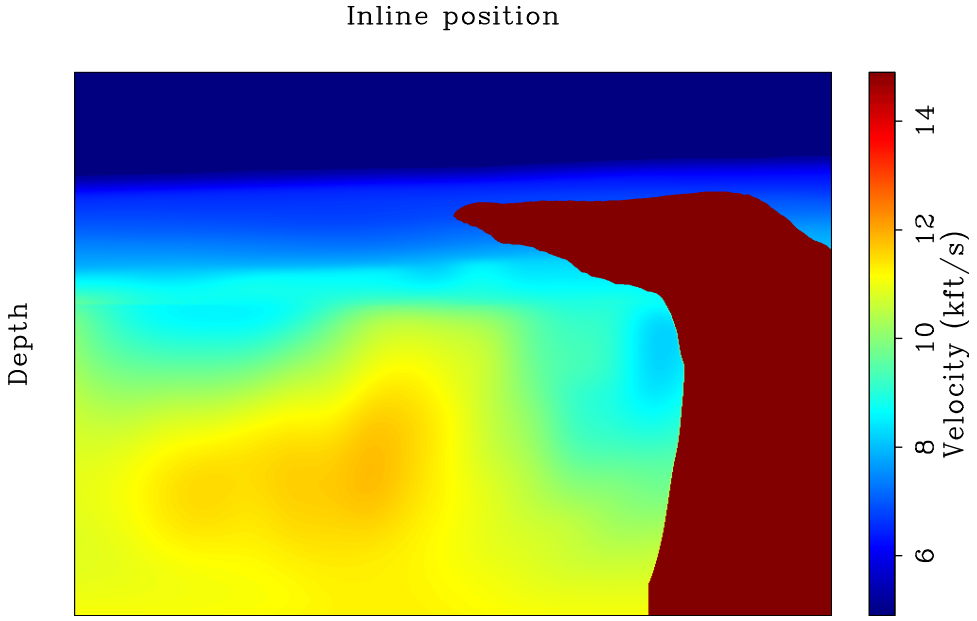


Figure 6: The inverted velocity model after 30 nonlinear iterations. [**NR**] yaxun1/. wemva-bpgom2d-invt-vmod

CONCLUSIONS

We have presented a cost-effective method for image-domain wavefield tomography. Instead of using the original data set for velocity estimation, our method uses demigrated Born data, which can be simulated in a target-oriented fashion and hence much smaller in size. Field-data examples demonstrate that the simulated new data set can successfully preserve velocity information that is useful for velocity analysis and can be used for velocity inversion with low computational cost.

ACKNOWLEDGMENTS

We thank BP and ExxonMobil for making the field data set available.

REFERENCES

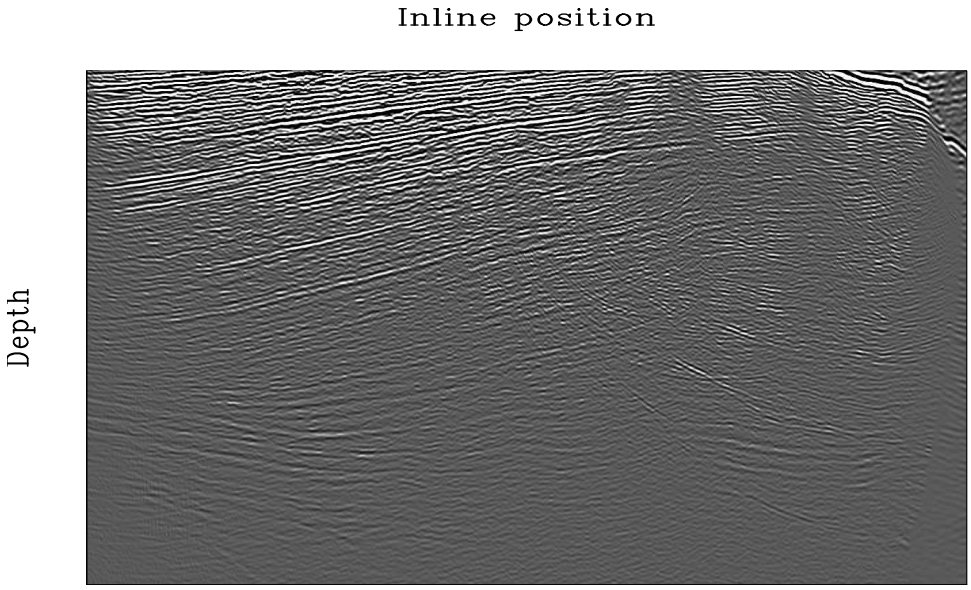
Albertin, U., P. Sava, J. Etgen, and M. Maharramov, 2006, Adjoint wave-equation velocity analysis: SEG Technical Program Expanded Abstracts, **25**, 3345–3349.

Biondi, B., 2006, Prestack exploding-reflectors modeling for migration velocity analysis: SEG Technical Program Expanded Abstracts, **25**, 3056–3060.

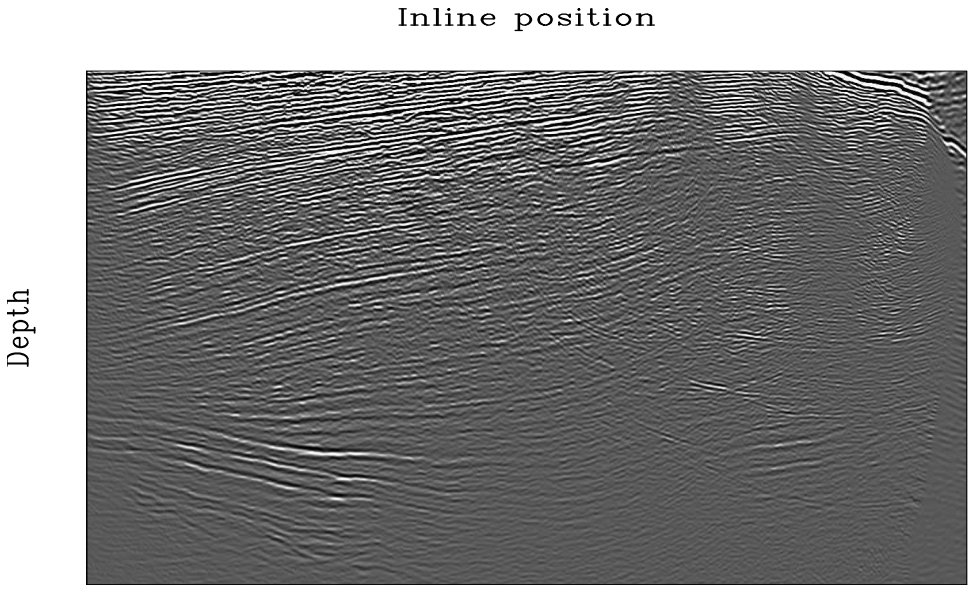
———, 2007, Prestack modeling of image events for migration velocity analysis: **SEP-131**, 101–118.

Biondi, B., S. Fomel, and N. Chemingui, 1998, Azimuth moveout for 3-d prestack imaging: Geophysics, **63**, 574–588.

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



(a)



(b)

Figure 7: Stacked images (zero-subsurface-offset images) obtained using (a) the initial velocity model and (b) the inverted velocity model. The original data set is used for migration. [**NR**] yaxun1/. wemva-bpgom2d-bimg-stack-raw-slides,wemva-bpgom2d-invt-stack-raw-slides

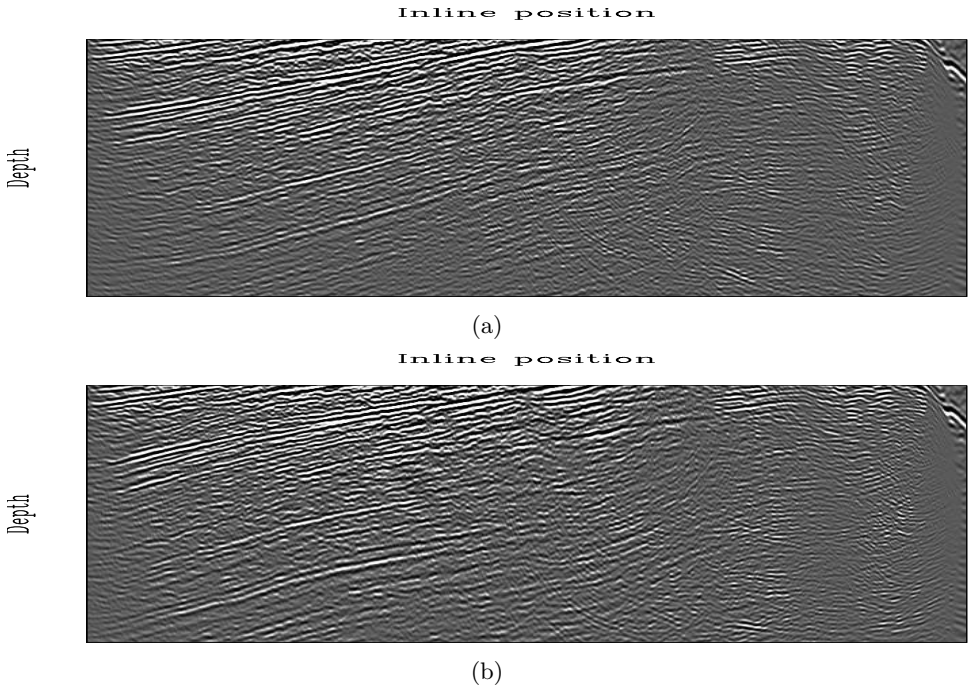


Figure 8: A close-up view of the upper section in Figure 7. (a) is obtained using the initial velocity model and (b) is obtained using the inverted velocity model. [**NR**] yaxun1/. wemva-bpgom2d-bimg-stack-raw-zoom1,wemva-bpgom2d-invt-stack-raw-zoom1

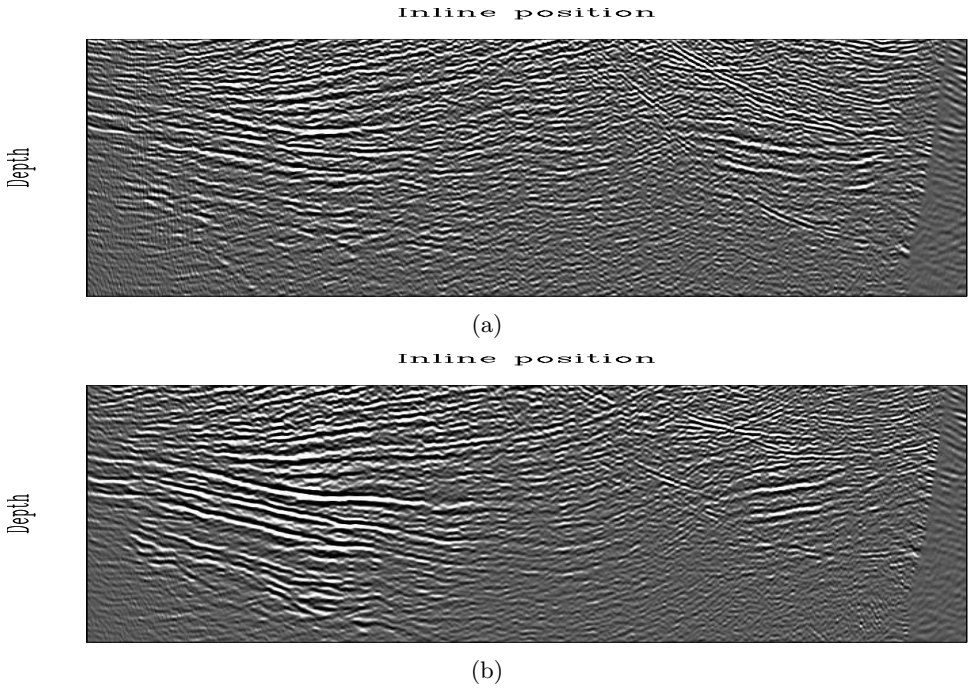


Figure 9: A close-up view of the lower section in Figure 7. (a) is obtained using the initial velocity model and (b) is obtained using the inverted velocity model. [**NR**] yaxun1/. wemva-bpgom2d-bimg-stack-raw-zoom2,wemva-bpgom2d-invt-stack-raw-zoom2

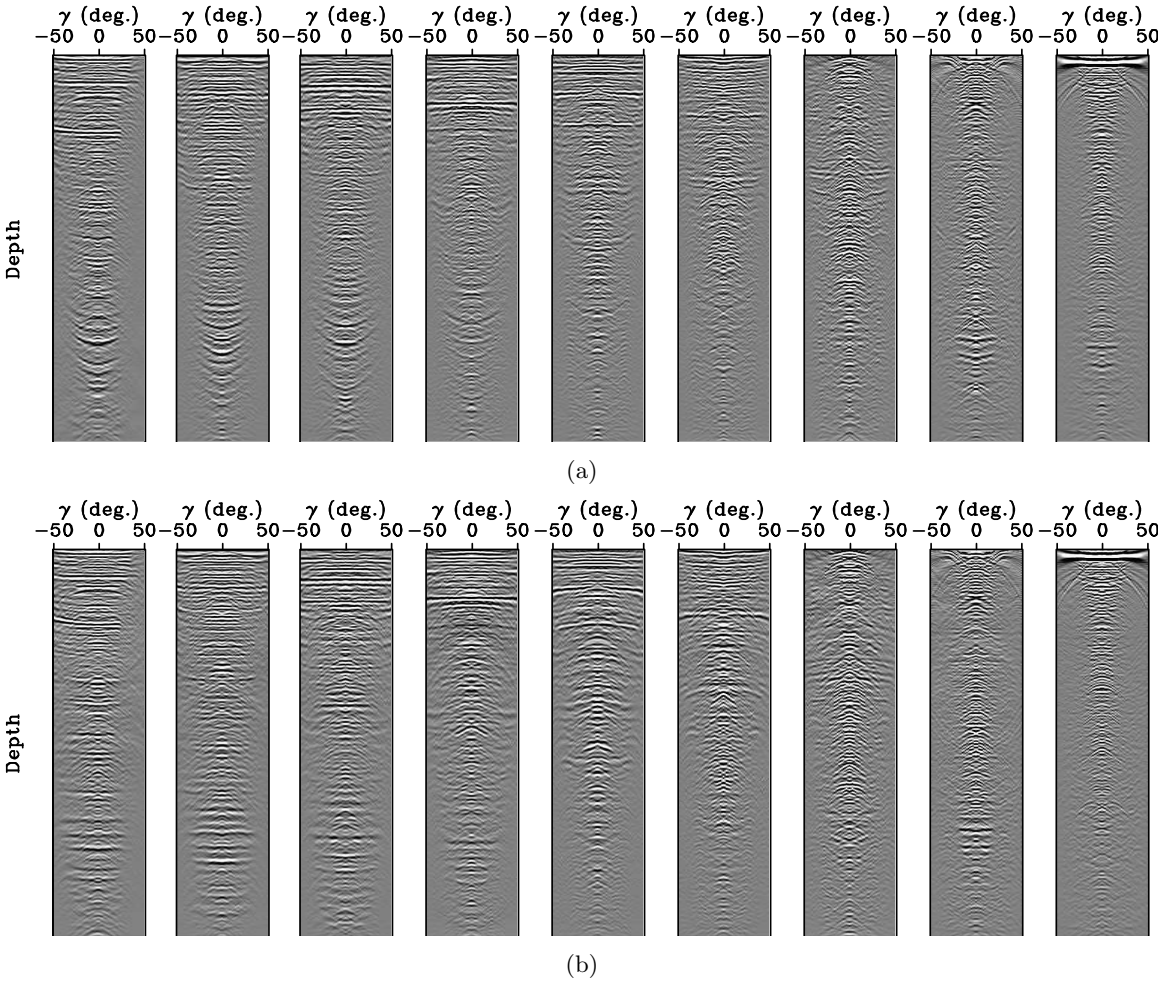


Figure 10: Angle-domain common-image gathers obtained using (a) the initial velocity model and (b) the inverted velocity model. The original data set is used for migration. [**NR**] yaxun1/. wemva-bpgom2d-bimg-adcig-slides,wemva-bpgom2d-imag-invt-adcig-slides

Bleistein, N. and H. Jaramillo, H., 2000, A platform for Kirchhoff data mapping in scalar models of data acquisition: Geophysical Prospecting, **48**, 135–161.

Guerra, C. and B. Biondi, 2010, Fast 3D velocity updates using the pre-stack exploding reflector model: **SEP-140**, 1–10.

Guerra, C., Y. Tang, and B. Biondi, 2009, Wave-equation tomography using image-space phase encoded data: SEG Technical Program Expanded Abstracts, **28**, 3964–3968.

Hoffmann, J., 2001, Illumination, resolution, and image quality of PP- and PS-waves for survey planning: The Leading Edge, **20**, 1008–1014.

Hubral, P., J. Schleicher, and M. Tygel, 1996, A unified approach to 3-D seismic reflection imaging, part I: Basic concepts: Geophysics, **61**, 742–758.

Mora, P., 1989, Inversion = migration + tomography: Geophysics, **54**, 1575–1586.

Plessix, R.-E. and W. A. Mulder, 2004, Frequency-domain finite-difference amplitude-preserving migration: Geophys. J. Int., **157**, 975–987.

Pratt, R. G., 1999, Seismic waveform inversion in the frequency domain, Part 1: Theory and verification in a physical scale model: Geophysics, **64**, 888–901.

Rickett, J. E., 2003, Illumination-based normalization for wave-equation depth migration: Geophysics, **68**, 1371–1379.

Sacchi, M. D. and T. J. Ulrych, 1995, High-resolution velocity gathers and offset space reconstruction: Geophysics, **60**, 1169–1177.

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

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

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.

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

Tang, Y. and B. Biondi, 2010, Target-oriented wavefield tomography using demigrated Born data: **SEP-140**, 11–22.

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

Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, **49**, 1259–1266.

Trad, D. O., T. J. Ulrych, and M. D. Sacchi, 2002, Accurate interpolation with high-resolution time-variant radon transforms: Geophysics, **67**, 644–656.

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

Wang, B., F. Qin, F. Audebert, and V. Dirks, 2005, A fast and low cost alternative to subsalt wave equation migration perturbation scans: SEG Technical Program Expanded Abstracts, **24**, 2257–2260.

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

Migration velocity analysis based on linearization of the two-way wave equation

Ali Almomin and Yaxun Tang

ABSTRACT

Wave equation migration velocity analysis (WEMVA) is a family of techniques that aims to improve the subsurface velocity model by minimizing the residual in image space. This process is performed iteratively by linearizing the imaging operator in order to relate image perturbations to model updates. This linearization is conventionally based on the one-way wave equation, which has some pitfalls in terms of accuracy and ability to image certain wavepaths in complex geology. We present a formulation of WEMVA based on the two-way wave equation which can improve the accuracy of the model estimate. There are two approximations used to linearize this operator. First is the Born approximation and the second envolves dropping the second order slowness perturbation term. In this paper, we show preliminary results of using the two-way based WEMVA, as well as the resolution matrix of the operator.

INTRODUCTION

Seismic velocity analysis methods can be divided into two major groups. First, there are techniques that aim at minimizing the misfit in the data domain such as full waveform inversion (Tarantola, 1984; Luo and Schuster, 1990; Biondi, 2009). Second, there are other techniques that aim at improving the quality in the image domain such as migration velocity analysis (MVA)(Symes and Carazzone, 1991; Biondi and Sava, 1999; Shen, 2004). These techniques try to measure the quality of the image several ways and then invert the estimated image perturbation using a linearized wave equation operator. This tomographic operator is based on a Taylor expansion of the image around a background slowness model.

There are several advantages to minimizing the residual in image-space, such as increasing signal-to-noise ratio and decreasing the complexity of the data (Tang et al., 2008). The linearization in WEMVA is conventionally done based on the one-way wave equation. This approach has some advantages, such as the computational efficiency of one-way wave equation operators. However, it also suffers from disadvantages such as decreased accuracy or the inability to model wide-angle propagations.

In this paper, we show the derivation of a linearized tomographic operator that is based on the two-way wave equation. This operator is the essential part in constructing the gradient of any two-way wave equation based MVA, such as WEMVA by residual moveout fitting (Biondi, 2010). The two-way wave equation is linearized over slowness by dropping the second order slowness perturbation term. Also, the Born approximation is used to

derive this operator. We also show a few ways to interpret and implement this operator. Finally, we show the resolution matrix of this operator.

THEORY

First, we start with the imaging condition as the following:

$$I(\mathbf{x},\mathbf{h}) = \sum_{\omega,\mathbf{x}_s,\mathbf{x}_r} G^*(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega)G^*(\mathbf{x}+\mathbf{h},\mathbf{x}_r,\omega)d(\mathbf{x}_r,\mathbf{x}_s,\omega), \tag{1}$$

where I is the image, G is the Green's function, d is the surface data, $\mathbf{x}_s$ and $\mathbf{x}_r$ are the source and receiver coordinates, $\mathbf{h}$ is the subsurface offset, $\mathbf{x}$ is the Green's functions' coordinate and ω is frequency. Next, we define the Green's functions based on the two-way wave equation as follows:

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

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

where s is slowness. Then, we can obtain the derivative of I with respect to the slowness as follows;

$$\begin{aligned}\frac{\partial I(\mathbf{x},\mathbf{h})}{\partial s(\mathbf{y})} = &\sum_{\omega,\mathbf{x}_s,\mathbf{x}_r} \left(\frac{\partial G(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega)}{\partial s(\mathbf{y})}\right)^* G^*(\mathbf{x}+\mathbf{h},\mathbf{x}_r,\omega)d(\mathbf{x}_r,\mathbf{x}_s,\omega) \\ &+ \sum_{\omega,\mathbf{x}_s,\mathbf{x}_r} G^*(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega)\left(\frac{\partial G(\mathbf{x}+\mathbf{h},\mathbf{x}_r,\omega)}{\partial s(\mathbf{y})}\right)^* d(\mathbf{x}_r,\mathbf{x}_s,\omega),\end{aligned} \tag{4}$$

where $\mathbf{y}$ is the slowness coordinate. Now, we can perturb the slowness:

$$s(\mathbf{x}) = s_0(\mathbf{x}) + \Delta s(\mathbf{x}), \tag{5}$$

where s_0 is the background slowness. Then, we apply a first order approximation by squaring the slowness and ignoring the second order perturbation term as follows:

$$s^2(\mathbf{x}) \approx s_0^2(\mathbf{x}) + 2s_0(\mathbf{x})\Delta s(\mathbf{x}). \tag{6}$$

We define a background Green's function that corresponds to the background slowness:

$$\left(\nabla^2 + \omega^2 s_0^2(\mathbf{x})\right) G_0(\mathbf{x},\mathbf{x}_s,\omega) = -\delta(\mathbf{x}-\mathbf{x}_s). \tag{7}$$

By substituting this into the original wave equation, we arrive at the following:

$$\left(\nabla^2 + \omega^2 s_0^2(\mathbf{x})\right) G(\mathbf{x},\mathbf{x}_s,\omega) = -2\omega^2 s_0(\mathbf{x})\Delta s(\mathbf{x})G(\mathbf{x},\mathbf{x}_s,\omega) - \delta(\mathbf{x}-\mathbf{x}_s). \tag{8}$$

Now, we apply Born's approximation to simplify the previous equation to the following expression:

$$\left(\nabla^2 + \omega^2 s_0^2(\mathbf{x})\right) \Delta G(\mathbf{x},\mathbf{x}_s,\omega) = -2\omega^2 s_0(\mathbf{x})\Delta s(\mathbf{x})G_0(\mathbf{x},\mathbf{x}_s,\omega), \tag{9}$$

where ΔG is the perturbed Green's function. Then, we solve for the perturbed Green's function as follows:

$$\Delta G(\mathbf{x},\mathbf{x}_s,\omega) = -2\omega^2 \sum_{\mathbf{y}} s_0(\mathbf{y})G_0(\mathbf{y},\mathbf{x}_s,\omega)\Delta s(\mathbf{y})G_0(\mathbf{x},\mathbf{y},\omega), \tag{10}$$

which enables us to find the derivative of the Green's function with respect to slowness as shown in the following:

$$\frac{\partial G(\mathbf{x},\mathbf{x}_s,\omega)}{\partial s(\mathbf{y})} = -2\omega^2 s_0(\mathbf{y}) G_0(\mathbf{y},\mathbf{x}_s,\omega) G_0(\mathbf{x},\mathbf{y},\omega). \tag{11}$$

We can follow the same steps for the receiver Green's function to get:

$$\frac{\partial G(\mathbf{x},\mathbf{x}_r,\omega)}{\partial s(\mathbf{y})} = -2\omega^2 s_0(\mathbf{y}) G_0(\mathbf{y},\mathbf{x}_r,\omega) G_0(\mathbf{x},\mathbf{y},\omega), \tag{12}$$

Then, we substitute equations (11) and (12) in the image derivative to get the result:

$$\begin{aligned}
&\frac{\partial I(\mathbf{x},\mathbf{h})}{\partial s(\mathbf{y})}|_{s_0} = \\
&\sum_{\omega,\mathbf{x}_s,\mathbf{x}_r} \left\{-2\omega^2 s_0(\mathbf{y}) G_0^*(\mathbf{y},\mathbf{x}_s,\omega) G_0^*(\mathbf{x}-\mathbf{h},\mathbf{y},\omega)\right\} G_0^*(\mathbf{x}+\mathbf{h},\mathbf{x}_r,\omega) d(\mathbf{x}_r,\mathbf{x}_s,\omega) \\
+ &\sum_{\omega,\mathbf{x}_s,\mathbf{x}_r} \left\{-2\omega^2 s_0(\mathbf{y}) G_0^*(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega) G_0^*(\mathbf{x}+\mathbf{h},\mathbf{y},\omega)\right\} G_0^*(\mathbf{y},\mathbf{x}_r,\omega) d(\mathbf{x}_r,\mathbf{x}_s,\omega).
\end{aligned} \tag{13}$$

Finally, we can express the image perturbation as the following:

$$\begin{aligned}
\Delta I(\mathbf{x},\mathbf{h}) &= \sum_{\mathbf{y}} \frac{\partial I(\mathbf{x},\mathbf{h})}{\partial s(\mathbf{y})} \Delta s(\mathbf{y}) \\
&= \sum_{\omega,\mathbf{x}_s,\mathbf{x}_r,\mathbf{y}} \left\{-2\omega^2 s_0(\mathbf{y}) G_0^*(\mathbf{y},\mathbf{x}_s,\omega) G_0^*(\mathbf{x}-\mathbf{h},\mathbf{y},\omega)\right\} G_0^*(\mathbf{x}+\mathbf{h},\mathbf{x}_r,\omega) d(\mathbf{x}_r,\mathbf{x}_s,\omega) \Delta s(\mathbf{y}) \\
&+ \sum_{\omega,\mathbf{x}_s,\mathbf{x}_r,\mathbf{y}} \left\{-2\omega^2 s_0(\mathbf{y}) G_0^*(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega) G_0^*(\mathbf{x}+\mathbf{h},\mathbf{y},\omega)\right\} G_0^*(\mathbf{y},\mathbf{x}_r,\omega) d(\mathbf{x}_r,\mathbf{x}_s,\omega) \Delta s(\mathbf{y}),
\end{aligned} \tag{14}$$

Similarly, we can now compute the gradient, as given by:

$$\begin{aligned}
\Delta s(\mathbf{y}) &= \sum_{\mathbf{x},\mathbf{h}} \left(\frac{\partial I(\mathbf{x},\mathbf{h})}{\partial s(\mathbf{y})}|_{s_0}\right)^* \Delta I(\mathbf{x},\mathbf{h}) \\
&= \sum_{\omega,\mathbf{x}_s,\mathbf{x}_r,\mathbf{x},\mathbf{h}} \left\{-2\omega^2 s_0(\mathbf{y}) G_0(\mathbf{y},\mathbf{x}_s,\omega) G_0(\mathbf{x}-\mathbf{h},\mathbf{y},\omega)\right\} G_0(\mathbf{x}+\mathbf{h},\mathbf{x}_r,\omega) d^*(\mathbf{x}_r,\mathbf{x}_s,\omega) \Delta I(\mathbf{x},\mathbf{h}) \\
&+ \sum_{\omega,\mathbf{x}_s,\mathbf{x}_r,\mathbf{x},\mathbf{h}} \left\{-2\omega^2 s_0(\mathbf{y}) G_0(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega) G_0(\mathbf{x}+\mathbf{h},\mathbf{y},\omega)\right\} G_0(\mathbf{y},\mathbf{x}_r,\omega) d^*(\mathbf{x}_r,\mathbf{x}_s,\omega) \Delta I(\mathbf{x},\mathbf{h}) \\
&= -2\omega^2 s_0(\mathbf{y}) \sum_{\omega,\mathbf{x}_s,\mathbf{x}_r} G_0(\mathbf{y},\mathbf{x}_s,\omega) d^*(\mathbf{x}_r,\mathbf{x}_s,\omega) \sum_{\mathbf{x},\mathbf{h}} G_0(\mathbf{x}-\mathbf{h},\mathbf{y},\omega) G_0(\mathbf{x}+\mathbf{h},\mathbf{x}_r,\omega) \Delta I(\mathbf{x},\mathbf{h}) \\
&- 2\omega^2 s_0(\mathbf{y}) \sum_{\omega,\mathbf{x}_s,\mathbf{x}_r} G_0(\mathbf{y},\mathbf{x}_r,\omega) d^*(\mathbf{x}_r,\mathbf{x}_s,\omega) \sum_{\mathbf{x},\mathbf{h}} G_0(\mathbf{x}+\mathbf{h},\mathbf{y},\omega) G_0(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega) \Delta I(\mathbf{x},\mathbf{h}).
\end{aligned} \tag{15}$$

INTERPRETATION

There are several ways to interpret equations (14) and (15). For simplicity, let us first break each perturbation into two components, one from the source side and one from the receiver

side. So, for equation (14), the first component will be as following:

$$\Delta I_S(\mathbf{x},\mathbf{h}) = \sum_{\omega,\mathbf{x}_s}\left\{\sum_{\mathbf{y}} -2\omega^2 s_0(\mathbf{y})G_0^*(\mathbf{y},\mathbf{x}_s,\omega)\Delta s(\mathbf{y})G_0^*(\mathbf{x}-\mathbf{h},\mathbf{y},\omega)\right\} \left\{\sum_{\mathbf{x}_r} G_0^*(\mathbf{x}+\mathbf{h},\mathbf{x}_r,\omega)d(\mathbf{x}_r,\mathbf{x}_s,\omega)\right\}. \tag{16}$$

Now, we can further break equation (16) into two components that we define as follows:

$$\Delta S(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega) = -2\omega^2\sum_{\mathbf{y}} s_0(\mathbf{y})G_0(\mathbf{y},\mathbf{x}_s,\omega)\Delta s(\mathbf{y})G_0(\mathbf{x}-\mathbf{h},\mathbf{y},\omega), \tag{17}$$

and

$$R_0(\mathbf{x}+\mathbf{h},\mathbf{x}_s,\omega) = \sum_{\mathbf{x}_r} G_0^*(\mathbf{x}+\mathbf{h},\mathbf{x}_r,\omega)d(\mathbf{x}_r,\mathbf{x}_s,\omega). \tag{18}$$

We can see that equation (17) represents the Born-modeled-wavefield due to the slowness perturbation and equation (18) represents the background receiver wavefield. So, we can now present the source side of the image perturbation as the following:

$$\Delta I_S(\mathbf{x},\mathbf{h}) = \sum_{\omega,\mathbf{x}_s}\Delta S_0^*(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega)R_0(\mathbf{x}+\mathbf{h},\mathbf{x}_s,\omega). \tag{19}$$

Now, we can perform a similar analysis on the other component of equation (14), which is:

$$\Delta I_R(\mathbf{x},\mathbf{h}) = \sum_{\omega,\mathbf{x}_s} G_0^*(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega)\{\sum_{\mathbf{y}} -2\omega^2 s_0(\mathbf{y})G_0^*(\mathbf{x}+\mathbf{h},\mathbf{y},\omega)\Delta s(\mathbf{y})\} \{\sum_{\mathbf{x}_r} G_0^*(\mathbf{y},\mathbf{x}_r,\omega)d(\mathbf{x}_r,\mathbf{x}_s,\omega)\}. \tag{20}$$

Again, let us define a perturbed receiver wavefield and a background source wavefield as the following:

$$R_0(\mathbf{y},\mathbf{x}_s,\omega) = \sum_{\mathbf{x}_r} G_0^*(\mathbf{y},\mathbf{x}_r,\omega)d(\mathbf{x}_r,\mathbf{x}_s,\omega), \tag{21}$$

$$\Delta R(\mathbf{x}+\mathbf{h},\mathbf{x}_s,\omega) = -2\omega^2\sum_{\mathbf{y}} s_0(\mathbf{y})G_0^*(\mathbf{x}+\mathbf{h},\mathbf{y},\omega)\Delta s(\mathbf{y})R_0(\mathbf{y},\mathbf{x}_s,\omega), \tag{22}$$

$$S_0(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega) = G_0(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega). \tag{23}$$

This enables us to represent the receiver side of the image perturbation as the following:

$$\Delta I_R(\mathbf{x},\mathbf{h}) = \sum_{\omega,\mathbf{x}_s} S_0^*(\mathbf{x}-\mathbf{h},\mathbf{x}_s,\omega)\Delta R(\mathbf{x}+\mathbf{h},\mathbf{x}_s,\omega). \tag{24}$$

As for equation (15), we do the same analysis to arrive at the following gradient formulae:

$$\Delta s_R(\mathbf{y}) = \sum_{\omega,\mathbf{x}_s} S_0(\mathbf{y},\mathbf{x}_s,\omega)\Delta R^*(\mathbf{y},\mathbf{x}_s,\omega), \tag{25}$$

$$\Delta s_S(\mathbf{y}) = \sum_{\omega,\mathbf{x}_s} \Delta S(\mathbf{y},\mathbf{x}_s,\omega)R_0^*(\mathbf{y},\mathbf{x}_s,\omega), \tag{26}$$

where the residual wavefields are defined as the following:

$$\Delta R(\mathbf{y}, \mathbf{x}_s, \omega) = -2\omega^2 \sum_{\mathbf{x},\mathbf{h}} s_0(\mathbf{y}) G_0^*(\mathbf{x}-\mathbf{h}, \mathbf{y}, \omega) \Delta I(\mathbf{x}, \mathbf{h}) R_0(\mathbf{x}+\mathbf{h}, \mathbf{x}_s, \omega), \tag{27}$$

$$\Delta S(\mathbf{y}, \mathbf{x}_s, \omega) = -2\omega^2 \sum_{\mathbf{x},\mathbf{h}} s_0(\mathbf{y}) G_0(\mathbf{x}+\mathbf{h}, \mathbf{y}, \omega) G_0(\mathbf{x}-\mathbf{h}, \mathbf{x}_s, \omega) \Delta I(\mathbf{x}, \mathbf{h}). \tag{28}$$

In summary, the tomographic operator computes the image perturbation or slowness perturbation by correlating background and residual wavefields of both source and receiver sides.

SYNTHETIC EXAMPLES

Although the derivation was performed in the frequency-domain, we will apply the tomographic operator in the time-domain. First, we will start with a simple example with a constant background velocity of 2500 m/s. The spatial sampling is 10 m and the temporal sampling is 2 ms. A Ricker wavelet with a fundamental frequency of 20 Hz is used to model the data. There is one reflector at the bottom of the model at a depth of 900 m. Now, we will input a slowness perturbation to the forward operator to generate a corresponding image perturbation. Three slowness perturbations are supplied. First, a spike located at a depth of 400 m. Second, a vertical line extending from a depth of 300 m to 500 m. Third, a horizonal line at a depth of 400 m. The three slowness perturbations are shown in Figure 1. We apply the forward tomographic operator on these slowness perturbations to get the corresponding image perturbations. Figure 2 shows the image perturbation (at zero subsurface offset only). Then, we apply the adjoint tomographic operator to these image perturbations to recreate the slowness perturbations. The results of applying the adjoint scattering operator are shown in Figure 3. As expected, the reconstructed slowness perturbations have higher horizontal resolution than vertical resolution. Figure 4 shows the amplitude spectrum of the recreated slowness perturbation in Figure 3(a).

For a second test, we will repeat a similar experiment but with a different background velocity model. As shown in Figure 5, the velocity model includes areas of low velocity to the top, and areas of high velocity in the middle, representing a salt body. There is one reflector at the bottom of the model at a depth of 3500 m. The spatial sampling is 25 m and the temporal sampling is 4 ms. A Ricker wavelet with a fundamental frequency of 10 Hz is used to model the data. The slowness perturbation is a spike located at a depth of 2600 m, which is located in between the reflector and the salt body. The corresponding image perturbation resulted from applying the forward tomographic operator is shown in Figure 6(a) and the reconstructed slowness perturbation is shown in Figure 6(b). Figure 6(c) shows the amplitude spectrum of the reconstructed slowness perturbation. The change in the background velocity affected the reconstructed slowness perturbation, both in physical space and in Fourier space.

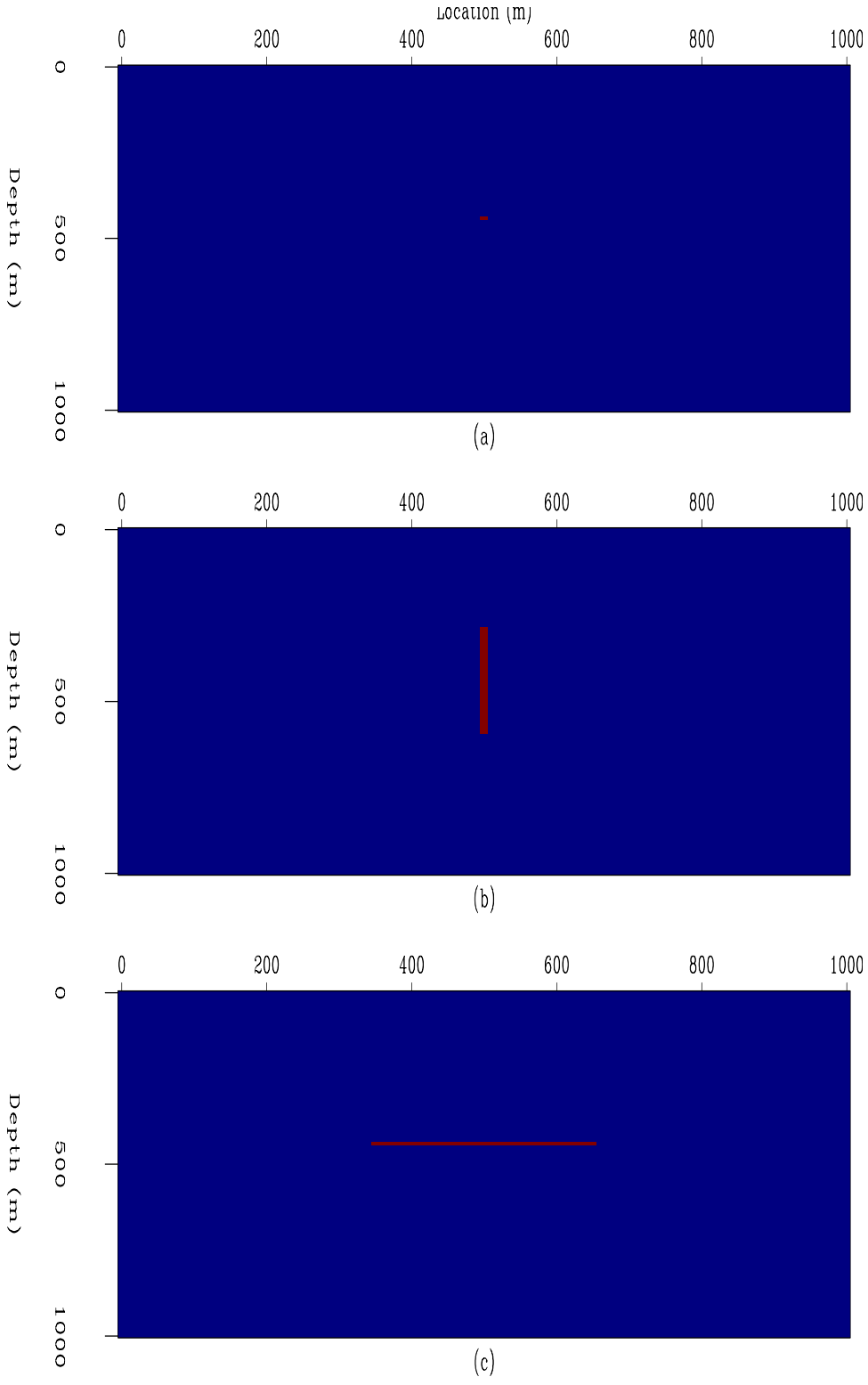


Figure 1: Three slowness perturbations that will be used in the forward operator. [**ER**] ali1/. deltaS

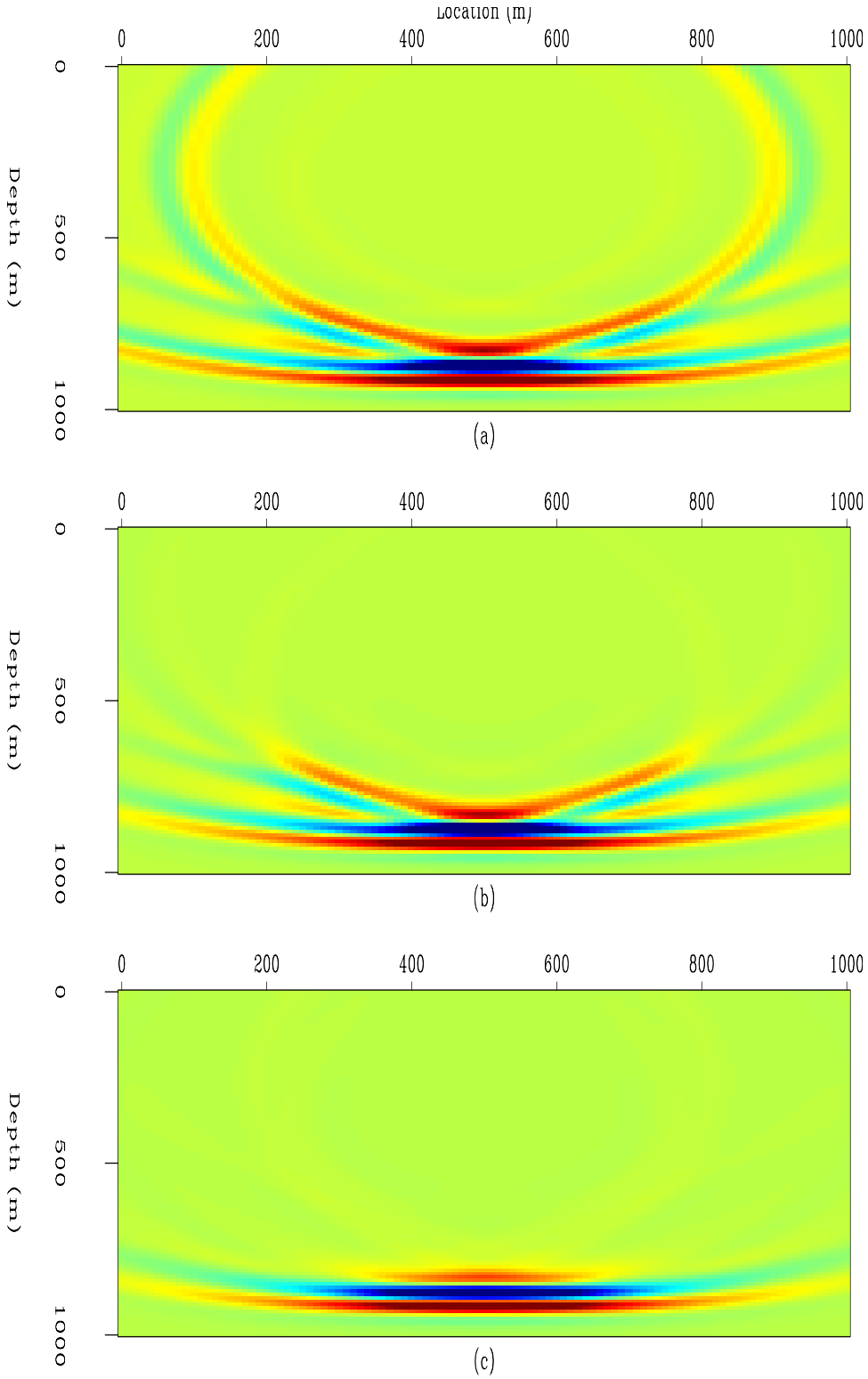


Figure 2: The three image perturbations corresponding to slowness perturbations in Figure 1, produced by the forward scattering operator. [**ER**] ali1/. deltaI

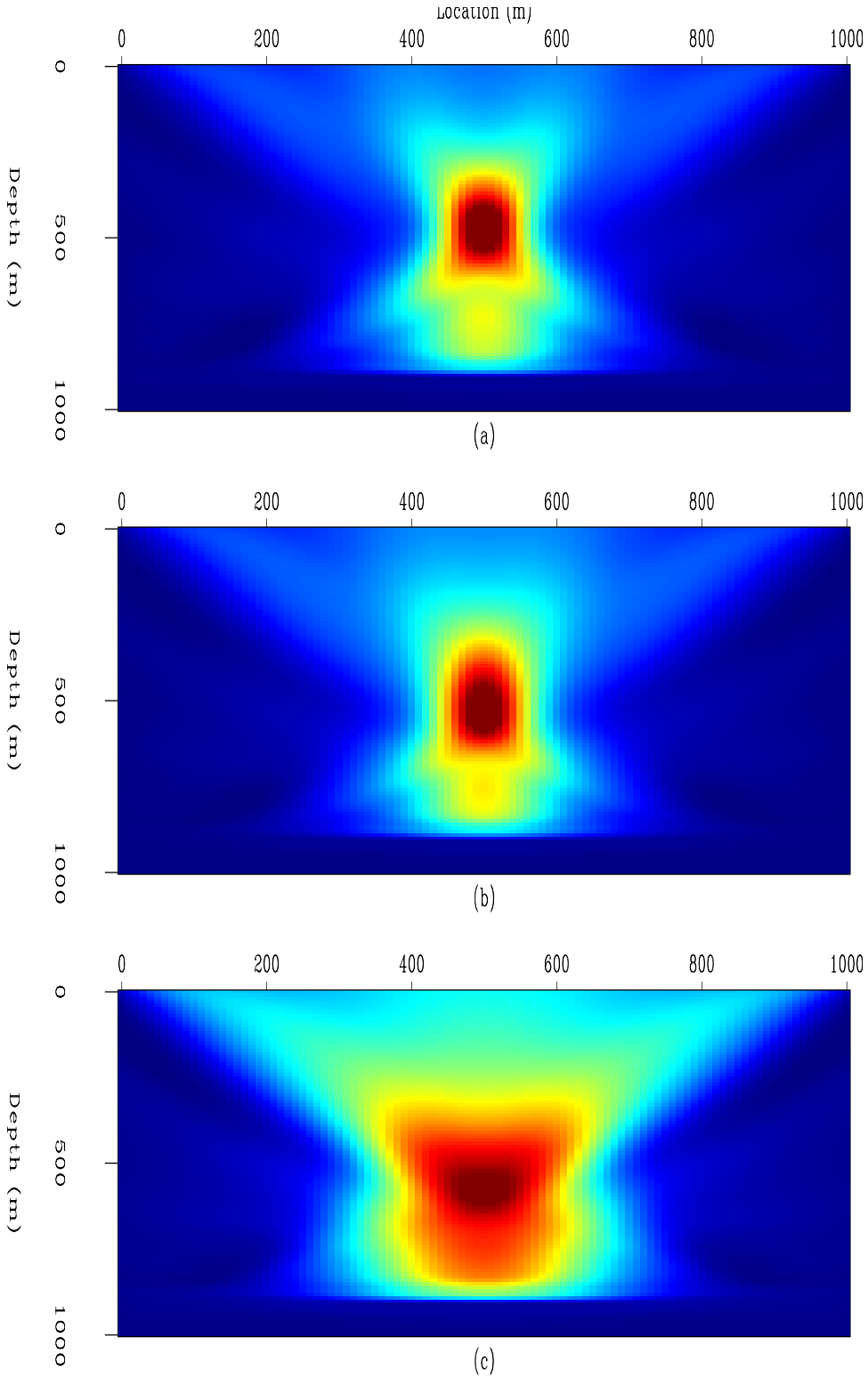


Figure 3: The reconstructed slowness perturbations by the adjoint scattering operator. [**ER**] ali1/. deltaS2

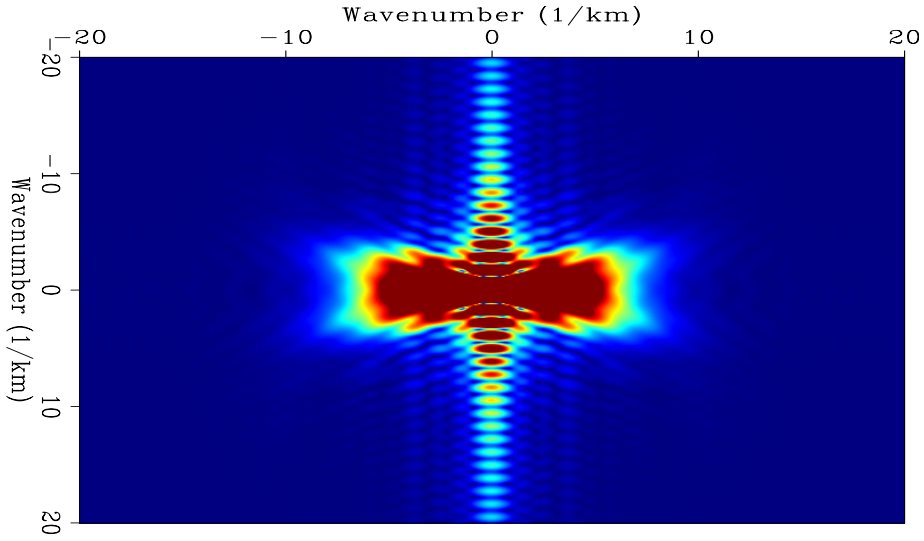


Figure 4: The Fourier transform of spike response in Figure 3(a). [**ER**] ali1/. spect

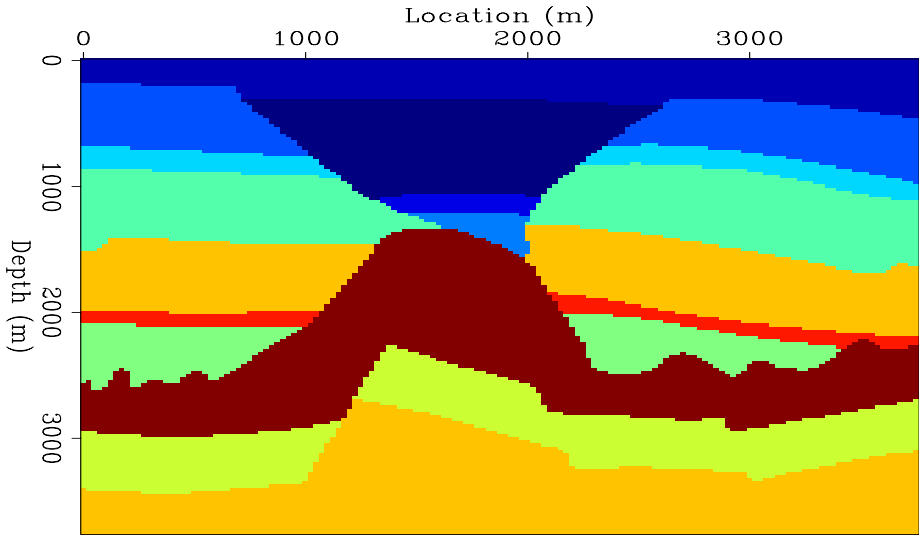


Figure 5: The background velocity model for the second test. [**ER**] ali1/. velbg3

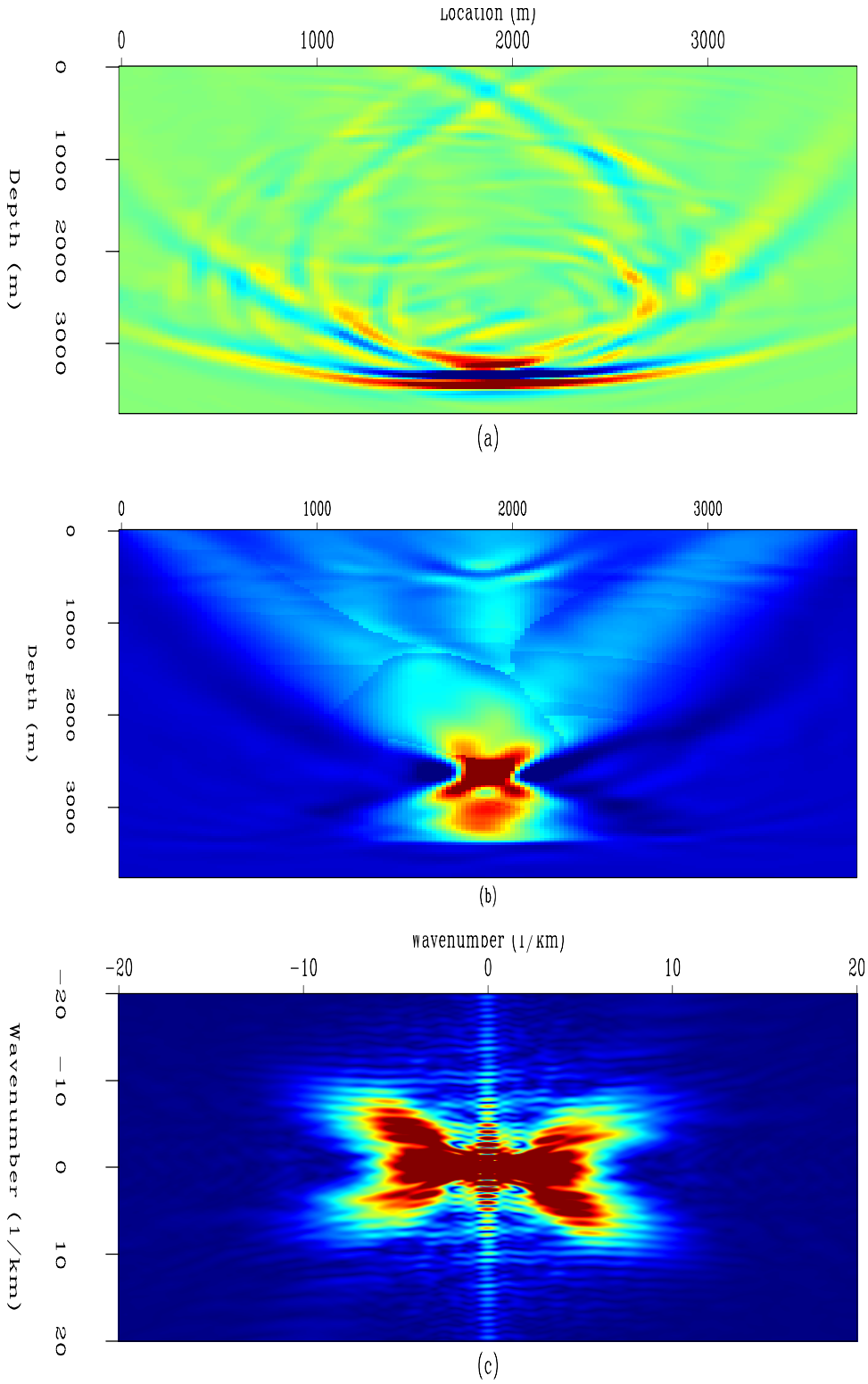


Figure 6: Using the background velocity in Figure 5, (a) the image perturbation, (b) reconstructed slowness perturbation, and (c) the Fourier transform for the reconstructed slowness perturbation. [**ER**] ali1/. results3

CONCLUSIONS

In this paper, we derived the tomographic operator for wave equation migration velocity analysis tools based on the two-way wave equation. Only two approximations were used to derive the tomographic operator. Then, we tested the operator using two synthetic velocity models: first with constant velocity, and second with a more complex model that includes a high velocity salt body. The operator seems to give satisfactory results in both physical and Fourier spaces.

FUTURE WORK

The tomographic operator we derive can potentially produce superior results to the operators based on one-way wave equation since it can capture wave paths with more accuracy. So, the next step is to test the migration velocity estimation results on a complex synthetic model, where one-way propagation operators fail and measure the amount of improvement. Then, the velocity estimation process should be tested on a real dataset.

Finally, since this method is very computationally intensive, it is crucial to adapt it on accelerated and parallel processing units such as GPUs. This is especially attractive since GPUs are particularly efficient for finite-difference-based convolution algorithms such as propagation in time domain.

REFERENCES

Biondi, B., 2009, Measuring image focusing for velocity analysis: SEP-Report, **138**, 59–80.

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

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

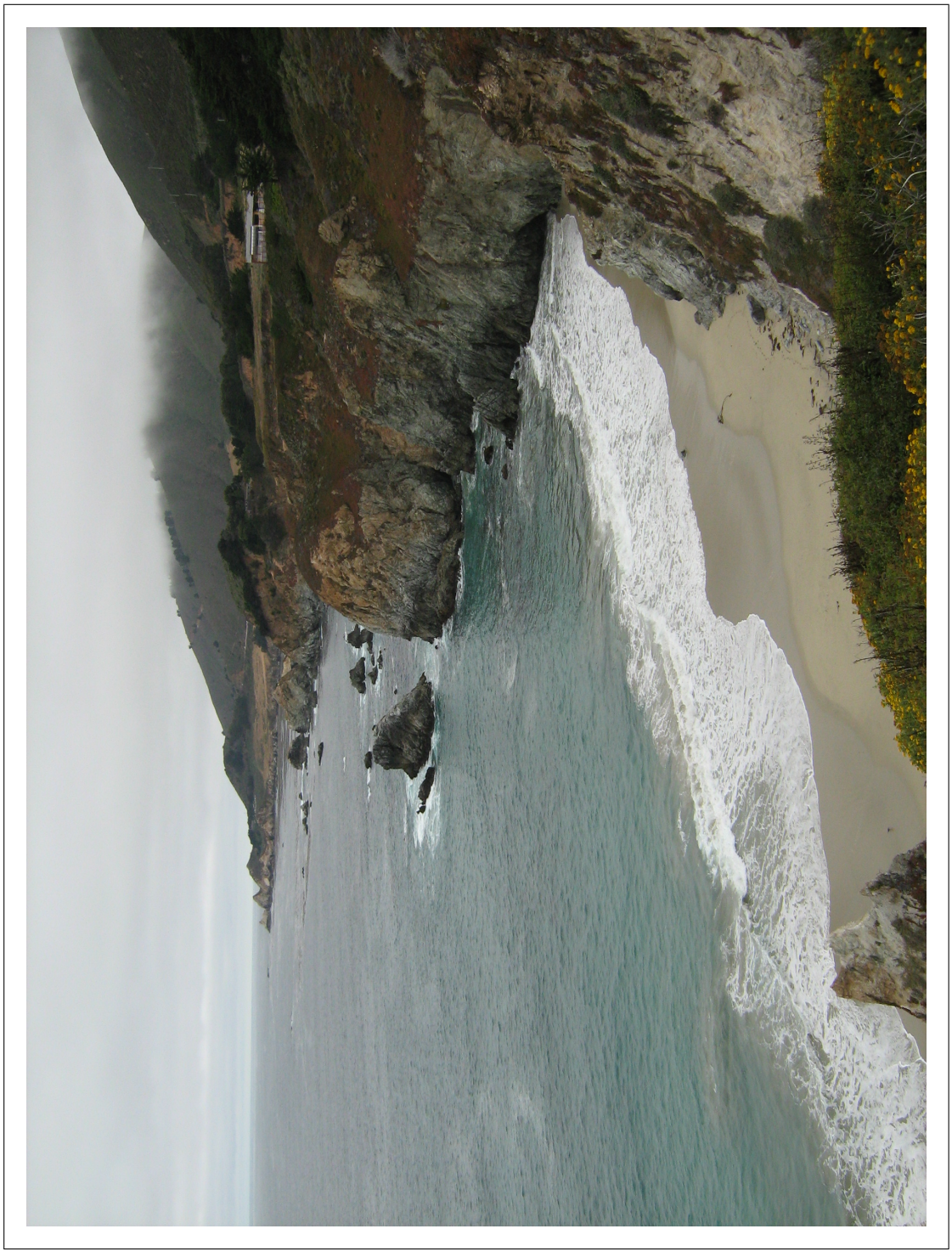
Luo, Y. and G. T. Schuster, 1990, Wave-equation traveltime inversion: SEG Expanded Abstracts.

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

Symes, W. W. and J. J. Carazzone, 1991, Velocity inversion by differential semblance optimization: Geophysics, **56**, 654–663.

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

Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, **49**, 1259–1266.



Wave-equation migration velocity analysis by residual moveout fitting

Biondo Biondi

ABSTRACT

Flatness in migrated angle-domain common image gathers is an effective criterion for measuring migration-velocity accuracy. An objective function that measures the power of the stack as a function of residual-moveout parameters directly, and indirectly as a function of migration velocity, can be robustly maximized by using a gradient-based method. This paper presents a method to compute the gradient of this objective function by wave-equation operators. The proposed algorithm has the additional advantage of not requiring the picking of the residual-moveout parameters.

INTRODUCTION

In this paper I build on the framework I presented in Biondi (2010). In that report I presented a tomographic velocity estimation that aims to maximize image focusing using wave-equation operators. In SEP 140 I developed the theory and showed the results of numerical tests for a transmission tomography problem, because transmission tomography is simpler than reflection tomography. In this paper I extend that theory to the broader application of migration velocity analysis (MVA).

Conventional MVA methods are often based on the maximization of the stack power of migrated angle-domain common image gathers. However, direct maximization of the stack power as a function of velocity by wave-equation operators has well-known convergence problems (Chavent and Jacewitz, 1995; Biondi, 2006; Symes, 2008). To overcome these challenges, I propose to maximize stack power as a function of residual-moveout parameters, instead of directly as a function of velocity. In turn, the residual-moveout parameters are defined as solutions of fitting problems that maximize the correlation between the moved-out gathers and the gathers obtained by migrating the recorded data with the given velocity. These fitting problems can be quickly solved by using one-parameter gradient methods and thus do not require the explicit picking of residual-moveout parameters. The avoidance of parameter picking is an important advantage with respect to conventional wave-equation MVA methods (Biondi and Sava, 1999; Sava and Biondi, 2004a,b; Sava, 2004).

This paper presents just the theoretical development without any numerical examples illustrating the proposed method. I plan to present the application and the testing of this theory in upcoming reports.

THEORY

In wave-equation migration, as for example reverse-time migration, the image is computed from the back-propagated receiver wavefield, $P_g\left(t,\vec{x},x_s;s\right)$, and the forward-propagated

source wavefield, $P_s\left(t,\vec{x},x_s;s\right)$, where t is the recording time, $\vec{x}=z\vec{z}_0+x\vec{x}_0$ is the model-coordinate vector, x_s is the source position at the surface, and $s\left(\vec{x}\right)$ is the slowness model.

The prestack image, $I_h\left(\vec{x},x_h\right)$, is computed as the zero lag of the temporal cross-correlation between the spatially-shifted back-propagated receiver wavefield and forward-propagated source wavefield as (Rickett and Sava, 2002):

$$I_h\left(\vec{x},\vec{x_h}\right)\left[P_s\left(t,\vec{x}\right),P_g\left(t,\vec{x}\right)\right]=\sum_t\sum_{x_s}P_g\left(t,\vec{x}-\vec{x_h},x_s\right)P_s\left(t,\vec{x}+\vec{x_h},x_s\right), \tag{1}$$

where $\vec{x_h}=x_h\vec{x}_0$ is the *half subsurface offset*, which in this paper I will assume to be horizontal, but it does not need to be in the general case (Biondi and Symes, 2004).

The prestack image as a function of subsurface offset can be transformed to an image as a function of reflection aperture angle, $I_\gamma\left(\vec{x},\gamma\right)$ by using a linear operator $\mathbf{\Gamma}$ (Sava and Fomel, 2003). In matrix notation, if $\mathbf{I}_h$ is a $N_{\vec{x}}N_h\times 1$ matrix and $\mathbf{I}_\gamma$ is a $N_{\vec{x}}N_\gamma\times 1$ matrix, the image transformation from subsurface offset into the angle domain can be expressed as:

$$\mathbf{I}_\gamma=\mathbf{\Gamma}\mathbf{I}_h. \tag{2}$$

I introduce an objective function that maximizes the flatness of the angle-domain image along the aperture-angle axis at all spatial locations $\vec{x}$. This objective function aims at maximizing image flatness not directly as a function of the slowness, but indirectly through the application of an angle-domain moveout operator $\mathcal{M}_\gamma$, which depends on the column vector of $N_\mu=N_{\vec{x}}$ moveout parameters $\boldsymbol{\mu}_{\vec{x}}$.

I define the application of the moveout operators $\mathcal{M}_\gamma$ to a prestack image computed by equations 1 and 2 with a background slowness $\bar{s}$, as

$$I_\gamma\left(\vec{x}+\vec{\zeta}\left(\boldsymbol{\mu}_{\vec{x}}\right),\gamma;\bar{s}\right)=\mathcal{M}_\gamma\left[\boldsymbol{\mu}_{\vec{x}}\right]I_\gamma\left(\vec{x},\gamma;\bar{s}\right), \tag{3}$$

where $\vec{\zeta}\left(\boldsymbol{\mu}_{\vec{x}}\right)=\zeta\left(\boldsymbol{\mu}_{\vec{x}}\right)\vec{z}_0$ are the moveout shifts, assumed here to be simple depth shifts. The operator $\mathcal{M}_\gamma$ is linear with respect to the input image, but it is nonlinear with respect to the vector of moveout parameters $\boldsymbol{\mu}_{\vec{x}}$. In matrix notation, the application of the moveout operator to the background image $\bar{\mathbf{I}}_\gamma$ can be expressed as $\mathcal{M}_\gamma\left[\boldsymbol{\mu}_{\vec{x}}\right]\bar{\mathbf{I}}_\gamma$.

I further define the stacking operator $\mathbf{S}_\gamma$ that sums the image along the aperture-angle axis γ. I can now introduce the objective function that measures the flatness of the image as:

$$J\left(\boldsymbol{\mu}_{\vec{x}}\left(\mathbf{s}\right)\right)=\frac{1}{2}\left\|\mathbf{S}_\gamma\mathcal{M}_\gamma\left[\boldsymbol{\mu}_{\vec{x}}\left(\mathbf{s}\right)\right]\bar{\mathbf{I}}_\gamma\right\|_2^2, \tag{4}$$

where $\mathbf{s}$ is the slowness vector. This objective function is not a direct function of $\mathbf{s}$, but it depends on it indirectly through the moveout parameters $\boldsymbol{\mu}_{\vec{x}}$. The dependency of the moveout parameters from the slowness function is not defined explicitly; these parameters are defined as the solutions of $N_{\vec{x}}$ independent fitting problems, one for each spatial location in the image.

The fitting problems maximize the zero lag of the cross-correlation between the prestack image computed for a realization of the slowness vector $\mathbf{s}$ and the moved-out image computed with the background slowness $\bar{\mathbf{s}}$. For the sake of keeping the notation as compact as possible,

I combine the $N_{\vec{x}}$ independent fitting problems into one by defining the following objective function:

$$J_{\mathrm{F}}\left(\boldsymbol{\mu}_{\vec{x}}\left(\mathbf{s}\right)\right) = \mathbf{S}_{\vec{x}}\left\langle \mathcal{M}_{\gamma}\left[\boldsymbol{\mu}_{\vec{x}}\right]\bar{\mathbf{I}}_{\gamma}, \mathbf{I}_{\gamma}\left(\mathbf{s}\right)\right\rangle_{\gamma}, \tag{5}$$

where with the notation $\langle \mathbf{x}, \mathbf{y}\rangle_{\gamma}$ I indicate the ensemble of inner products between the image vectors $\mathbf{x}$ and $\mathbf{y}$ which spans only the aperture-angle axis γ; the result of these inner products is a vector of length $N_{\vec{x}}$. The stacking operator $\mathbf{S}_{\vec{x}}$ sums the elements of this vector to combine the objective functions into one.

The vector of moveout parameters is therefore the solutions of the following maximization problem:

$$\max_{\boldsymbol{\mu}_{\vec{x}}} J_{\mathrm{F}}\left(\boldsymbol{\mu}_{\vec{x}}\left(\mathbf{s}\right)\right). \tag{6}$$

For velocity estimation in the angle domain, an effective parametrization of the moveout is the "curvature" $\mu_{\vec{x}}$, that defines the following moveout equation

$$\zeta\left(\mu_{\overline{x}}\right) = \mu_{\vec{x}} \tan^2\gamma. \tag{7}$$

Notice that when the slowness $\mathbf{s}$ is equal to the background slowness $\bar{\mathbf{s}}$, the corresponding best-fitting moveout parameters $\bar{\mu}_{\vec{x}}$ are obviously the ones corresponding to no moveout; that is, $\zeta\left(\bar{\mu}_{\vec{x}}\right) = 0$, and consequently $\bar{\mu}_{\vec{x}} = 0$.

Gradient of the objective function

I plan to solve the optimization problem defined in 4 by a gradient-based optimization algorithm. Therefore, the development of an algorithm for efficiently computing the gradient of the objective function with respect to slowness is an essential step to make the method practical. In this section I outline the methodology to compute the gradient, and I leave some of the details to Appendix A.

The gradient is computed using the chain rule. The first term of the chain is the derivative of the objective function in equation 4 with respect the moveout parameters. The second term is the derivatives of the moveout parameters with respect to slowness that are computed from the objective function 5.

Derivatives of objective function (J) with respect to moveout parameters ($\boldsymbol{\mu}_{\vec{x}}$)

The derivatives of 4 with respect to the vector of moveout parameters is easily evaluated using the following expression:

$$\frac{\partial J'}{\boldsymbol{\mu}_{\vec{x}}} = \frac{\partial \mathcal{M}_{\gamma}{}'}{\partial \boldsymbol{\mu}_{\vec{x}}} \mathbf{S}'_{\gamma}\mathbf{S}_{\gamma}\mathcal{M}_{\gamma}\left[\bar{\boldsymbol{\mu}}_{\vec{x}}\right]\bar{\mathbf{I}}_{\gamma}. \tag{8}$$

The linear operator $\frac{\partial \mathcal{M}_{\gamma}}{\partial \boldsymbol{\mu}_{\vec{x}}}$ can be represented as a $N_{\vec{x}}N_{\gamma} \times N_{\mu_{\overline{x}}}$ matrix. The elements of this matrix are given by:

$$\frac{\partial \mathcal{M}_{\gamma}}{\partial \boldsymbol{\mu}_{\vec{x}}}\left(\vec{x}, \gamma, \mu_{\vec{x}}\right) = \underbrace{\mathcal{M}_{\gamma}\left[\boldsymbol{\mu}_{\vec{x}}\right]\dot{I}_{\gamma}\left(\vec{x}, \gamma; \bar{s}\right)}_{\mathrm{I}} \underbrace{\frac{\partial \zeta}{\partial \mu_{\vec{x}}}}_{\mathrm{II}}. \tag{9}$$

The first term (I) is given by the depth-derivative of the image $\partial I_\gamma(\vec{x},\gamma;\bar{s})/\partial z$ after moveout. This term can be numerically evaluated by applying to the moved-out image a finite-difference approximation of the first-derivative operator. The second term (II) is different from zero only when the spatial coordinate $\vec{x}$ of the image element $I_\gamma(\vec{x},\gamma)$ is the same as the coordinate corresponding to the moveout parameter $\mu_{\vec{x}}$. When they do, and for the choice of moveout parameters expressed in equation 7, we have $\partial\zeta/\partial\mu_{\vec{x}} = \tan^2\gamma$.

The preceding expression simplifies when the gradient is evaluated for $\boldsymbol{\mu}_{\vec{x}} = 0$. This simplifying condition is actually always fulfilled unless the optimization algorithm includes inner iterations for fitting the moveout parameters using a linearized approach. Under this condition, equation 8 becomes

$$\left.\frac{\partial J}{\boldsymbol{\mu}_{\vec{x}}}\right|'_{\boldsymbol{\mu}_{\vec{x}}=0} = \left.\frac{\partial \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}}\right|'_{\boldsymbol{\mu}_{\vec{x}}=0} \mathbf{S}'_\gamma \mathbf{S}_\gamma \bar{\mathbf{I}}_\gamma, \tag{10}$$

and equation 9 becomes

$$\frac{\partial \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}}(\vec{x},\gamma,\mu_{\vec{x}}=0) = \dot{I}_\gamma(\vec{x},\gamma;\bar{s})\frac{\partial\zeta}{\partial\boldsymbol{\mu}_{\vec{x}}}. \tag{11}$$

Derivatives of moveout parameters ($\mu_{\vec{x}}$) with respect to slowness (s)

The evaluation of the derivatives of the moveout parameters with respect to slowness takes advantage of the fact that we need to evaluate the derivatives only at maxima for the objective function in equation 5. At the maxima, the objective function is stationary and thus its derivatives with respect to the moveout parameters are zero, and we can write:

$$\left.\frac{\partial J_{\mathrm{F}}(\boldsymbol{\mu}_{\vec{x}})}{\partial\boldsymbol{\mu}_{\vec{x}}}\right|_{\boldsymbol{\mu}_{\vec{x}}=\bar{\boldsymbol{\mu}}_{\vec{x}}} = \dot{J}_{\mathrm{F}}(\bar{\boldsymbol{\mu}}_{\vec{x}}) = \mathbf{S}_{\vec{x}}\left\langle \left.\frac{\partial\mathcal{M}_\gamma}{\partial\boldsymbol{\mu}_{\vec{x}}}\right|_{\boldsymbol{\mu}_{\vec{x}}=\bar{\boldsymbol{\mu}}_{\vec{x}}}, \mathbf{I}_\gamma\right\rangle_\gamma = 0. \tag{12}$$

As discussed above, the derivatives in the second term (II) of equation 9 are different from zero only when the moveout coefficient $\mu_{\vec{x}}$ and the image element share the same spatial coordinate. Consequently, for each $\mu_{\overline{x}}$ there is only one $\vec{x}$ for which the inner products above are different from zero. Equation 12 can thus be simplified into:

$$\dot{J}_{\mathrm{F}}(\bar{\boldsymbol{\mu}}_{\vec{x}}) = \left\langle \left.\frac{\partial\mathcal{M}_\gamma}{\partial\boldsymbol{\mu}_{\vec{x}}}\right|_{\boldsymbol{\mu}_{\vec{x}}=\bar{\boldsymbol{\mu}}_{\vec{x}}}, \mathbf{I}_\gamma\right\rangle_\gamma = 0. \tag{13}$$

Using the rule for differentiating implicit functions, and taking advantage of the fact that the fitting problems are all independent from each other (i.e. the cross derivatives with respect to the moveout parameters are all zero), we can formally write:

$$\left.\frac{\partial\boldsymbol{\mu}_{\vec{x}}}{\partial\mathbf{s}}\right|_{\mu_{\overline{x}}=\bar{\mu}_{\vec{x}}} = -\frac{\frac{\partial\dot{J}_{\mathrm{F}}(\boldsymbol{\mu}_{\vec{x}})}{\partial\mathbf{s}}}{\frac{\partial\dot{J}_{\mathrm{F}}(\boldsymbol{\mu}_{\vec{x}})}{\partial\boldsymbol{\mu}_{\vec{x}}}}. \tag{14}$$

From equation 13 and 14, the derivative of the local moveout parameters with respect to slowness is:

$$\left.\frac{\partial \boldsymbol{\mu}_{\vec{x}}}{\partial \mathbf{s}}\right|_{\mu_{\overline{x}}=\bar{\mu}_{\vec{x}}} = -\frac{\left\langle \left.\frac{\partial \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}}\right|_{\boldsymbol{\mu}_{\vec{x}}=\bar{\boldsymbol{\mu}}_{\vec{x}}}, \frac{\partial \mathbf{I}_\gamma}{\partial \mathbf{s}} \right\rangle_\gamma}{\left\langle \left.\frac{\partial^2 \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}^2}\right|_{\boldsymbol{\mu}_{\vec{x}}=\bar{\boldsymbol{\mu}}_{\vec{x}}}, \mathbf{I}_\gamma \right\rangle_\gamma}. \tag{15}$$

Appendix A presents the development for the expressions to compute the terms $\partial^2 \mathcal{M}_\gamma / \partial \boldsymbol{\mu}_{\vec{x}}^2$ (A-3), and $\partial \mathbf{I}_\gamma / \partial \mathbf{s}$ (A-5).

Combining the derivatives in equation 15 with the derivatives in equations 10-11 we can easily compute the gradient of the objective function 4 with respect to slowness that can be written, when $\bar{\boldsymbol{\mu}}_{\vec{x}} = 0$, as:

$$\nabla J = -\underbrace{\left(\frac{\partial \mathbf{P}_s{}'}{\partial \mathbf{s}} \bar{\mathbf{P}}'_g + \frac{\partial \mathbf{P}_g{}'}{\partial \mathbf{s}} \bar{\mathbf{P}}'_s \right) \boldsymbol{\Gamma}'}_{\text{I}} \underbrace{\frac{\partial \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}}}_{\text{II}} \underbrace{\frac{\frac{\partial \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}}{}' \mathbf{S}'_\gamma \mathbf{S}_\gamma \bar{\mathbf{I}}_\gamma}{\frac{\partial^2 \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}^2}{}' \bar{\mathbf{I}}_\gamma}}_{\text{III}}. \tag{16}$$

I will now examine the effects of each of the terms in equation 16 starting from the rightmost one. The third term (III) produces a scalar for each local curvature parameter $\mu_{\vec{x}}$. This scalar multiplies the image for each physical location after it has been differentiated in depth and scaled by $\partial \zeta / \partial \mu_{\overline{x}}$, as described by the second term (II). Notice that the phase introduced by the depth derivative of the image in (II) is crucial for the successful backprojection into the slowness model that is accomplished by the first term (I). In this term, first $\boldsymbol{\Gamma}'$ transforms the image from the aperture-angle domain into the subsurface-offset domain. The transformed image is scaled, horizontally shifted, and spread across the shot axis by the application of $\bar{\mathbf{P}}'_s$ and $\bar{\mathbf{P}}'_g$. Finally, the adjoint of operators $\frac{\partial \mathbf{P}_s}{\partial \mathbf{s}}{}'$ and $\frac{\partial \mathbf{P}_g}{\partial \mathbf{s}}{}'$ backproject the image perturbations into the slowness model along the source wavepaths and the receiver wavepaths, respectively.

Computational cost

The computational cost and storage overheads for evaluating terms II and III in the gradient expression 16 are limited because only operations on the prestack image are required. On the contrary, the computation of term I is computationally more demanding. It requires the forward propagation and backward propagation of wavefields. The application of $\bar{\mathbf{P}}'_s$ and $\bar{\mathbf{P}}'_g$ requires the storage, and retrieval, of the source wavefield and receiver wavefield. Furthermore, to apply $\frac{\partial \mathbf{P}_s}{\partial \mathbf{s}}{}'$ and $\frac{\partial \mathbf{P}_g}{\partial \mathbf{s}}{}'$ we need to correlate the source and receiver wavefields with the wavefields generated by the image derivatives. In summary, the computational cost of one gradient computation of the proposed method can be roughly estimated to be twice the computational cost of one gradient computation of a full-waveform inversion algorithm. The factor of two occurs because two propagations are needed to backproject the image perturbations into the slowness model along both the source wavepaths and the receiver wavepaths.

The data-handling task could be simplified if the frequency-domain downward-continuation migration is used instead of reverse-time migration, because computations can be performed independently for each frequency. The adaptation of the theory presented in this paper to

downward-continuation migration is fairly straightforward. It would only require to exchange expressions 1 and A-4 with the corresponding frequency-domain expressions.

CONCLUSIONS

The theoretical framework I presented in Biondi (2010) can be extended from transmission-tomography problems to MVA problems. The computational cost of the proposed method can be high, though the cost of one iteration is comparable with the cost of one iteration of full waveform inversion. Numerical tests of the gradient operator and of the complete velocity-estimation method are needed.

ACKNOWLEDGMENTS

I would like to thank Yang Zhang for useful suggestions to make my mathematical notation a little bit easier to follow.

REFERENCES

Almomin, A. and Y. Tang, 2010, Migration velocity analysis based on linearization of the two-way wave equation: SEP-Report, **142**, X–Y.

Biondi, B., 2006, 3D Seismic Imaging: Society of Exploration Geophysicists.

———, 2010, Wave-equation tomography by beam focusing: SEP-Report, **140**, 23–38.

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

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

Chavent, G. and C. A. Jacewitz, 1995, Determination of background velocities by multiple migration fitting: Geophysics, **60**, 476–490.

Rickett, J. and P. Sava, 2002, Offset and angle-domain common image-point gathers for shot-profile migration: Geophysics, **67**, 883–889.

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

Sava, P. and B. Biondi, 2004a, Wave-equation migration velocity analysis—I: Theory: Geophysical Prospecting, **52**, 593–623.

———, 2004b, Wave-equation migration velocity analysis—II: Examples: Geophysical Prospecting, **52**, 607–623.

Sava, P. and S. Fomel, 2003, Angle-domain common-image gathers by wavefield continuation methods: Geophysics, **68**, 1065–1074.

Symes, W. W., 2008, Migration velocity analysis and waveform inversion: Geophysical Prospecting, **55**, 765–804.

APPENDIX A

DETAILS OF GRADIENT COMPUTATION

In this appendix I present the analytical development needed to compute all the terms in equation 16. Equations 14 and 15 provide the expression for computing the derivatives of the moveout parameters with respect to slowness as:

$$\left.\frac{\partial \boldsymbol{\mu}_{\vec{x}}}{\partial \mathbf{s}}\right|_{\mu_{\overline{x}}=\bar{\mu}_{\vec{x}}} = -\frac{\frac{\partial \dot{J}_{\mathrm{F}}(\boldsymbol{\mu}_{\vec{x}})}{\partial \mathbf{s}}}{\frac{\partial \dot{J}_{\mathrm{F}}(\boldsymbol{\mu}_{\vec{x}})}{\partial \boldsymbol{\mu}_{\vec{x}}}} = -\frac{\left\langle \left.\frac{\partial \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}}\right|_{\boldsymbol{\mu}_{\vec{x}}=\bar{\boldsymbol{\mu}}_{\vec{x}}}, \frac{\partial \mathbf{I}_\gamma}{\partial \mathbf{s}} \right\rangle_\gamma}{\left\langle \left.\frac{\partial^2 \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}^2}\right|_{\boldsymbol{\mu}_{\vec{x}}=\bar{\boldsymbol{\mu}}_{\vec{x}}}, \mathbf{I}_\gamma \right\rangle_\gamma}, \tag{A-1}$$

where the elements of the matrix $\frac{\partial \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}}$ are computed using either equation 9 or equation 11, and the elements of the matrix $\frac{\partial^2 \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}^2}$ are given by

$$\frac{\partial^2 \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}^2}\left(\vec{x},\gamma,\mu_{\vec{x}}\right) = \mathcal{M}_\gamma\left[\boldsymbol{\mu}_{\vec{x}}\right]\dot{I}_\gamma\left(\vec{x},\gamma;\bar{\mathbf{s}}\right)\frac{\partial^2 \zeta}{\partial \mu_{\overline{x}}^2} + \mathcal{M}_\gamma\left[\boldsymbol{\mu}_{\vec{x}}\right]\ddot{I}_\gamma\left(\vec{x},\gamma;\bar{\mathbf{s}}\right)\frac{\partial \zeta}{\partial \mu_{\overline{x}}}.$$

In this last expression $\ddot{I}_\gamma\left(\vec{x},\gamma;\bar{\mathbf{s}}\right) = \partial^2 I_\gamma\left(\vec{x},\gamma;\bar{\mathbf{s}}\right)/\partial z^2$. Given the moveout parametrization expressed in 7, $\partial^2\zeta/\partial\mu_{\overline{x}}^2 = 0$ and the previous expression simplifies into the following:

$$\frac{\partial^2 \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}^2} = \mathcal{M}_\gamma\left[\boldsymbol{\mu}_{\vec{x}}\right]\ddot{I}_\gamma\left(\vec{x},\gamma;\bar{s}\right)\frac{\partial \zeta}{\partial \mu_{\overline{x}}}. \tag{A-2}$$

Furthermore, when $\bar{\boldsymbol{\mu}}_{\vec{x}} = 0$, equation A-2 further simplifies into:

$$\frac{\partial^2 \mathcal{M}_\gamma}{\partial \boldsymbol{\mu}_{\vec{x}}^2} = \ddot{I}_\gamma\left(\vec{x},\gamma;\bar{s}\right)\frac{\partial \zeta}{\partial \mu_{\overline{x}}}. \tag{A-3}$$

The derivative of the image vector with respect to slowness, $\partial \mathbf{I}_\gamma/\partial \mathbf{s}$ are evaluated by applying the conventional wave-equation tomography operator that links perturbations in the slowness model to perturbations in the propagated wavefields by a first-order Born linearization of the wave equation.

Applying the chain rule to equation 1, and taking into account the offset-to-angle transformation 2, we can write

$$\frac{\partial I_\gamma\left(\vec{x},\gamma;s\right)}{\partial s} = \mathbf{\Gamma}\sum_t\sum_{x_s}\left[\bar{P}_g\left(t,\vec{x}-\vec{x_h},x_s\right)\frac{\partial P_s\left(t,\vec{x}+\vec{x_h},x_s\right)}{\partial s} + \bar{P}_s\left(t,\vec{x}-\vec{x_h},x_s\right)\frac{\partial P_g\left(t,\vec{x}-\vec{x_h},x_s\right)}{\partial s}\right], \tag{A-4}$$

where the wavefields $\bar{P}_s$ and $\bar{P}_g$ are computed with the background slowness, and the wavefield derivatives $\partial P_s/\partial s$ and $\partial P_g/\partial s$ are computed by the conventional adjoint-state methodology that is at the basis of full waveform inversion and wave-equation tomography.

In more compact matrix notation the previous expression can be written as

$$\frac{\partial \mathbf{I}_\gamma}{\partial \mathbf{s}} = \mathbf{\Gamma} \left(\bar{\mathbf{P}}_g \frac{\partial \mathbf{P}_s}{\partial \mathbf{s}} + \bar{\mathbf{P}}_s \frac{\partial \mathbf{P}_g}{\partial \mathbf{s}} \right), \tag{A-5}$$

where the matrices $\bar{\mathbf{P}}_s$ and $\bar{\mathbf{P}}_g$ are composed of the wavefields for every source and depth level, and properly shifted in space by the subsurface offset. For the computation of the gradient, we need to apply the adjoint operator that is:

$$\frac{\partial \mathbf{I}_\gamma}{\partial \mathbf{s}}' = \left(\frac{\partial \mathbf{P}_s}{\partial \mathbf{s}}' \bar{\mathbf{P}}'_g + \frac{\partial \mathbf{P}_g}{\partial \mathbf{s}}' \bar{\mathbf{P}}'_s \right) \mathbf{\Gamma}'. \tag{A-6}$$

Almomin and Tang (2010) present an equivalent, but different, derivation of an algorithm to compute the application of the operator $\frac{\partial \mathbf{I}_\gamma}{\partial \mathbf{s}}$, (or its adjoint) to a vector of slowness perturbations (or image perturbations).

Seismic reservoir monitoring with encoded permanent seismic arrays

Gboyega Ayeni

ABSTRACT

Hydrocarbon reservoirs can be efficiently monitored with encoded data recorded by permanent seismic arrays. Permanent seismic sources and receivers can yield a vast amount of data that may enable near–real-time monitoring. I propose an encoding approach that may overcome some of the operational, storage and processing challenges posed by these vast data volumes. Although data encoding introduces cross-talk artifacts, permanent arrays allow for good repeatability of such artifacts, thereby aiding time-lapse seismic cross-equalization. Because the proposed method utilize low-energy intermittent seismic sweeps, data must be recorded for longer durations compared to conventional data recording. Direct migration of these long-duration data is efficient and gives good-quality time-lapse images. Using a 2D numerical model, I show that this method can produce reliable time-lapse images of comparable quality to those from conventional seismic sources.

INTRODUCTION

Time-lapse seismic reservoir monitoring is an established technology. By repeating the seismic experiment over an evolving reservoir, changes in reservoir properties can be estimated from seismic amplitude and travel-time changes. Many successful case studies demonstrate the technical considerations and business impact of time-lapse seismic (Rickett and Lumley, 2001; Whitcombe et al., 2004; Ebaid et al., 2009).

By enabling seismic recordings at small time intervals, permanent seismic arrays can make near–real-time reservoir monitoring possible. Lumley (2001, 2004) discusses important business and technical drivers for permanent seismic arrays. Several field experiments have been published (Meunier et al., 2001; Smit et al., 2005; Forgues et al., 2006). Because permanent arrays do not suffer from positioning errors, seismic experiments can be repeated with high accuracy. However, in addition to the high operation and storage costs of the recorded data volumes, conventional processing cost of the recorded data can be high. Although, under certain conditions, simple (e.g. NMO) processing can give satisfactory results (Forgues et al., 2006), such methods are inadequate in many geological environments. In this paper, I show that direct wave-equation migration of encoded data sets from such permanent recording systems can provide high-quality time-lapse images at relatively low cost.

Encoded seismic data recording with permanent seismic arrays straddles conventional and passive data recording. Although our understanding of passive data imaging has improved over the past decade, several limitations still exist. Direct imaging of passive data suffers several pitfalls (Artman, 2006), and interferometric Green's function retrieval is

computationally expensive (de Ridder, 2009). In many scenarios, seismic reservoir monitoring with interferometric Green's function from surface passive seismic arrays is difficult (Lu et al., 2009). However, reservoir monitoring with active (virtual) source and ambient-noise interferometric Green's functions have been shown for borehole sensors (Bakulin and Calvert, 2004; Lu et al., 2009). Furthermore, it has been demonstrated that interferometric Green's functions from borehole systems may give satisfactory time-lapse responses in the well vicinity (Bakulin and Calvert, 2004; Lu et al., 2009). The proposed recording approach may overcome some of the current limitations in reservoir monitoring with pure passive data or well-bore virtual source methods.

Although encoded seismic recording is not new (Womack et al., 1990), recent advances in acquisition and processing technology have increased interest in the subject (Hampson et al., 2008; Beasley, 2008; Berkhout et al., 2008; Howe et al., 2009). Direct imaging of such encoded data is possible but suffers from *cross-talk* between data sets from different shots (Romero et al., 2000; Artman, 2006). To directly image field-encoded time-lapse data sets from non-permanent seismic arrays, a linearized inversion method can be used to attenuate artifacts caused by non-repeatable geometry and relative shot delays (Ayeni et al., 2009). Because permanent seismic arrays enable excellent repeatability of the geometry and encoding function, cross-term artifacts are similar between consecutive surveys, and linearized inversion is unnecessary.

To ensure good repeatability over the monitoring period, to limit operational cost, and to limit environmental impact, low-energy, low-footprint seismic sources are desirable. Each source waveform may be a long-duration sweep (Forgues et al., 2006), or intermittent sweeps from an idealized source. By stacking data from several low-energy sources, the signal-to-noise (S/N) ratio is increased and sufficiently high-quality data and images can be obtained. Encoding is important, because it reduces the total recording time for several shots, each requiring a long recording duration. In this paper, it is assumed that these conceptual low-energy sources are randomly and intermittently ignited over a long time period.

Using a phase-encoding migration operator and the relative time-delays between sources, the encoded data are migrated without any separation or interferometric Green's function retrieval. Because all the data are migrated with a baseline velocity model, images from different vintages are not aligned and must be cross-equalized. In this paper, the data are cross-equalized using a cyclic 1D correlation algorithm and an optimized local-matching method (**?**).

First, we give a conceptual description of the proposed data recording and imaging methods. Next, we summarize the cross-equalization methodology that is applied. Finally, using five data sets from a 2D numerical model, we show that the proposed method gives good-quality time-lapse images.

DATA RECORDING AND IMAGING

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

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

where ω is frequency, $m(\mathbf{x})$ is the *reflectivity* at image points $\mathbf{x}$, $f_s(\omega)$ is the source wavelet, and $G(\mathbf{x_s}, \mathbf{x}, \omega)$ and $G(\mathbf{x}, \mathbf{x_r}, \omega)$ are the Green's functions from $\mathbf{x_s}$ to $\mathbf{x}$ and from $\mathbf{x}$ to $\mathbf{x_r}$, respectively. When there are multiple seismic sources, the recorded seismic data is due to a concatenation of phase-shifted sources. For example, the recorded data due to shots starting from $\mathbf{s} = \mathbf{q}$ to $\mathbf{s} = \mathbf{p}$, is given by

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

where $a(\gamma_\mathbf{s})$ is given by

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

and γ_s, the phase-shift function, depends on the delay time t_s.

Randomized intermittent shooting of several shots is equivalent to repetition of equation 2 with a spatially and temporally varying encoding function. The recorded data are similar to passive data, except that all shot positions and timings are known. Therefore, the recording experiment can be regarded as a controlled-source continuous-recording experiment. Figures 1 and 2 show examples of idealized source waveforms at six shot positions. It is assumed that these sources are orders of magnitude weaker than conventional seismic sources and that data from a single sweep is insufficient to create a good-quality image.

Direct imaging of the recorded data, the adjoint to equation 2, is a linear phase-encoding migration operator (Romero et al., 2000). This is equivalent to a single shot-profile migration of the recorded data with an areal source-function derived by concatenating delayed source waveforms from all shot positions.

CROSS-EQUALIZATION

Cross-equalization processing removes unwanted differences between the imaged data sets. Such differences may be caused by uncorrelated noise or geomechanical changes. The post-imaging procedure can be divided into two steps. First, because direct velocity analysis with the recorded data is difficult, we assume that a baseline velocity model built from conventional seismic data will be used to image all data sets. This leads to image misalignment due to reservoir compaction and velocity changes between surveys. To align the monitor and baseline images, a cyclic 1D correlation method is used to estimate vertical and horizontal displacement components (Hale, 2009). The aligned images are then matched using an optimized matching procedure. Match filters estimated in non-reservoir regions are applied to the monitor data on a trace-by-trace basis. These cross-equalization steps are discussed in detail by **?**.

NUMERICAL EXAMPLE

Five data sets were modeled over a modified section of the 2D Marmousi model (Figure 3). The objective is to image seismic amplitude and geomechanical changes around the reservoir using encoded data sets from permanent arrays. Non-linear, discontinuous changes in the reservoir were modeled by velocity changes within the reservoir. Geomechanical changes due to reservoir compaction were modeled by a morphed expanding gaussian anomaly above the reservoir.

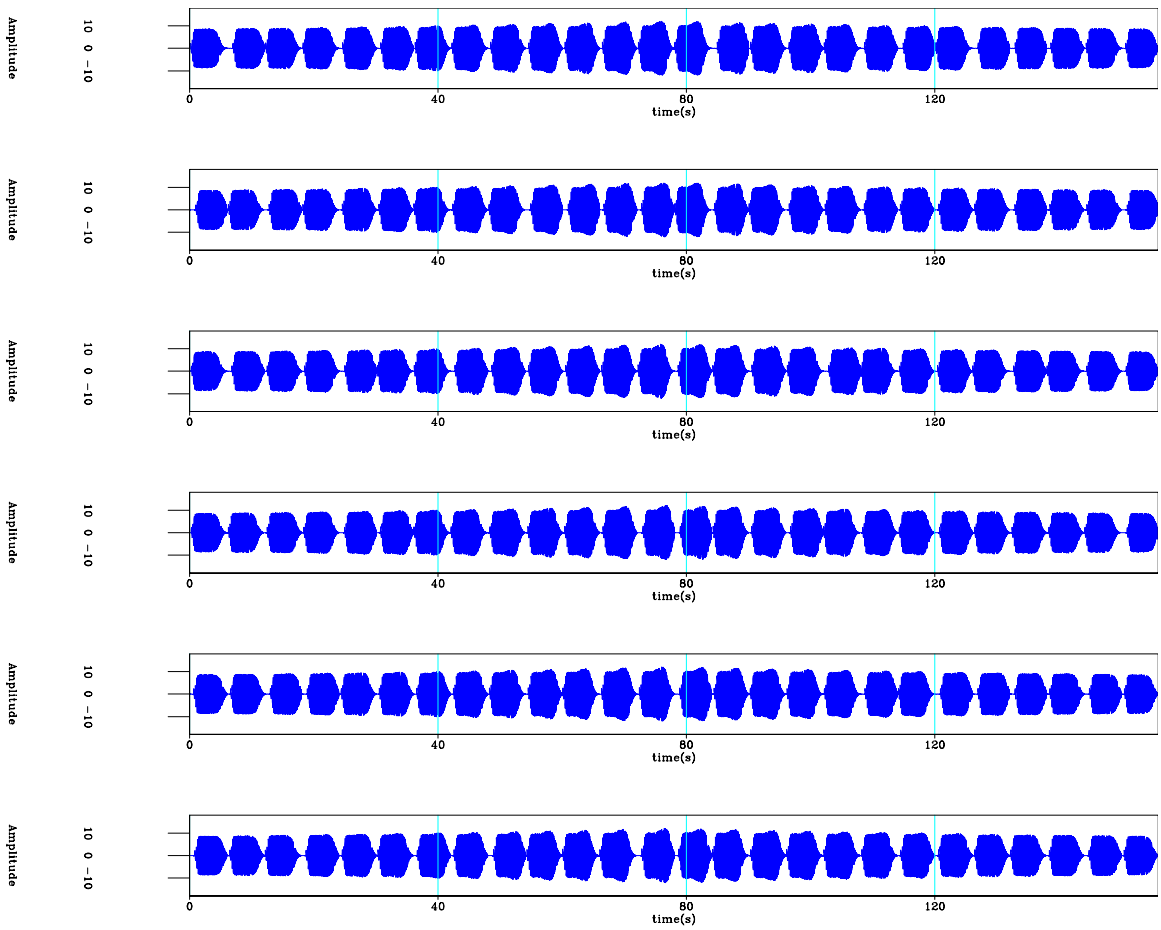


Figure 1: First 150 seconds of the 320 seconds long source waveforms at six source positions. Note that the relative delays between intermittent sweeps from different sources are determined by the encoding function in equation 3. [**ER**]. gayeni1/. perm-src

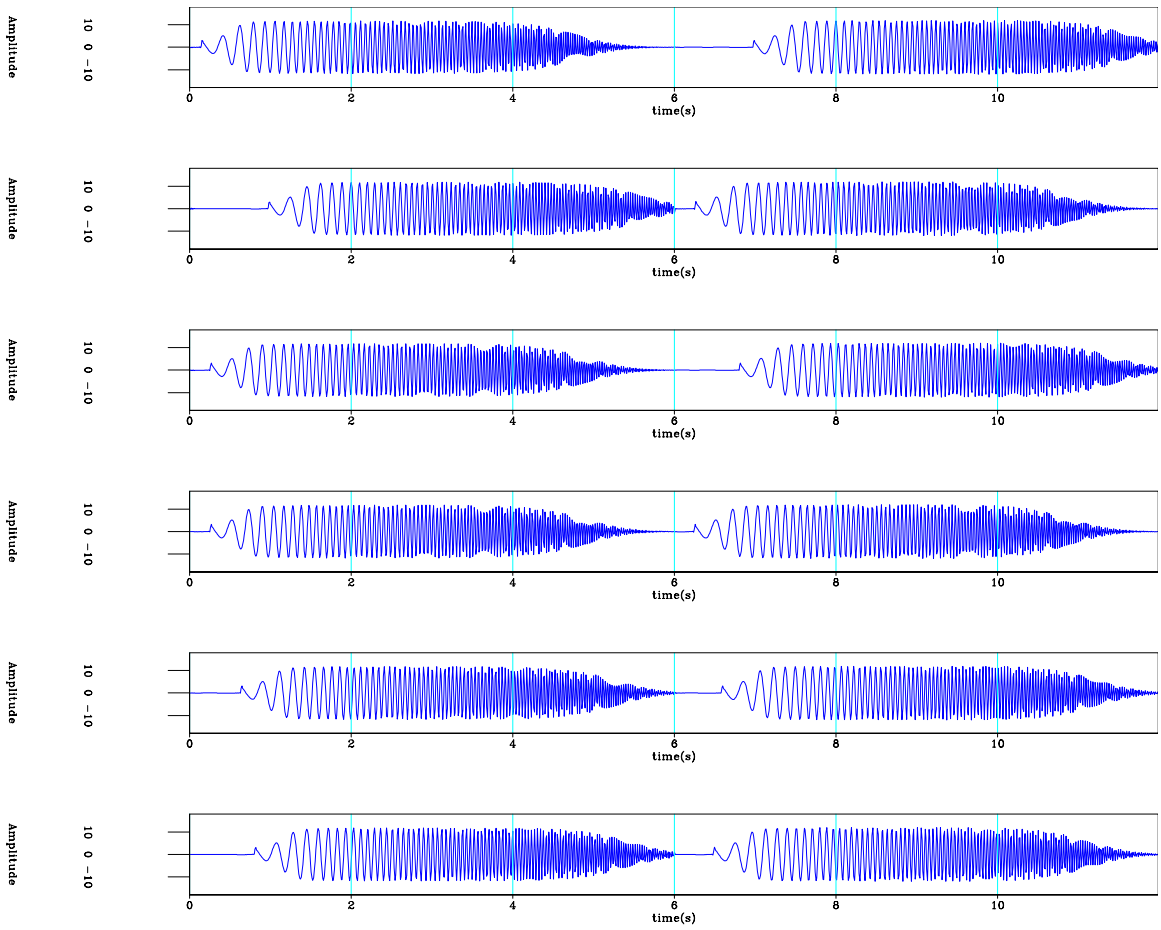


Figure 2: First 12 seconds of the source waveforms shown in Figure 1. Note that there are temporal and spatial differences in the starting times of the seismic sweep at all shot positions. [**ER**]. gayeni1/. perm-src-

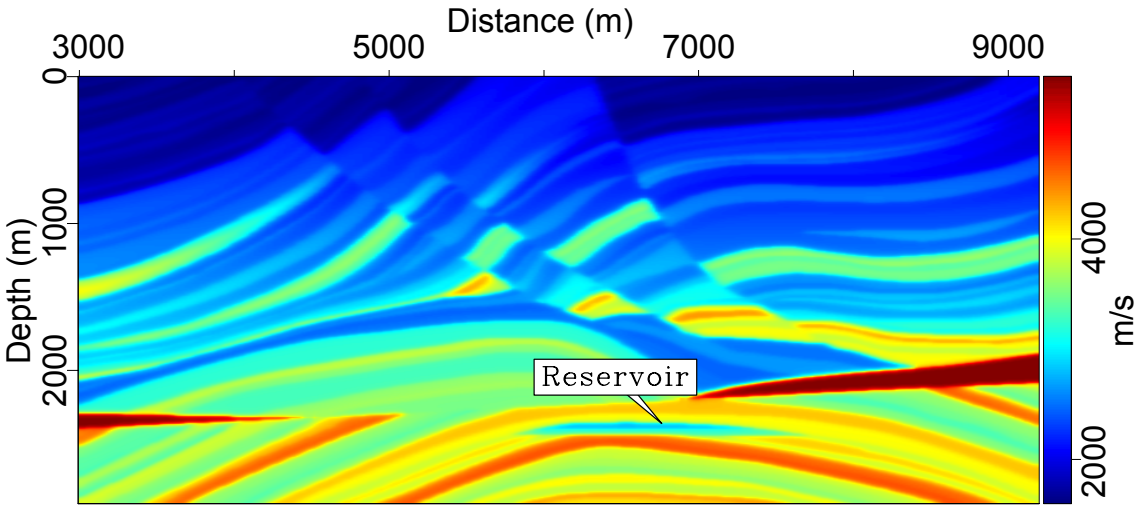


Figure 3: 2D Marmousi velocity model.[**ER**]. gayeni1/. vel-1

Each surface seismic source generates an intermittent sweep within a six-second window with unique random zero- to three-second delays between sources. 320 seconds of data from 388 seismic sources were recorded by surface receivers (Figure 4). Although it was not essential, because this method allows for perfect repeatability, the same encoding function was used for all surveys. However, different amounts of ambient noise (coherent plane-wave and uniform-distribution random noise) were introduced for each survey to produce data at S/N of 4:1. For each survey, using a strongly smoothed version of the baseline modeling velocity, all data were imaged directly with a phase-encoding one-way wave equation algorithm. For comparison, images from noise-free conventional data and processing are shown in Figure 5, those from only 6 seconds of data are shown in Figure 6, and those from the full 320 seconds of data are shown in Figure 7. Note that the longer data records (Figure 7) produce cleaner images than the shorter data records (Figure 6). Note that in the encoded data examples, only a single phase encoded shot-profile migration was required for each survey.

Time-lapse images computed from the migrated images show significant misalignments (Figure 8). Results obtained after alignment and after match filtering are shown in Figures 9 and 10. Note that time-lapse images obtained after matching are comparable to those obtained from the conventional example (Figure 11). In addition, note that apparent displacements—which carry geomechanical information—estimated from the proposed method (Figure 12) and from conventional methods (Figure 13) are similar.

DISCUSSION

Data from permanent encoded low-energy seismic sources have the form of passive data recording (Figure 4). However, because the shot locations and timings are known and can be perfectly repeated, a strong limitation of passive data is eliminated. Furthermore, because the seismic array is permanent, data recording can be repeated perfectly. If these data sets are recorded for long enough, direct imaging of these randomly encoded data can give images of comparable quality to conventional data recording and processing (Figures 5 to 7). The poor resolution of these images in Figure 7 is due to the fact that these data were imaged with the original seismic sweeps without any source designature. These results and it's derivatives can be imporved significantly by first deconvolving the source wavelet before migration. Even then, during imaging wavefield correlations, ambient noise and cross-talk artifacts from different seismic sweeps destructively interfere, whereas the true reflections constructively interfere. Whereas, the images from the short-duration data (Figure 6) are not clean enough to generate reliable time-lapse images, those from the long-duration data (Figure 7) are clean enough to generate interpretable time-lapse images.

It is important that the time-lapse images obtained from this type of recording can provide information similar to that provided by conventional recordings. The time-lapse images computed prior to cross-equalization (Figure 8) are similar to those from conventional data recording and processing (not shown). After cross-equalization, time-lapse images from the proposed method and from conventional methods are similar (Figures 10 and 11). Even when all the data recorded using the proposed method are imaged with a strongly smoothed baseline velocity model, interpretable geomechanical information can still be obtained (Figure 12). These apparent displacement components are similar to those from conventional time-lapse seismic data (Figure 13).

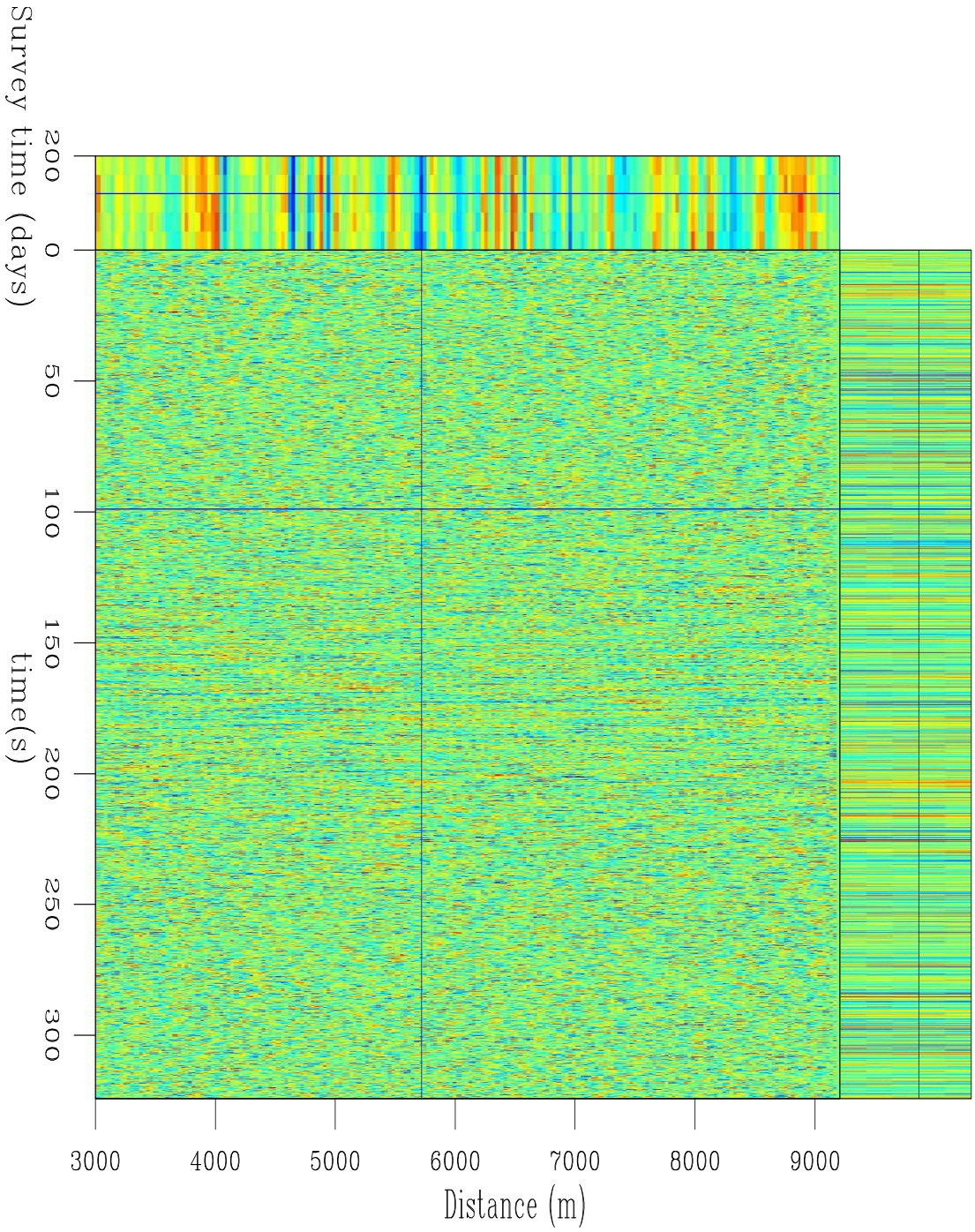


Figure 4: Five seismic data sets from intermittent seismic sweeps over the model in Figure 3. The intersecting lines indicate positions of the three panels within the data volume. [CR] gayeni1/. pdatn288

With this approach, there can be significant cost-savings in both data acquisition and processing. Encoding eliminates the need for long waiting periods that would be otherwise required for low-energy seismic sources that require data stacking. In addition, encoding reduces the data storage requirements. Furthermore, independent of the number of encoded sources, direct imaging can be posed as a single phase-encoding shot-profile migration.

CONCLUSIONS

A method for reservoir monitoring with encoded data from permanent seismic arrays has been proposed. Encoding randomly delayed intermittent sweeps from low-energy sources reduces the total recording time for each seismic experiment. Direct imaging of such data give good-quality images that can be used for near-continuous reservoir monitoring. The numerical experiment shows that seismic amplitude and geomechanical changes from the proposed method are similar to those from conventional seismic data.

REFERENCES

Artman, B., 2006, Imaging passive seismic data: Geophysics, **71**, SI177–SI187.

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.

Bakulin, A. and R. Calvert, 2004, Virtual source: new method for imaging and 4D below complex overburden: SEG Technical Program Expanded Abstracts, **23**, 2477–2480.

Beasley, C. J., 2008, Simultaneous sources: A technology whose time has come: SEG Technical Program Expanded Abstracts, **27**, 2796–2800.

Berkhout, A. J. G., G. Blacquière, and E. Verschuur, 2008, From simultaneous shooting to blended acquisition: SEG Technical Program Expanded Abstracts, **27**, 2831–2838.

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

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.

Forgues, E., J. Meunier, F. Grésillon, C. Hubans, and D. Druesne, 2006, Continuous high-resolution seismic monitoring of SAGD: SEG Technical Program Expanded Abstracts, **25**, 3165–3169.

Hale, D., 2009, A method for estimating apparent displacement vectors from time-lapse seismic images: Geophysics, **74**, V99–V107.

Hampson, G., J. Stefani, and F. Herkenhoff, 2008, Acquisition using simultaneous sources: SEG Technical Program Expanded Abstracts, **27**, 2816–2820.

Howe, D., M. Foster, T. Allen, I. Jack, D. Buddery, A. Choi, R. Abma, T. Manning, and M. Pfister, 2009, Independent simultaneous sweeping in Libya—full scale implementation and new developments: SEG Technical Program Expanded Abstracts, **28**, 109–111.

Lu, R., A. Venkataraman, M. Payne, and J. Zhang, 2009, Application of noise interferometry to obtain time-lapse velocity variations during a steam stimulation cycle at Cold lake: SEG Technical Program Expanded Abstracts, **28**, 1673–1677.

Lumley, D. E., 2001, The next wave in reservoir monitoring: The instrumented oil field: The Leading Edge, **20**, 640–648.

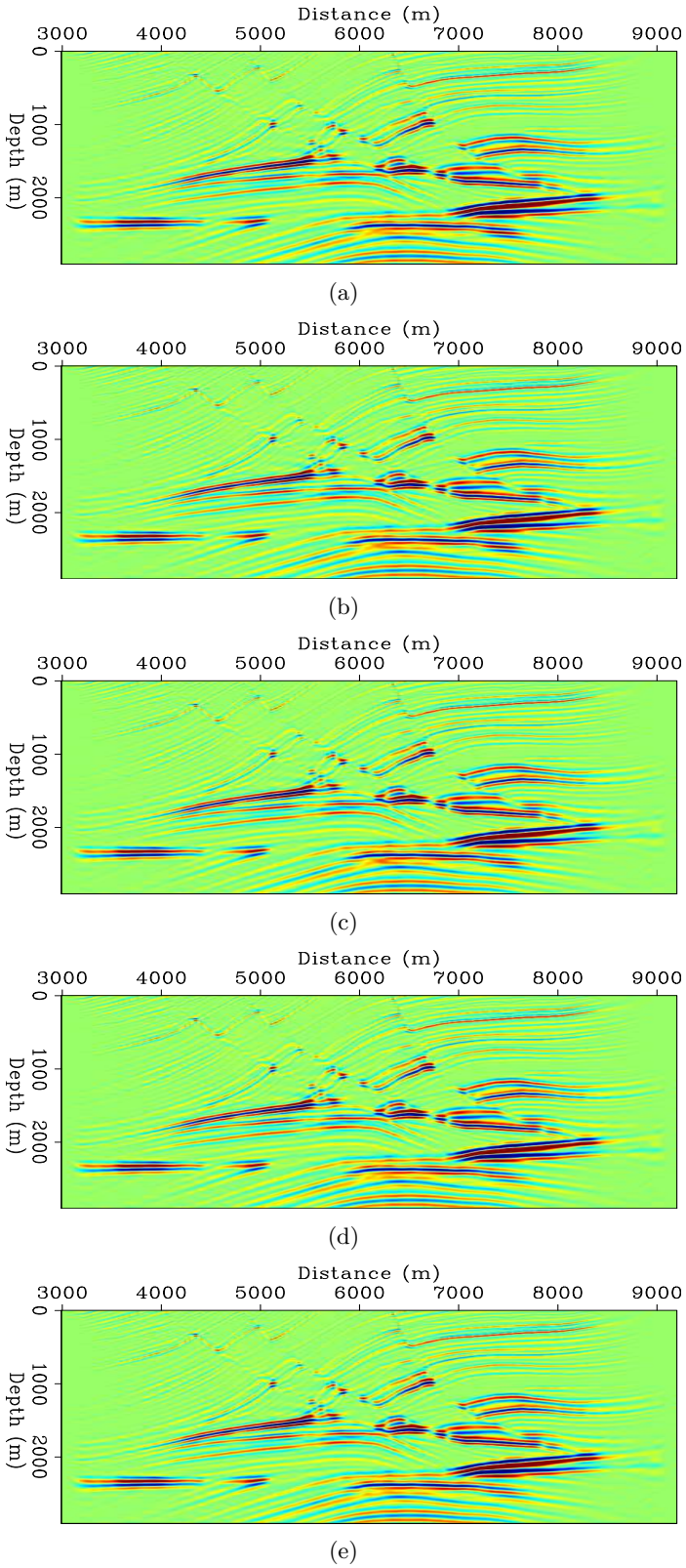


Figure 5: Images from shot-profile imaging of noise-free conventional data over five models modified from Figure 3. [CR] gayeni1/. pmigN1,pmigN2,pmigN3,pmigN4,pmigN5

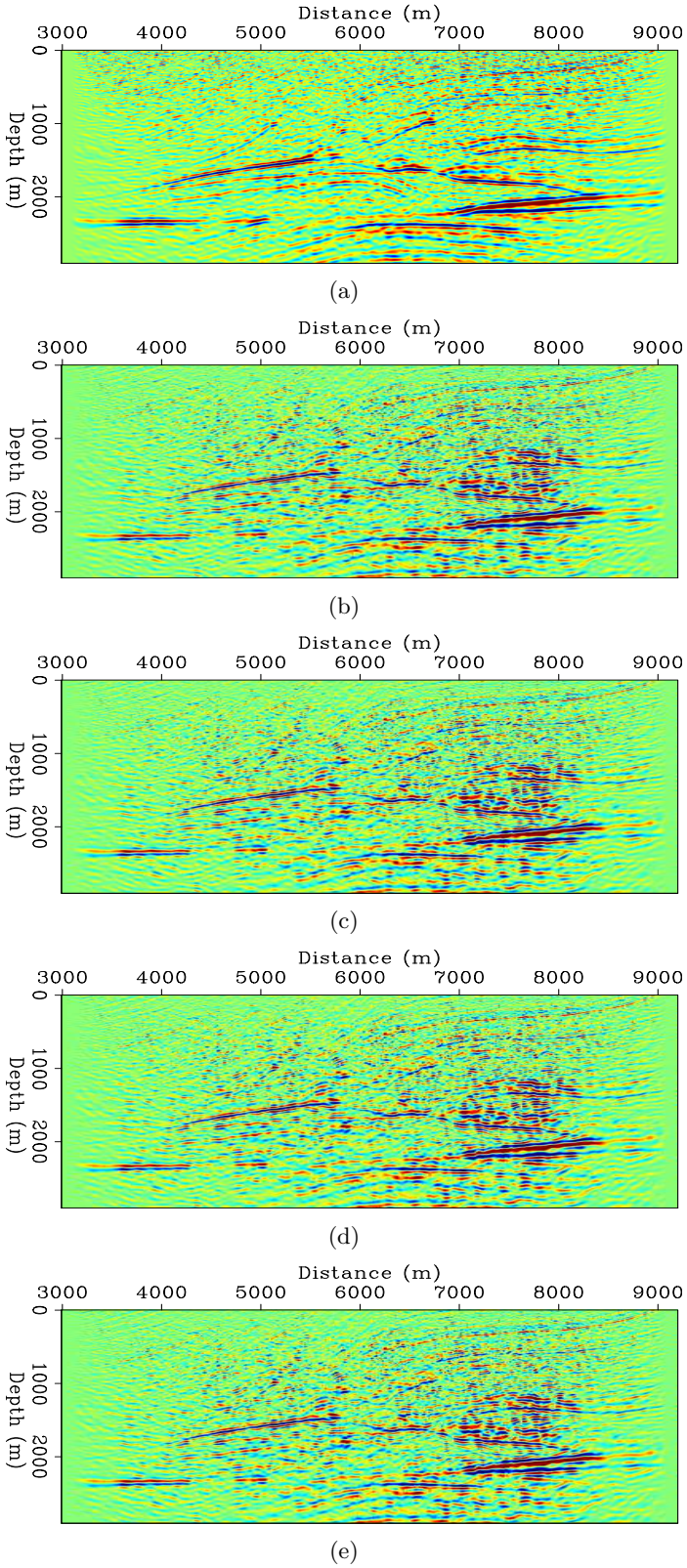


Figure 6: Images from direct imaging of 6 seconds encoded intermittent source data (Figure 4). Note the presence of numerous crosstalk and ambient noise artifacts. [CR] gayeni1/. pmig0n-6-1,pmig2n-6-2,pmig2n-6-3,pmig2n-6-4,pmig2n-6-5

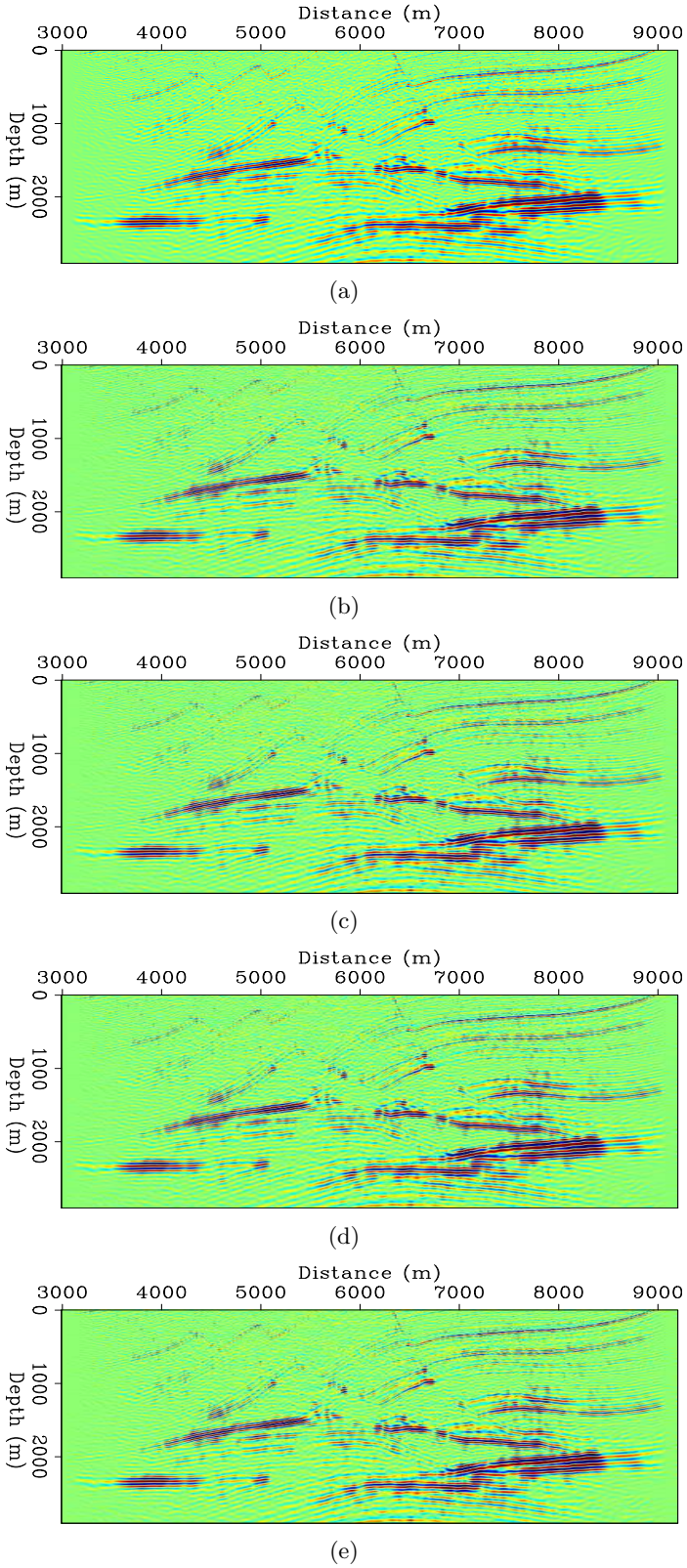


Figure 7: Images from direct imaging of 320 seconds of encoded intermittent sweep data (Figure 4). Note that the crosstalk and ambient noise artifacts in Figure 6 have been attenuated. Note that these images are comparable to those in Figure 5. [CR] gayeni1/. pmig0n-288-1,pmig2n-288-2,pmig2n-288-3,pmig2n-288-4,pmig2n-288-5

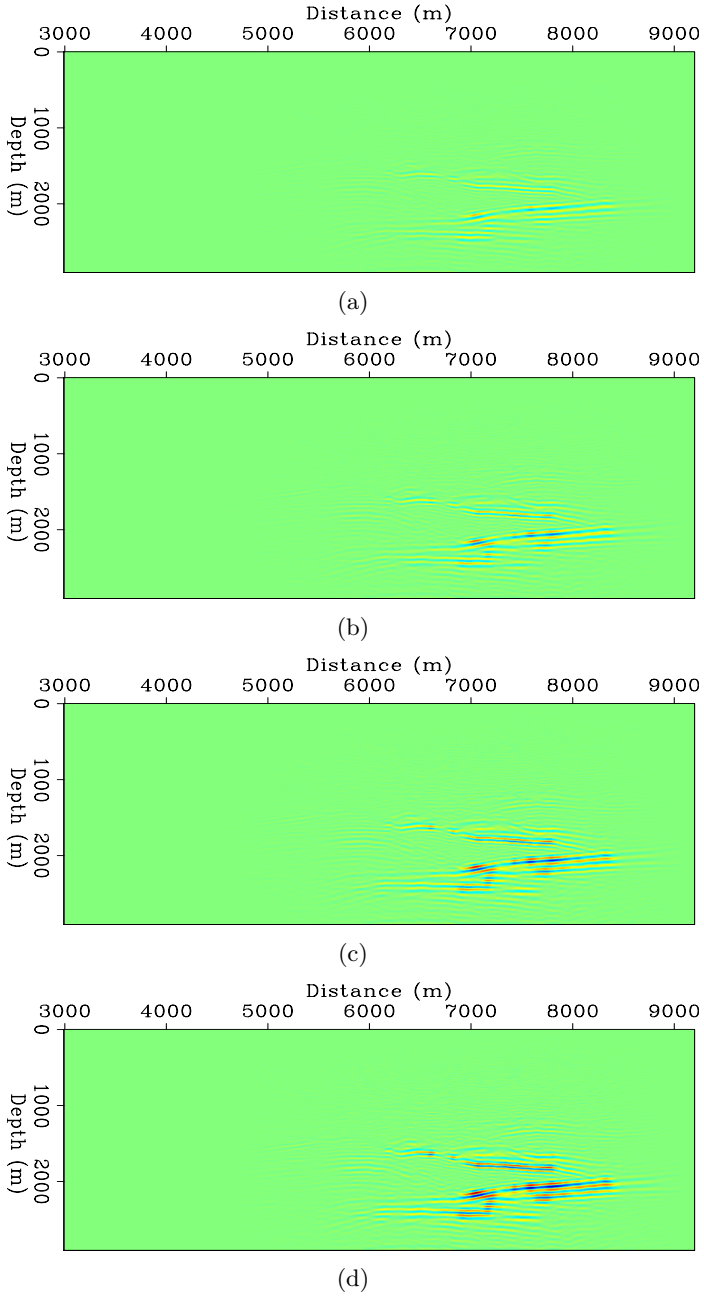


Figure 8: Time-lapse images from direct imaging of encoded intermittent source data (Figure 4). Note that the seismic amplitude changes at the reservoir are masked by the strong misalignment of the images which result from migration with a wrong *baseline* velocity. Reservoir change increases non-linearly from top to bottom. [CR]

gayeni1/. pmig0n-288-4d-2,pmig0n-288-4d-3,pmig0n-288-4d-4,pmig0n-288-4d-5

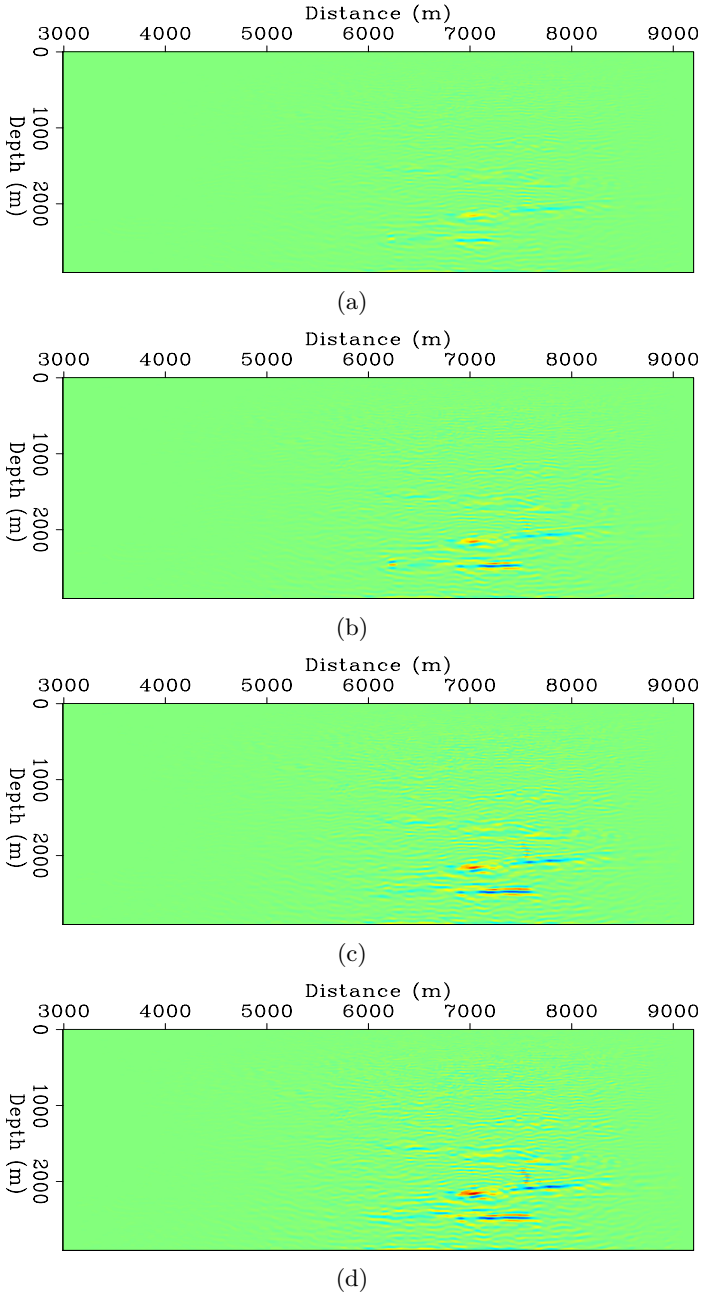


Figure 9: Time-lapse images from direct imaging of encoded intermittent source data (Figure 4) after cyclic 1D warping to remove image misalignments. Note that the seismic amplitude changes at the reservoir are better clearly defined than in Figure 8. However some artifacts (e.g. from the over-hanging salt wedge) persist. [CR] gayeni1/. pmig1n-288-4d-2,pmig1n-288-4d-3,pmig1n-288-4d-4,pmig1n-288-4d-5

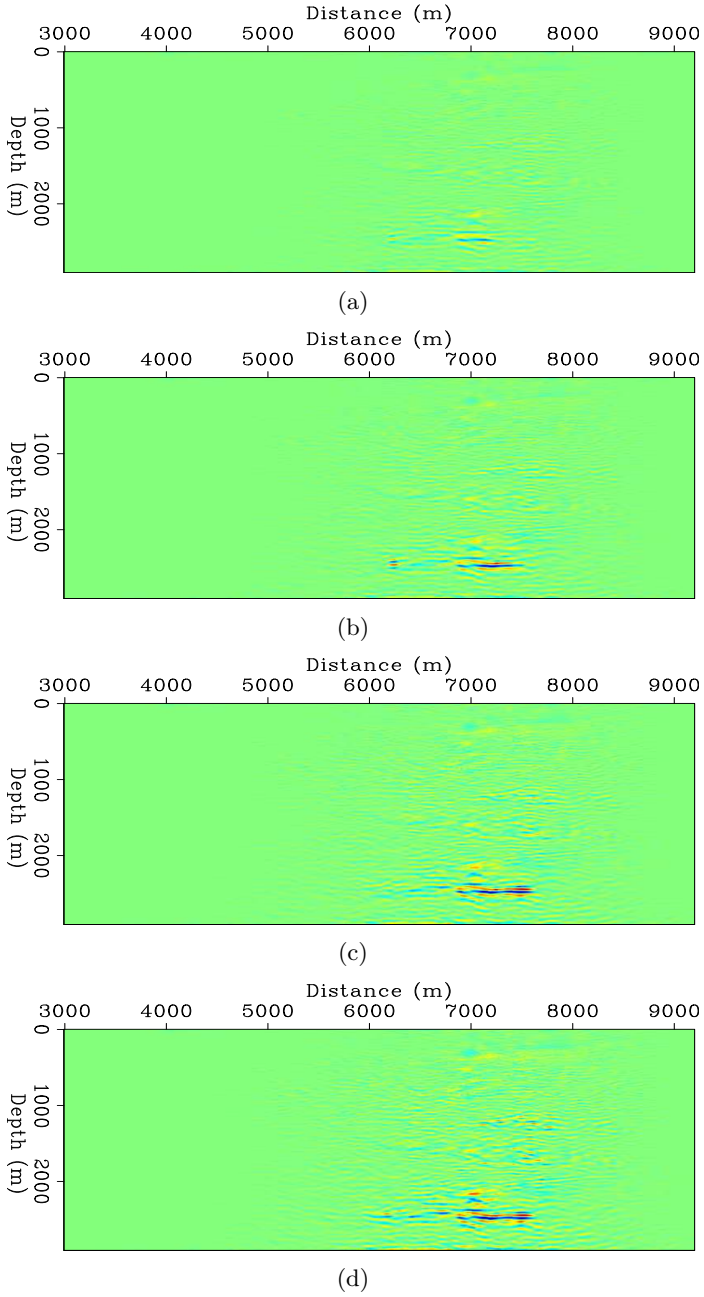


Figure 10: Time-lapse images from direct imaging of encoded intermittent source data (Figure 8) after match-filtering to remove residual artifacts. Compare this to Figures 8 and 9. Note that the seismic amplitude change (and discontinuities) are accurately imaged by the proposed method. [CR] gayeni1/. pmig2n-288-4d-2,pmig2n-288-4d-3,pmig2n-288-4d-4,pmig2n-288-4d-5

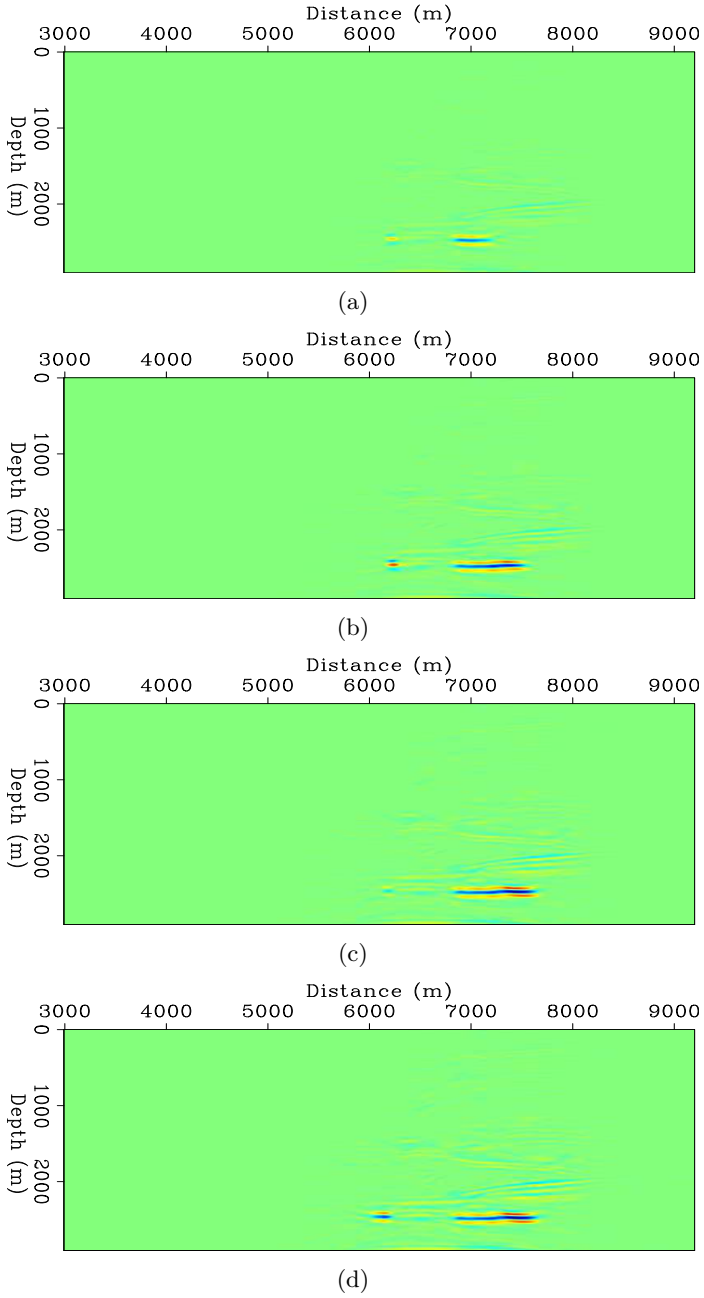


Figure 11: Time-lapse images from from conventional (single-source, high-energy, shot-period) seismic data after warping and match-filtering. These results were obtained using the same models as in the intermittent-source case. Note that these results are similar to those from the proposed method (Figure 10). [CR] gayeni1/. pmigN-4d-2,pmigN-4d-3,pmigN-4d-4,pmigN-4d-5

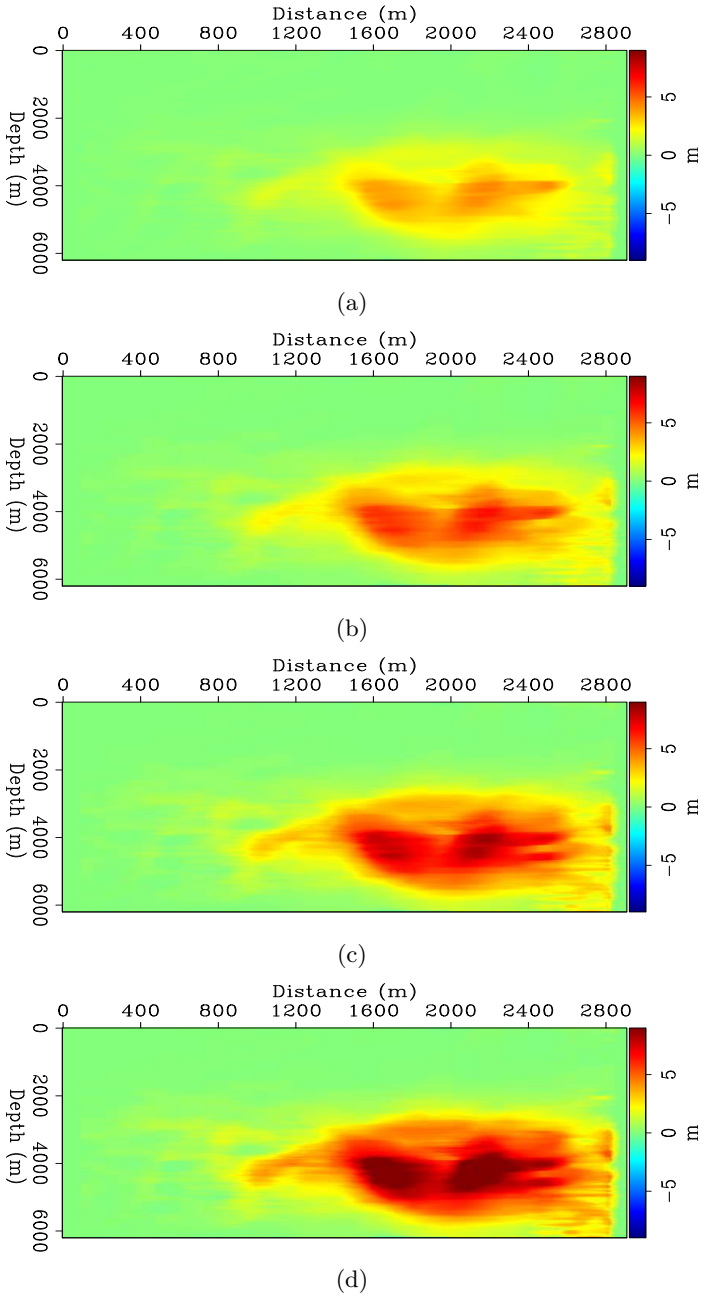


Figure 12: Vertical components of the apparent displacement vectors computed by warping the monitor images in Figure 7 to the Baseline (Figure 7(a)). Note that these are similar to those from conventional data recording/processing (Figure 13). [CR] gayeni1/. ptsn-288-2,ptsn-288-3,ptsn-288-4,ptsn-288-5

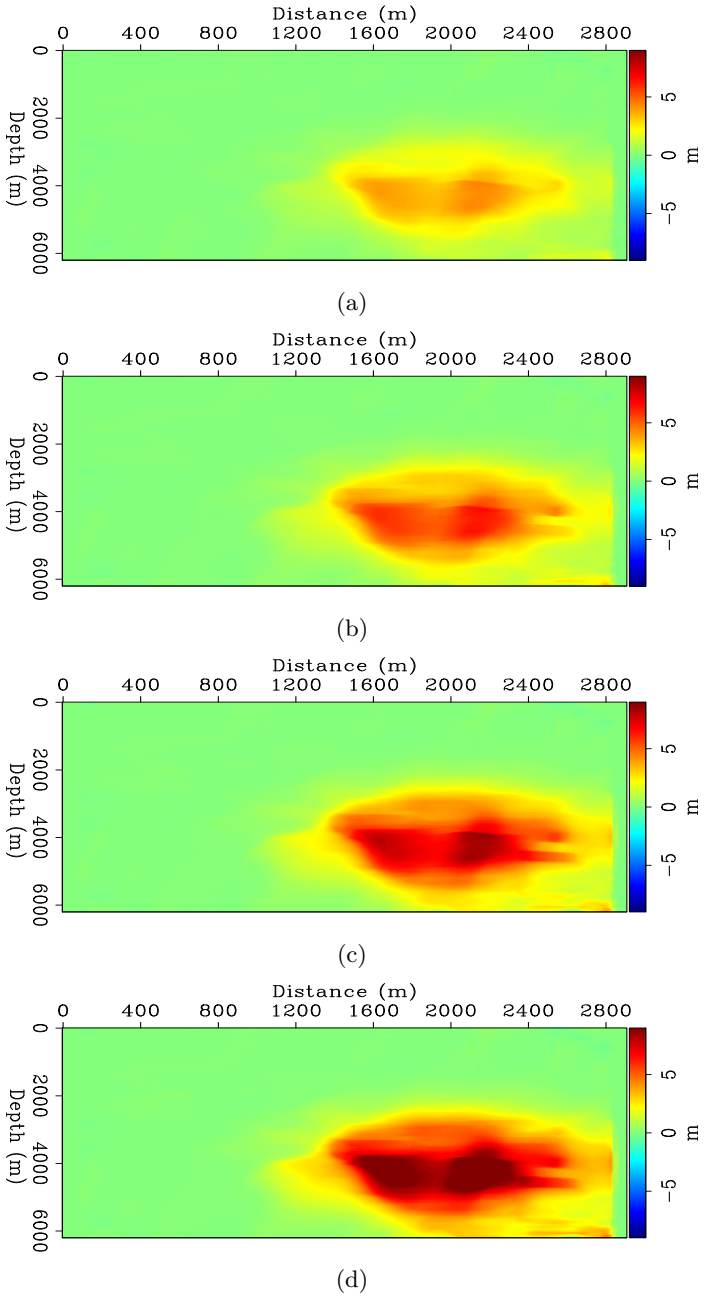


Figure 13: Vertical components of the apparent displacement vectors computed by warping the monitor images in Figure 5 to the Baseline (Figure 5(a)). Note that, in general, these are similar to those from the proposed method (Figure 12). [CR] gayeni1/. ptsn2,ptsn3,ptsn4,ptsn5

———, 2004, Business and technology challenges for 4D seismic reservoir monitoring: The Leading Edge, **23**, 1166–1168.

Meunier, J., F. Huguet, and P. Meynier, 2001, Reservoir monitoring using permanent sources and vertical receiver antennae: The Cere-la-Ronde case study: The Leading Edge, **20**, 622–629.

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

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

Smit, F., M. Ligtendag, P. Wills, and R. Calvert, 2005, Permanent seismic reservoir monitoring using the sparse OBC concept: SEG Technical Program Expanded Abstracts, **24**, 2402–2405.

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

Womack, J. E., J. R. Cruz, H. K. Rigdon, and G. M. Hoover, 1990, Encoding techniques for multiple source point seismic data acquisition: Geophysics, **55**, 1389–1396.

On the separation of simultaneous-source data by inversion

Gboyega Ayeni, Ali Almomin, and Dave Nichols

ABSTRACT

Simultaneous-source data can be adequately separated using an inversion formulation. To recover component shot records, we formulate the data-separation problem as a simultaneous Radon inversion problem. By minimizing the resulting objective function with a robust *hybrid* solver, we obtain high-quality estimates of the component shot records. Furthermore, regularization with directional Laplacians improves the data quality. In our approach, we estimate a single model that predicts all recorded data, and we treat all components of the recorded data as signal. Within limits of operational possibilities, our method can be applied to any number of sources within a single survey and can be easily extended to multiple (time-lapse) surveys. Using 2D sections extracted from the 3D SEAM geophysical model, we show that our method can give results of comparable quality to the original independent shot records.

INTRODUCTION

Conventionally, seismic data acquisition involves a single seismic source and an array of receivers. However, recent advances in acquisition technology enable seismic acquisition with multiple sources (Womack et al., 1990; Hampson et al., 2008; Beasley, 2008). By using simultaneous sources, it is possible to achieve longer offsets, better shot-sampling, and improved time and cost efficiency (van Mastrigt et al., 2002; Berkhout et al., 2008; Howe et al., 2009).

Although direct imaging of simultaneous-source data has several desirable properties, it also suffers from several pitfalls. The most important limitation of direct imaging is the introduction of crosstalk artifacts from incongruous sources. Under certain conditions, crosstalk artifacts may be sufficiently attenuated by stacking (Beasley, 2008; Hampson et al., 2008). Linearized inversion can attenuate crosstalk artifacts significantly (Ayeni et al., 2009; Dai and Schuster, 2009; Tang and Biondi, 2009). However, linearized inversion assumes that the true seismic velocities are known, which is not the case in any practical application. Therefore, most practitioners opt to separate simultaneous-source data sets into independent shot records followed by conventional processing.

Data separation may be treated as a filtering (Moore et al., 2008; Huo et al., 2009) or an inversion (Akerberg et al., 2008; Abma et al., 2010) problem. In this paper, we take an inversion approach, in which components of the simultaneous-source data are predictable from a single model. In our formulation, the simultaneous-source data are modeled by a composite Radon operator based on the recording geometries and relative shot times of the simultaneous sources. We solve the resulting regression using a robust *hybrid*-norm solver (Li et al., 2010). Model sparsity, introduced by the *hybrid*-norm, significantly improves the quality of the recovered data sets relative to those from an l_2 solver.

The quality of the separated data is further improved by introducing model regularization, which may be implemented in either the Radon or the shot space. In this paper, for the single-survey problem, we consider regularization by damping and by directional Laplacians (Hale, 2007). In our problem, non-stationary directional Laplacians are used to enforce smoothness along local dips. First, we solve the inversion problem using a damping regularization. Then, using the estimated independent data, we compute dip-components along constant-offset panels. From these dip estimates, regularization operators for the next inversion step are generated. These operators are used to regularize the inversion and generate new results that serve as inputs to the next inversion step. This procedure can be repeated as many times as necessary.

One potential application of simultaneous-source acquisition is in time-lapse seismic reservoir monitoring (Ayeni et al., 2009). For example, because this method reduces seismic acquisition cost, monitoring data sets can be acquired at shorter time intervals. However, because time-lapse monitoring requires high-quality data, amplitudes of separated data must be reliable. For the time-lapse seismic problem, we consider a spatio-temporal regularization scheme that utilizes a combination of directional Laplacians and temporal smoothness constraints.

In this paper, we first describe the inversion formulation of our separation approach. Next, we briefly discuss possible regularization schemes for this inversion problem. Finally, using data sets from 2D sections extracted from the SEAM geophysical model, we show that our method can produce high-quality results for both single and time-lapse surveys.

METHOD

We can represent the simultaneous-source acquisition process for n sources as follows:

$$\sum_{i=1}^{n} \mathbf{S}\mathbf{d}_i = \mathbf{d}, \tag{1}$$

where $\mathbf{S}$ is a shifting operator built from the relative time-delays between sources, $\mathbf{d}_i$ is the data due to source i, and $\mathbf{d}$ is the simultaneous-source data. We can rewrite this equation in the form

$$\top \mathbf{H}_i \mathbf{m} \approx \mathbf{d}, \tag{2}$$

where $\top$ is the summation operator, and $\mathbf{H}_i$ is an operator that models data $\mathbf{d}_i$ from model $\mathbf{m}$. Note here, that all component shots of the encoded data $\mathbf{d}$ are modeled from a single, consistent model $\mathbf{m}$. In this paper, $\mathbf{H}_i$ is a modified hyperbolic Radon operator that maps data from the the velocity-CMP space to shot-offset space, honoring the time delays at source i relative to a reference shot. Adding a regularization operator $\mathbf{A}$, we have

$$\begin{aligned} \top \mathbf{H}_i \mathbf{m} &\approx \mathbf{d}, \\ \epsilon \mathbf{A} \mathbf{m} &\approx \mathbf{0}, \end{aligned} \tag{3}$$

where ϵ, regularization parameter determines the regularization strength.

There are many possible choices for the regularization operator $\mathbf{A}$. Taking $\mathbf{A}$ to be an identity matrix and minimizing the regressions in equation 3 with a *hybrid* solver leads

to a sparse Radon inversion problem. Alternatively, we can regularize the problem with a shot-space operator $\mathbf{B}_i$ by re-writing equation 3 as follows:

$$\begin{aligned} \top\mathbf{H}_i\mathbf{m} &\approx \mathbf{d}, \\ \epsilon\mathbf{B}_i\mathbf{H}_i\mathbf{m} &\approx \mathbf{0}, \end{aligned} \tag{4}$$

which in matrix form can written as

$$\begin{bmatrix} \top \\ \epsilon\mathbf{B}_i \end{bmatrix} \mathbf{H}_i\mathbf{m} \approx \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix}. \tag{5}$$

In this paper, we define $\mathbf{B}$ as a system of non-stationary dip-filters. First, we compute local event dips using the plane-wave destruction method (Fomel, 2002), then we compute dip-filters using factorized directional Laplacians (Hale, 2007). Because of the random delays between simultaneous sources, for any given source, events from other sources are random in its corresponding common-offset gathers. By destroying predictable events corresponding to source i, operator $\mathbf{B}_i$ ensures that only these events are allowed in the final separated data sets, whereas unpredictable events are not. Events that are not predictable by $\mathbf{B}_i$ are passed on to other sources, where they are predictable by the corresponding operator $\mathbf{B}_j$. We call this inversion method dip-constrained sparse inversion (DCSI). In this paper, we refer to solution of equation 5, with $\mathbf{B}_i$ as an identity matrix, as unconstrained sparse inversion.

However, because the operator $\mathbf{B}_i$ is a function of the separated data, the problem becomes non-linear. To linearize this problem, we start by solving the equation 3 to get an initial estimate for $\mathbf{d}_i$. Then, using $\mathbf{d}_i$, we obtain an estimate of the operator $\mathbf{B}_i$, which is used to regularize the problem starting from initial model estimate $\mathbf{m}$. Results from this new step can then serve as inputs into the next inversion step. This process can be repeated as as many times as necessary.

Following the approach of Abma et al. (2010), instead of using $\mathbf{B}_i$ as a regularization operator, we can use $\mathbf{B}_i^{-1}$ as a smoothing operator by re-writing equation 4 as follows:

$$\begin{aligned} \top\mathbf{B}_i^{-1}\mathbf{H}_i\mathbf{m} &\approx \mathbf{d}, \\ \epsilon\mathbf{I}\mathbf{m} &\approx \mathbf{0}. \end{aligned} \tag{6}$$

In this paper, we implement $\mathbf{B}_i^{-1}$ as polynomial division (Claerbout and Fomel, 2008) with non-stationary directional Laplacians.

Equation 5 can be directly extended to multiple surveys. For example, for two surveys, we can minimize the regressions

$$\begin{aligned} \begin{bmatrix} \top_1 \\ \epsilon\mathbf{B}_{i1} \end{bmatrix} \mathbf{H}_{i1}\mathbf{m}_1 &\approx \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{0} \end{bmatrix}, \\ \begin{bmatrix} \top_2 \\ \epsilon\mathbf{B}_{i2} \end{bmatrix} \mathbf{H}_{i2}\mathbf{m}_2 &\approx \begin{bmatrix} \mathbf{d}_2 \\ \mathbf{0} \end{bmatrix}, \\ \begin{bmatrix} \lambda\mathbf{Z}_1\mathbf{m}_1 & -\lambda\mathbf{Z}_2\mathbf{S}_{1,2}\mathbf{m}_2 \end{bmatrix} &\approx \mathbf{0}, \end{aligned} \tag{7}$$

where for survey k, $\mathbf{H}_{ik}$ and $\mathbf{B}_{ik}$ are the modeling and shot-space regularization operators, respectively, for source i; $\mathbf{m}_k$ and $\mathbf{d}_k$ are the Radon model and simultaneous-source data;

$\mathbf{Z}_k$ is a temporal regularization operator; and $\mathbf{S}_{k,k+1}$ is a shifting operator that aligns the models $\mathbf{m}_k$ and $\mathbf{m}_{k+1}$. Note that $\mathbf{H}_{ik}$ incorporates both geometry and relative shot timing for survey k. Because of differences in geometry and relative shot timing between surveys, operator $\mathbf{H}_{i1}$ is different from $\mathbf{H}_{i2}$. The last regression in equation 7 minimizes the difference between models $\mathbf{m}_1$ and $\mathbf{m}_2$. Because we are interested only in production-related differences between $\mathbf{m}_1$ and $\mathbf{m}_2$, the difference between the two models is also very sparse. We can generalize equation 7 to an arbitrary number of surveys as follows:

$$\begin{bmatrix} \top_k \\ \epsilon\mathbf{B}_{ik} \end{bmatrix} \mathbf{H}_{ik}\mathbf{m}_k \approx \begin{bmatrix} \mathbf{d}_k \\ \mathbf{0} \end{bmatrix}, \qquad \begin{bmatrix} \lambda\mathbf{Z}_k\mathbf{m}_k & -\lambda\mathbf{Z}_{k+1}\mathbf{S}_{k,k+1}\mathbf{m}_{k+1} \end{bmatrix} \approx \mathbf{0}. \tag{8}$$

In this paper, we refer to the method of solving the joint-inversion problem represented by equation 8 as spatio-temporal constrained sparse inversion (STCSI).

EXAMPLES

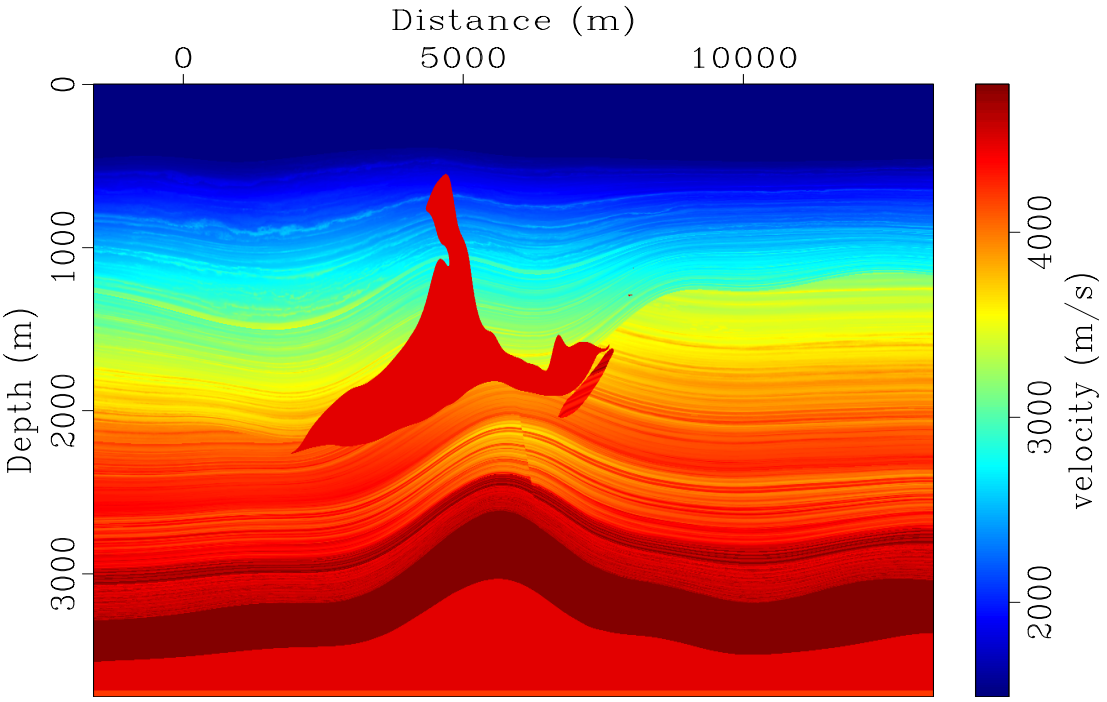
Using 2D models extracted from the 3D SEAM geophysical model (Figure 1), we consider data separation for three possible simultaneous-source applications. The grids in both models have been modified. All data were modeled with a 2D Acoustic finite-difference algorithm. In each example, we use the different formulations to solve the separation problem described above. All sparse inversion examples are generated using a *hybrid* solver (Li et al., 2010).

Example 1: Separation of complex data sets

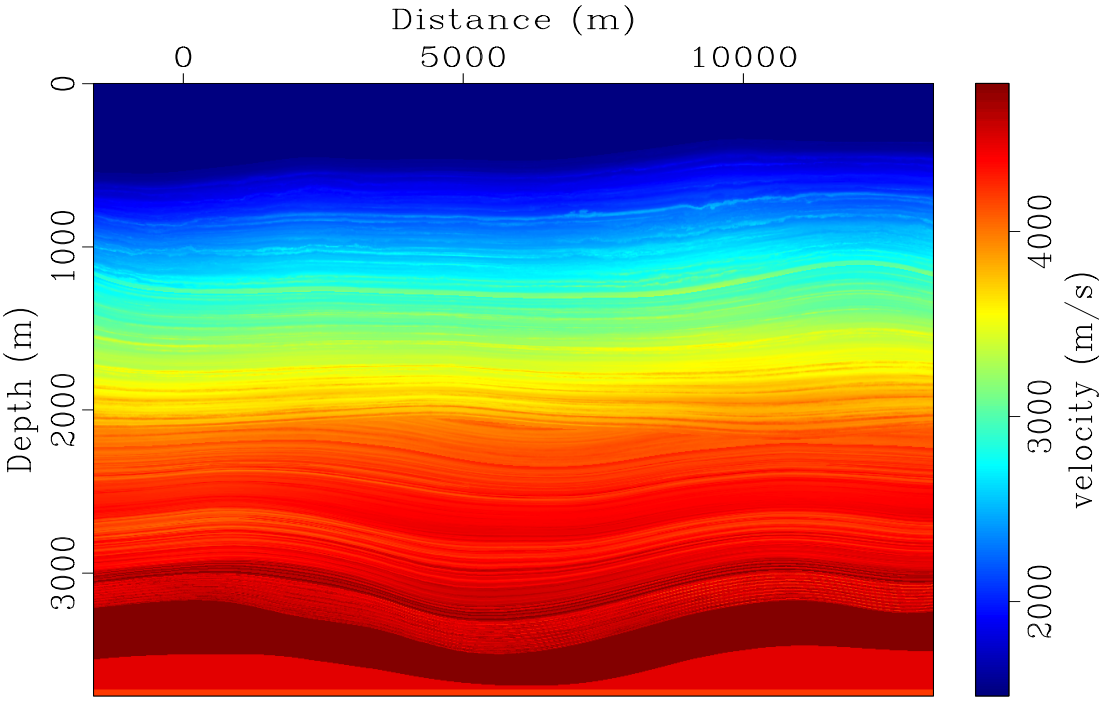
In this example, we consider data from a complex 2D salt model (Figure 1(a)). The simultaneous-source data comprise 330 shot gathers from two sources separated by 2400 m (Figure 2). This example represents the case where a front seismic vessel is pulling the streamer, and a second boat shoots from the end of the streamer cable— with both sources moving from left to right over the model in Figure 1(a). Note the randomness of data corresponding to the unaligned source in the common-offset plane. The single-source records are shown in Figure 3. The separated data sets recovered by l_2 (conjugate-gradient) inversion are shown in Figure 4. Comparing these results to the single-source data (Figure 3), we see that there are numerous *crosstalk* artifacts in each of the two sources. Separation results obtained by sparse inversion of the data without and with regularization by directional Laplacians are shown in Figures 5 and 6, respectively. Note that in both inversion results, the data are well separated into the component shot records. The residual artifacts present in the unconstrained sparse-inversion results (Figure 5) have been attenuated by regularization (Figure 6). Dips estimated from the unconstrained results (Figure 5) and used to obtain the dip-constrained results (Figure 6) are shown in Figure 7.

Example 2: Separation of multiple sources

In this example, we consider data from the model in Figure 1(b). The simultaneous-source data comprise of data from four sources with 800 m separation (Figure 8). The individual



(a)



(b)

Figure 1: 2D velocity models extracted from the 3D SEAM geophysical model. [**ER**] gayeni2/. vel-seam2,vel-seam3

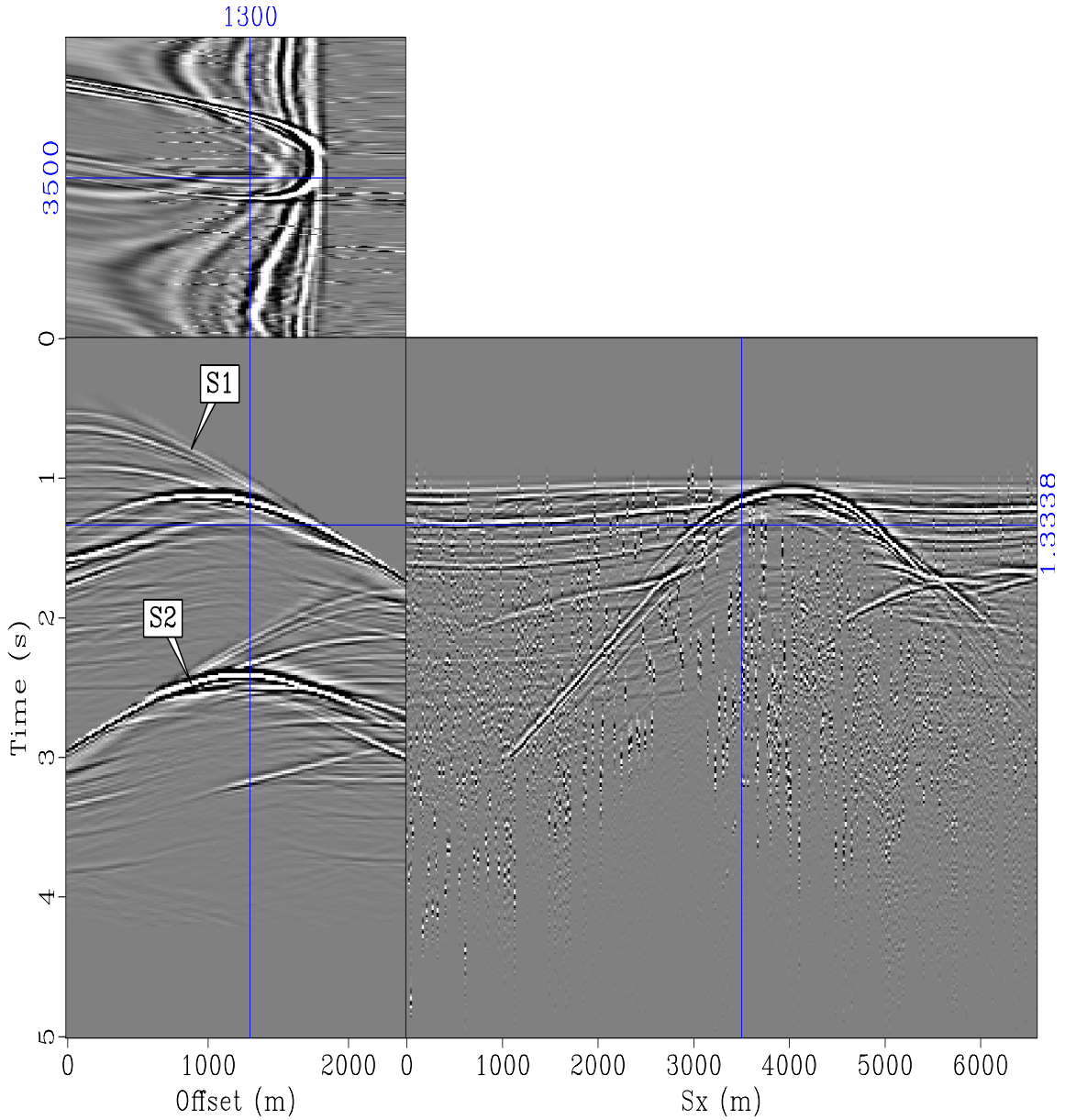
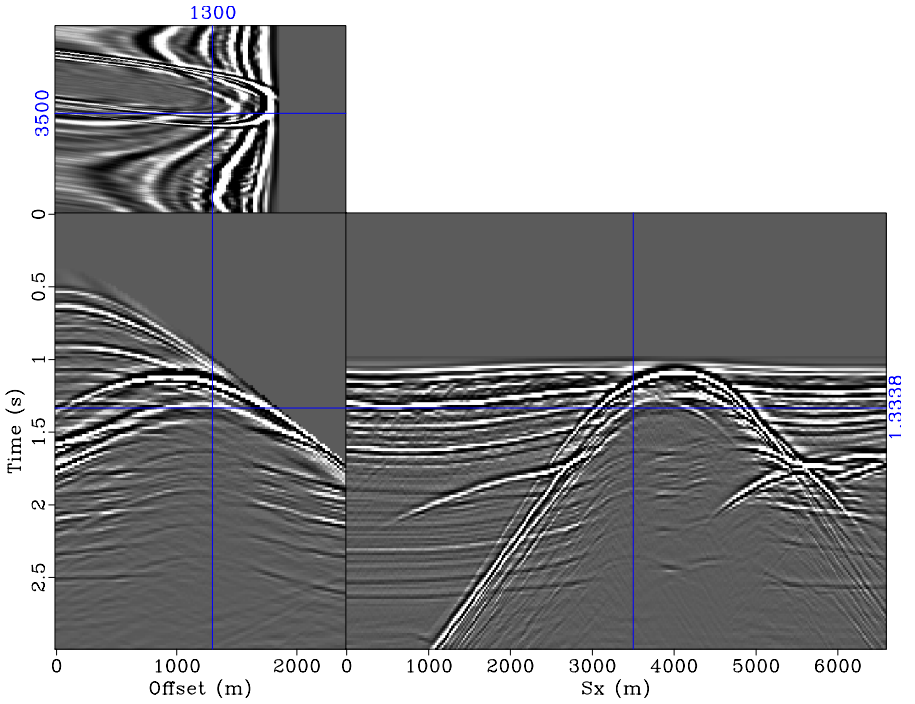
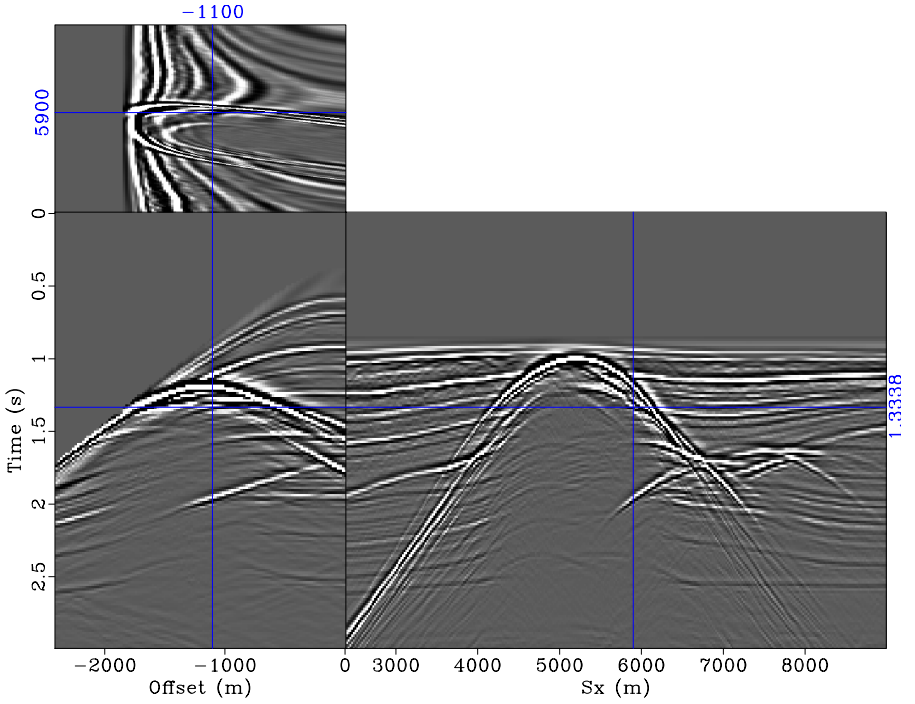


Figure 2: Simultaneous-source data comprising shot-records from two end-on sources (S1 and S2) over the model in Figure 1(a). In this and subsequent figures, the second dimension is offset, and the third dimension is shot position. Note that along the common-offset axis, because the shot times have been referenced to source S1, data corresponding to this source are aligned, whereas those corresponding to S2 are not aligned. [**ER**] gayeni2/. 2s5-or-1

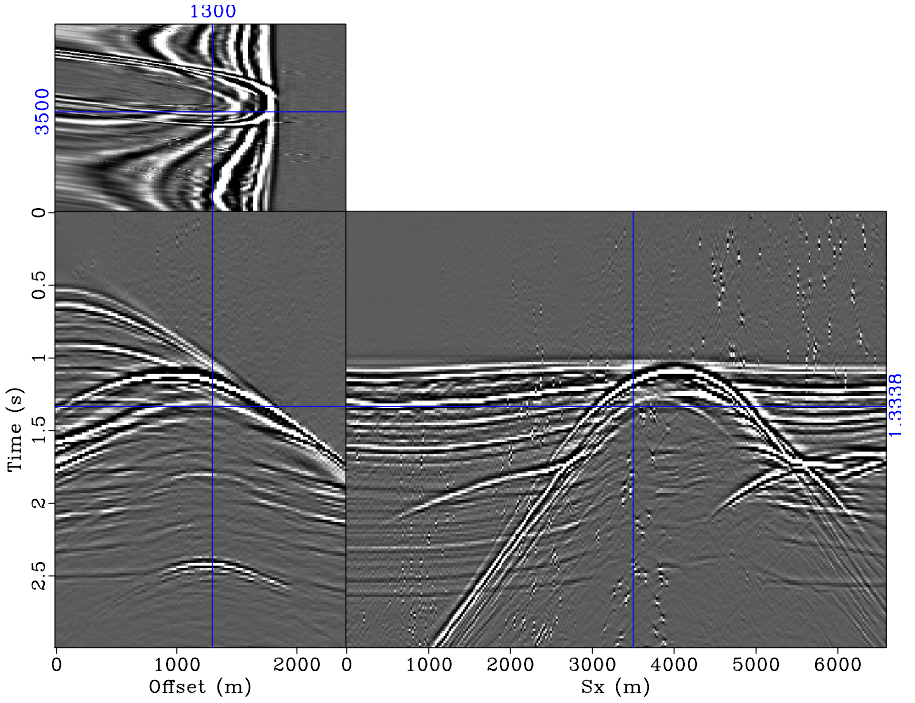


(a)

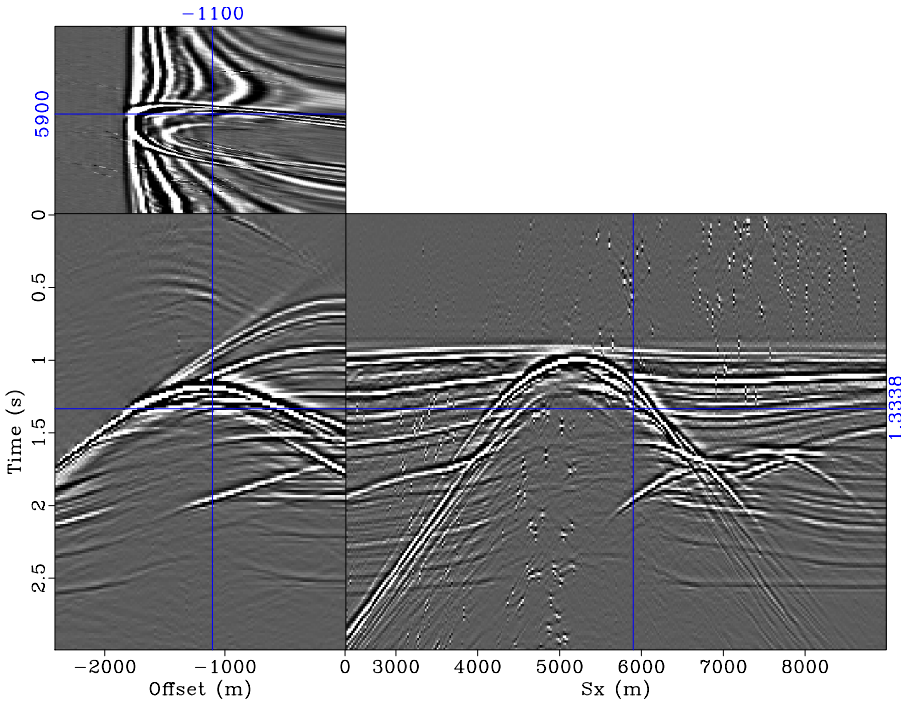


(b)

Figure 3: Single-source data that would have been recorded by (a) source 1 and (b) source 2 over the model in Figure 1(a). These two shot records are the components of the data shown in Figure 2. [**CR**] gayeni2/. 2s5-1,2s5-2

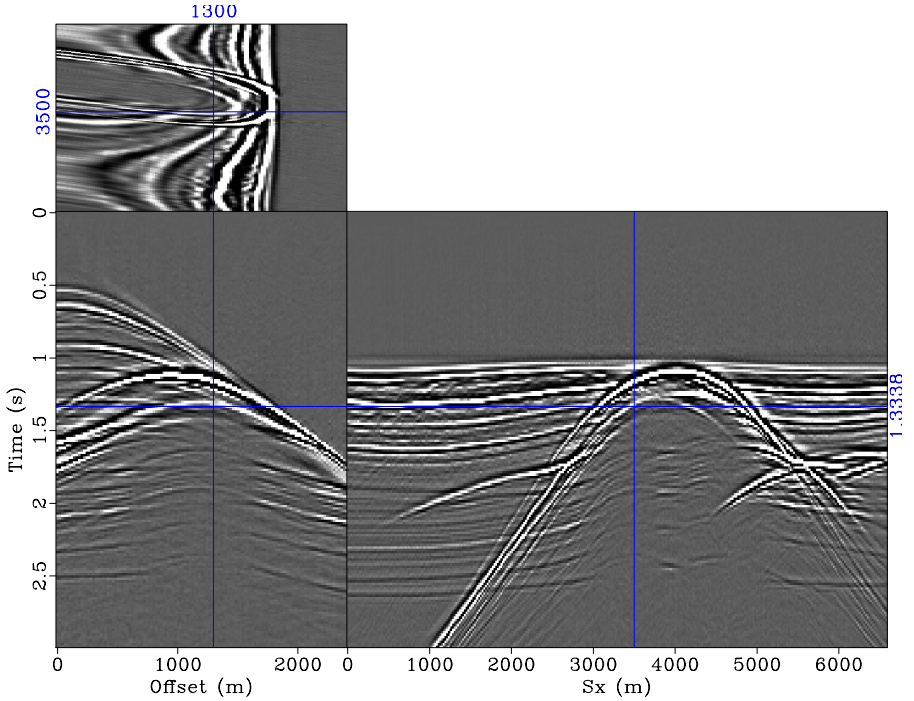


(a)

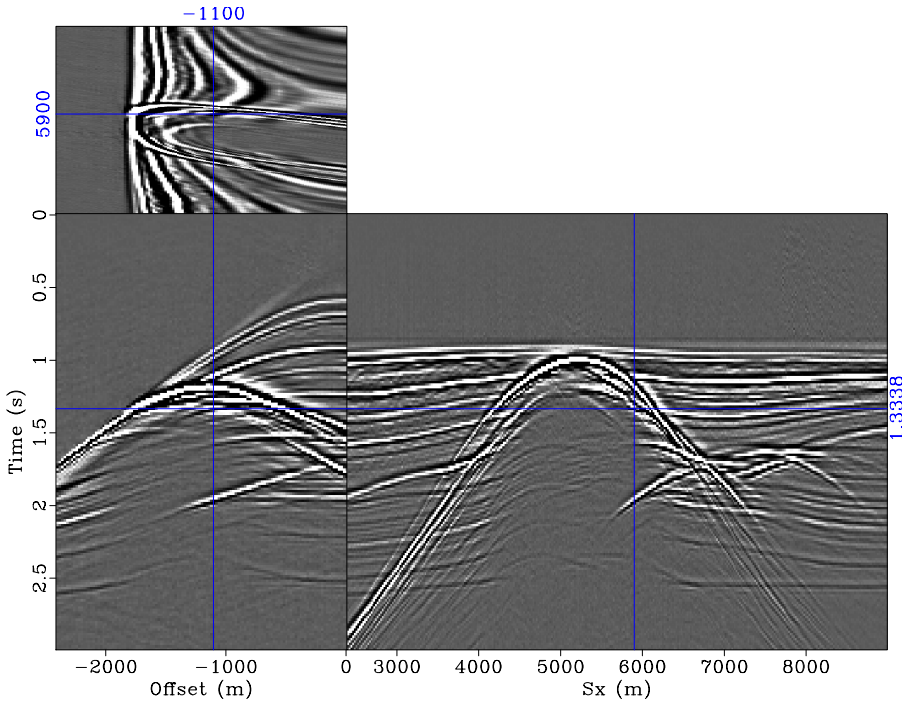


(b)

Figure 4: Shot gathers recovered by unconstrained l_2 inversion for (a) source 1 and (b) source 2. Note that the two data sets are not well separated, as several events which do not exist in the single-source data (Figure 3) are present. [**CR**] gayeni2/. 2s5-l2-1,2s5-l2-2

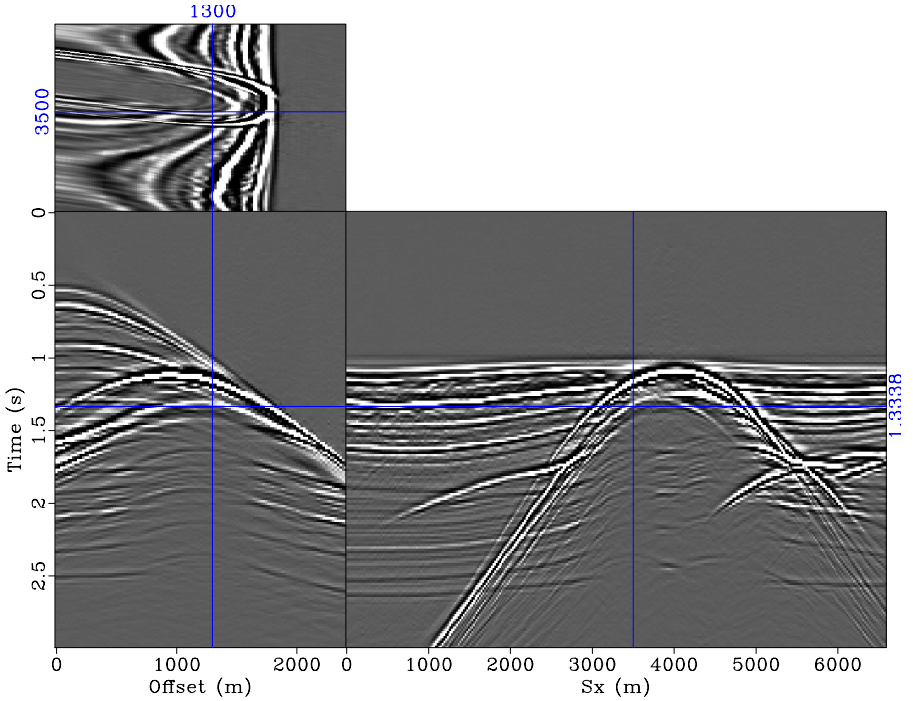


(a)

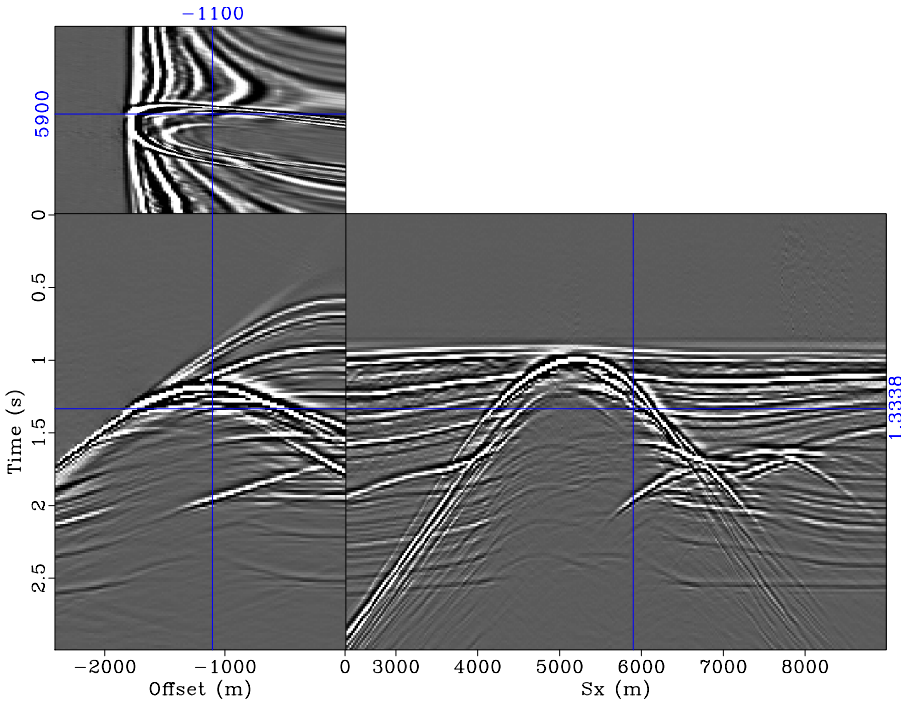


(b)

Figure 5: Shot gathers recovered by unconstrained sparse inversion for (a) source 1 and (b) source 2. Note that the two data sets are well separated and are comparable to the original data (Figure 3). [**CR**] gayeni2/. 2s5-hb-1,2s5-hb-2



(a)



(b)

Figure 6: Shot gathers recovered by dip-constrained sparse inversion for (a) source 1 and (b) source 2. Note that with regularization the residual artifacts present in the unconstrained example (Figure 5) have been attenuated. [**CR**] gayeni2/. 2s5-reg-1,2s5-reg-2

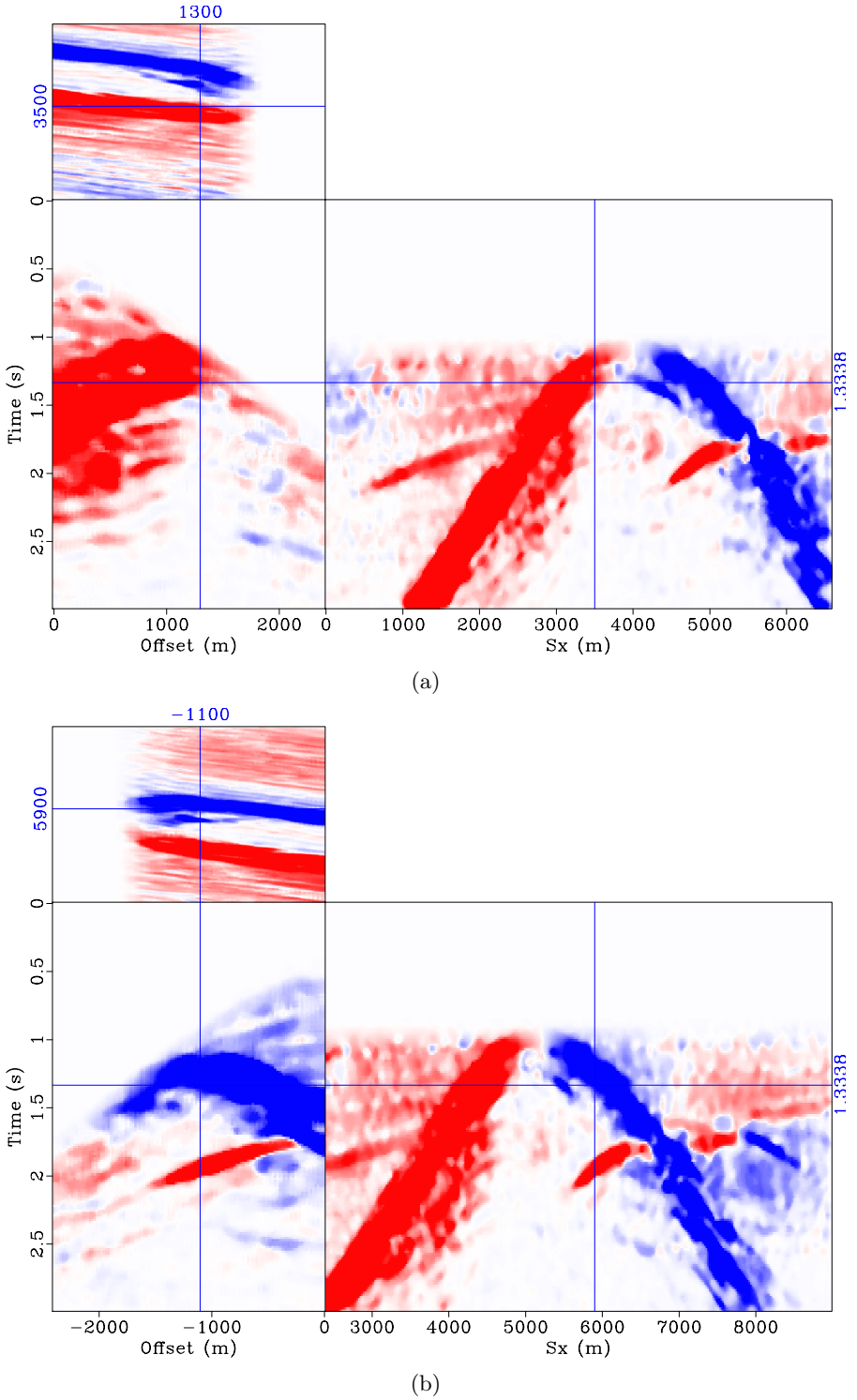


Figure 7: Local dips (common-offset components) for (a) source 1 and (b) source 2, obtained from the unconstrained sparse inversion (Figure 5) and used in the dip-constrained sparse inversion to obtain the results in Figure 6. [**CR**] gayeni2/. 2s5-dip-1,2s5-dip-2

source records are shown in Figure 9. Separation results obtained by inverting the data without dip constraint are shown in Figure 10. Results obtained using the reformulated regressions in equation 6 are shown in Figure 11. Note that residual artifacts seen in the unconstrained results (Figure 10) have been attenuated. Dips estimated from the unconstrained results and used to obtain the constrained results are shown in Figure 12.

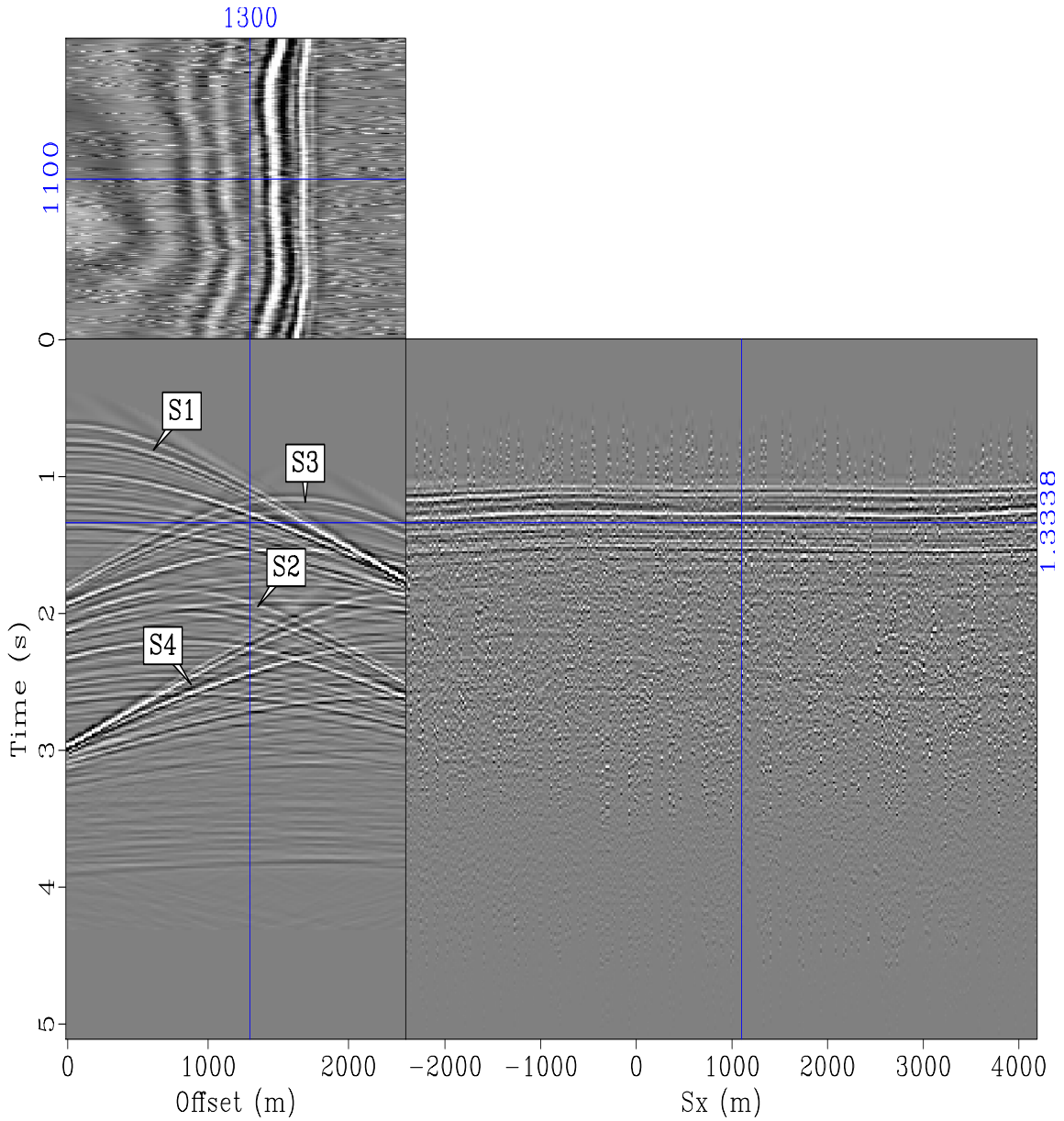


Figure 8: Simultaneous-source data comprising shot-records from four sources (S1, S2, S3 and S4) over the model in Figure 1(b). [**CR**] gayeni2/. 4s4-or-1

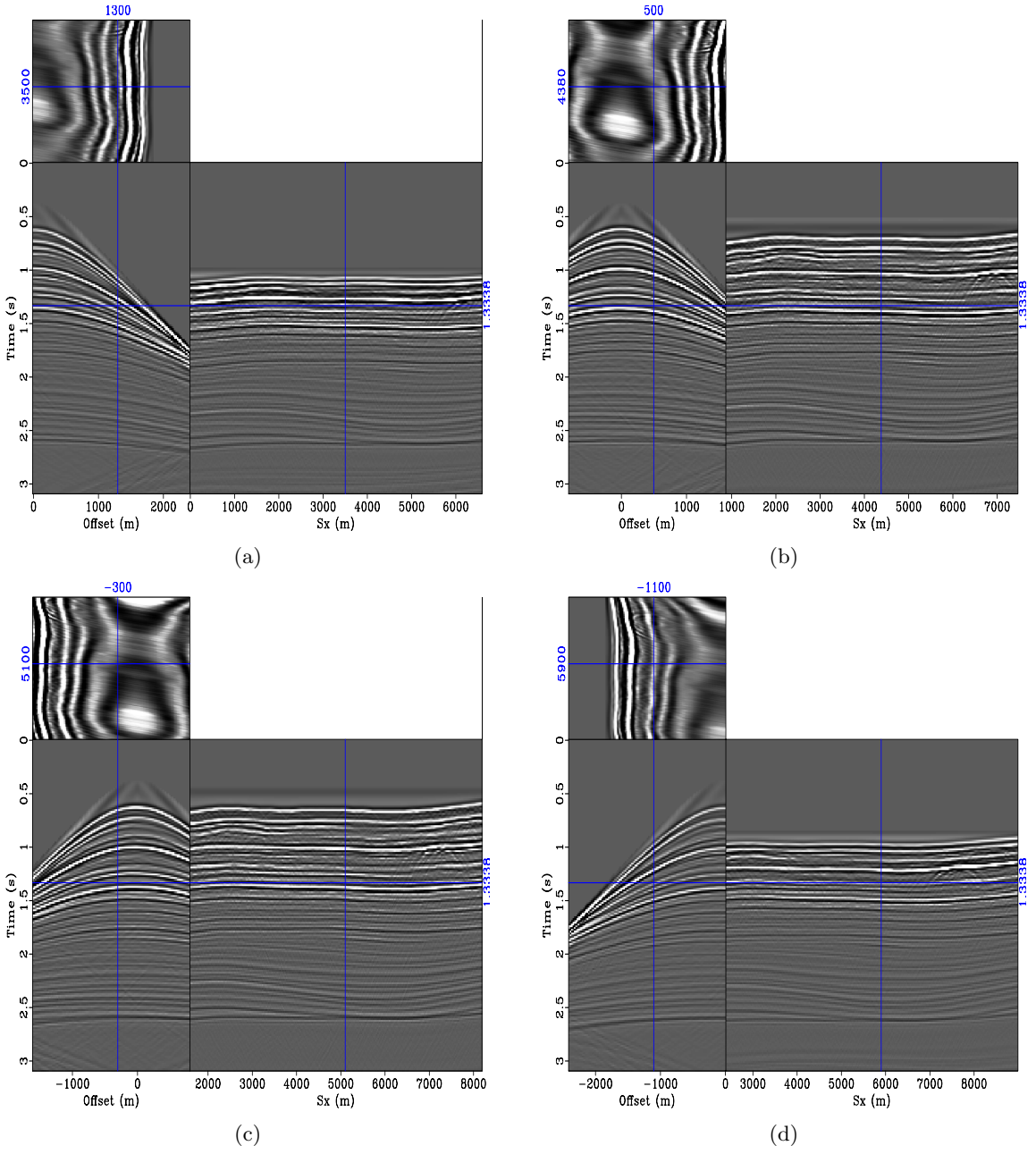


Figure 9: Single-source data that would have been recorded by (a) source 1, (b) source 2, (c) source 3, and (d) source 4 over the model in Figure 1(b). These shot records are the components of the data shown in Figure 8. [**CR**] gayeni2/. 4s4-1,4s4-2,4s4-3,4s4-4

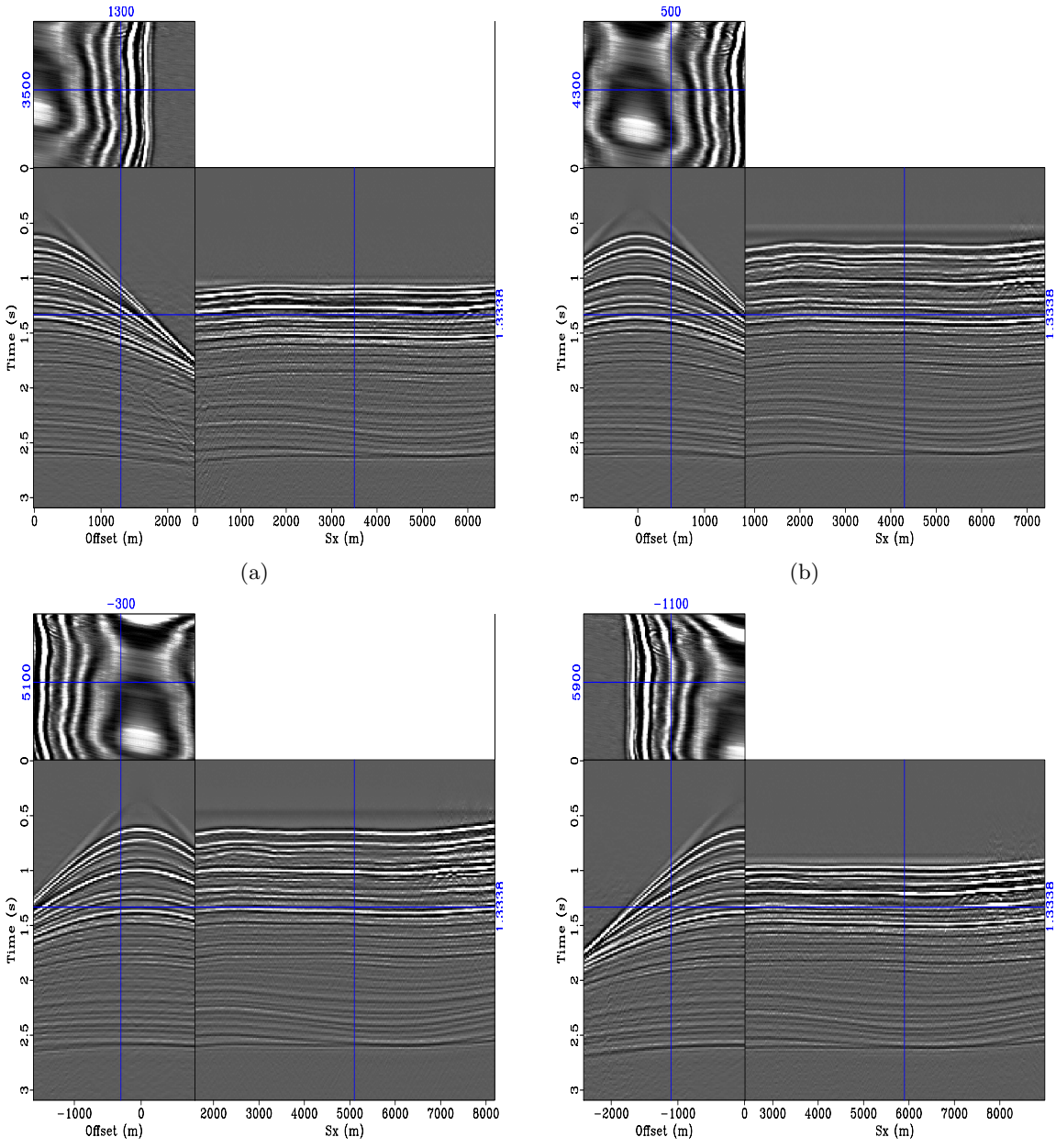


Figure 10: Shot gathers recovered by unconstrained sparse inversion for (a) source 1, (b) source 2, (c) source 3, and (d) source 4. Note that these results contain several residual artifacts compared to the reference single-source records (Figure 9). [**CR**] gayeni2/. 4s4-hb-1,4s4-hb-2,4s4-hb-3,4s4-hb-4

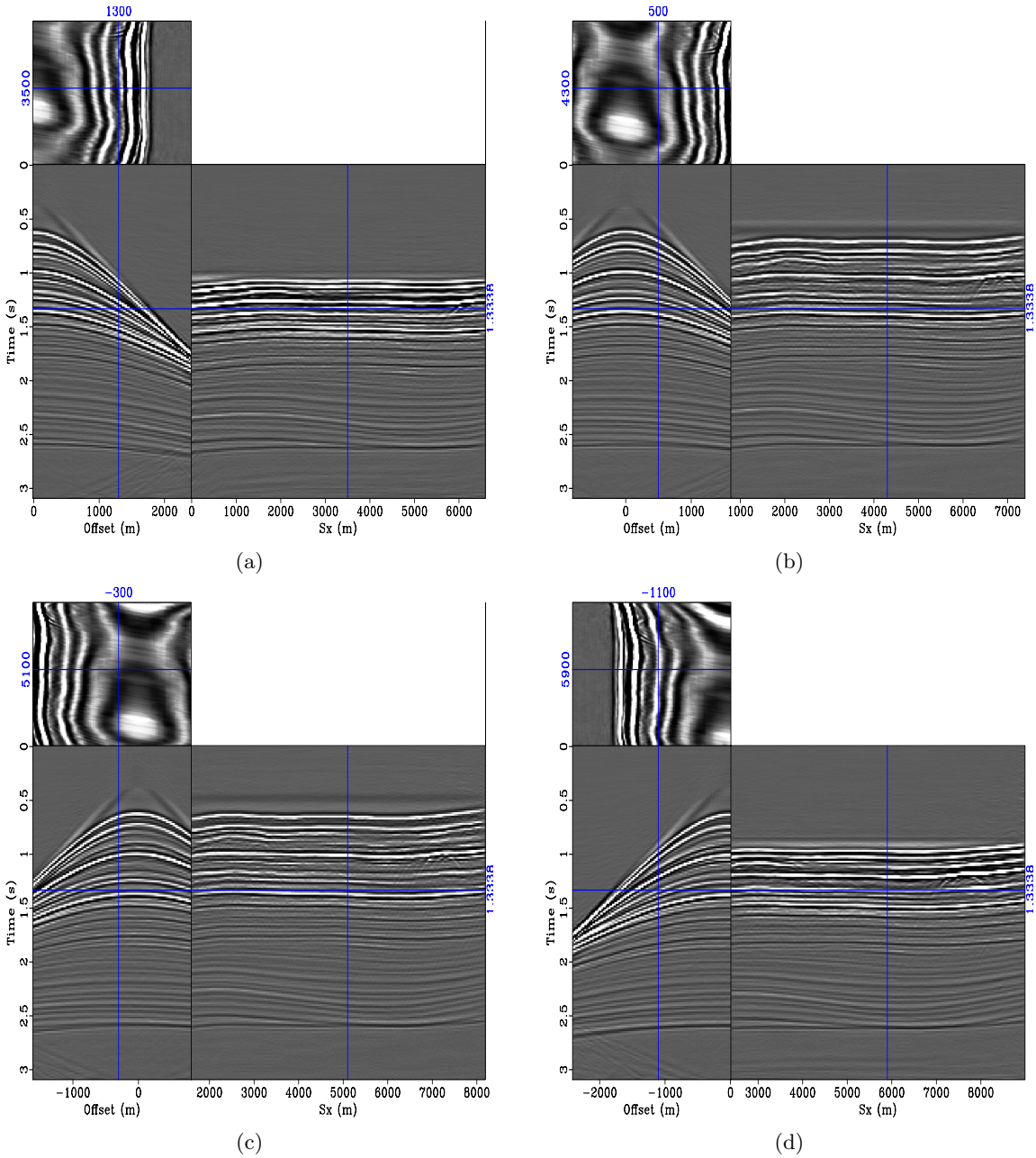


Figure 11: Shot gathers recovered by dip-constrained sparse inversion for (a) source 1, (b) source 2, (c) source 3, and (d) source 4. Residual artifacts present in the unconstrained example (Figure 10) have been attenuated by DCSI. [**CR**] gayeni2/. 4s4-prc-1,4s4-prc-2,4s4-prc-3,4s4-prc-4

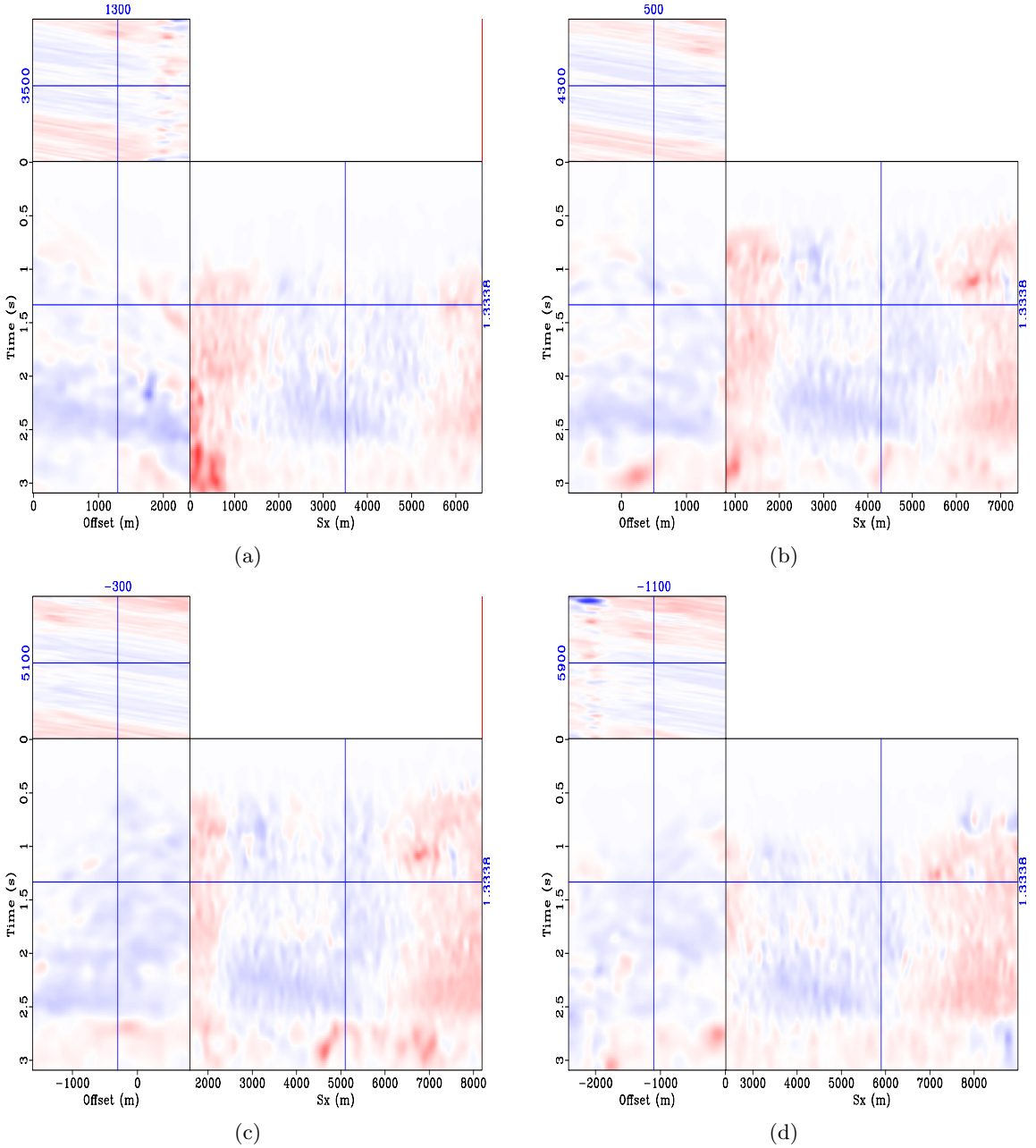


Figure 12: Local dips (common-offset components) for (a) source 1, (b) source 2, (c) source 3, and (d) source 4, obtained from the unconstrained sparse inversion (Figure 10) and used in DCSI to obtain the results in Figure 11. [**CR**] gayeni2/. 4s4-dip-1,4s4-dip-2,4s4-dip-3,4s4-dip-4

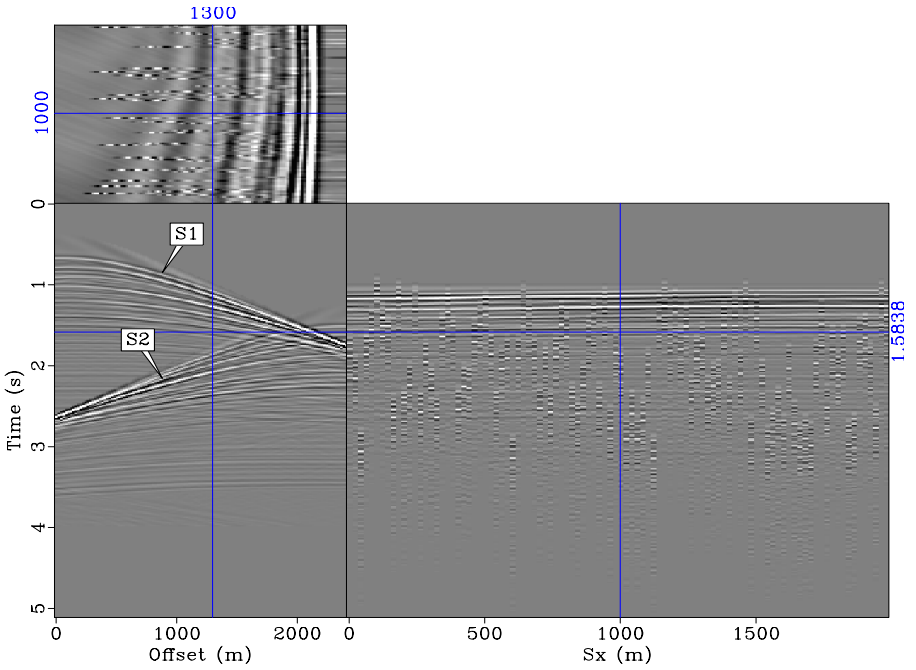
Example 3: Joint inversion of simultaneous-source time-lapse data

In this example, we consider conduct a repeatability test on two different data sets from the same model (Figure1(b)). This represents a repeatability test for a time-lapse seismic monitoring case, where only amplitude differences from production-related changes are of interest. We modeled two sets of simultaneous-source data, each comprised of two sources with 2400 m separation (Figure 13). Because it is impractical to repeat both the geometry and relative shooting times between surveys, time-lapse data acquired with simultaneous sources will have high non-repeatability. Therefore, our data separation procedure serves the dual purpose of separating each simultaneous-source data into component shot records and cross-equalizing the time-lapse data sets. The individual source records for each survey are shown in Figure 14.

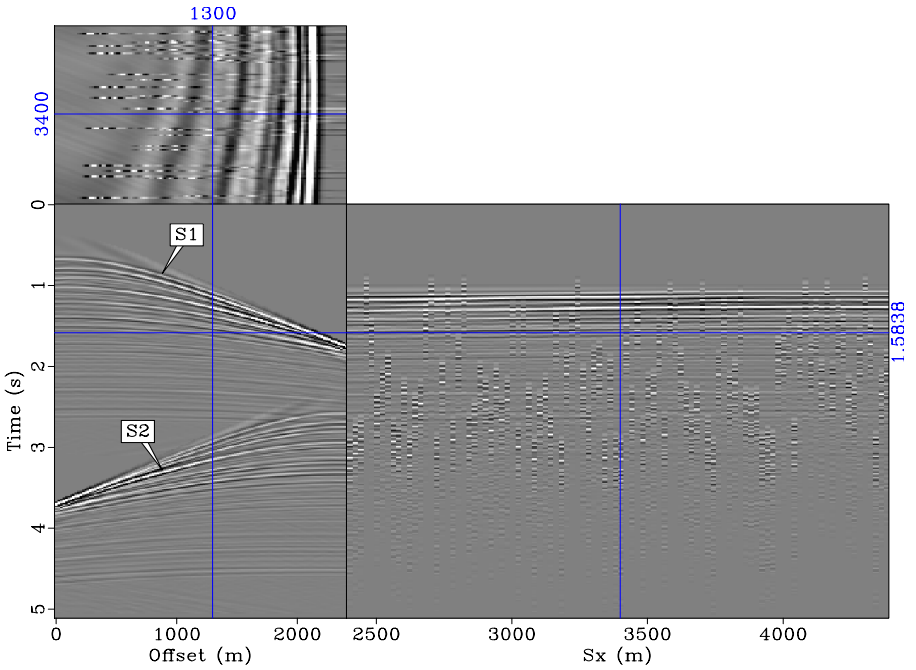
In this example, because we assume no change in the reservoir between surveys, the difference between the two sets of data is zero (Figures 14(e) and 14(f)). Separation results obtained by inverting the data sets separately without spatio-temporal constraints are shown in Figure 15. Note the presence of several artifacts in the difference data computed from the retrieved data sets (Figures 15(e) and 15(f)). Separation results obtained by joint-inversion with spatio-temporal regularization (equation 7) are shown in Figure 16. Note that the STCSI results are cleaner than the unconstrained results. Also, note that residual artifacts present in the unconstrained difference data sets (Figures 15(e) and 15(f)) have been attenuated in the dip-constrained results (Figures 16(e) and 16(f)). Dips estimated from the unconstrained results and used to obtain the constrained results are shown in Figure 17.

DISCUSSION

Any reliable separation method for simultaneous-source data sets must be applicable to any kind of seismic data, must be independent of the number of seismic sources, and must retain important amplitude information. In the first example, we showed that our inversion formulation (DCSI) can be used to separate data from complex (sub-salt) geological environments. The separation results in Figure 4 show that l_2 inversion is inadequate for data separation. Significant improvement is obtained in the quality of these results by using a *hybrid* instead of the l_2 norm (Figure 5). In addition, the separation results can be further improved by dip-constrained inversion (Figure 6) to produce results of comparable quality to the original single-source data (Figure 3). In the second example, we showed that with our approach, we can separate any number of seismic sources. Whereas the unconstrained results (Figure 10) contain several residual artifacts, the dip-constrained results (Figure 11) are comparable to the reference single-source data (Figure 9). In the last example, we showed that this method can be applied to amplitude-sensitive studies such as time-lapse seismic reservoir monitoring. This repeatability test, shows that our method can be used to regularize and cross-equalize time-lapse simultaneous-source data sets. By incorporating both spatial and temporal constraints into the inversion, we are able to attenuate differences caused by non-repeatable acquisition parameters during the separation problem. The residual artifacts present in separately inverted data sets (Figure 15) are removed by our STCSI formulation (Figure 16).



(a)



(b)

Figure 13: Simultaneous-source data sets comprising shot-records from two sources (S1 and S2) over a segment of the model in Figure 1(b). Note that the survey parameters in (a) survey 1 are different from those of (b) survey two. If not taken into account, this discrepancy (non-repeatability) will affect the quality of the time-lapse (difference) data and the resulting estimate of reservoir property change. [**CR**] gayeni2/. 4d-or-1,4d-or-2

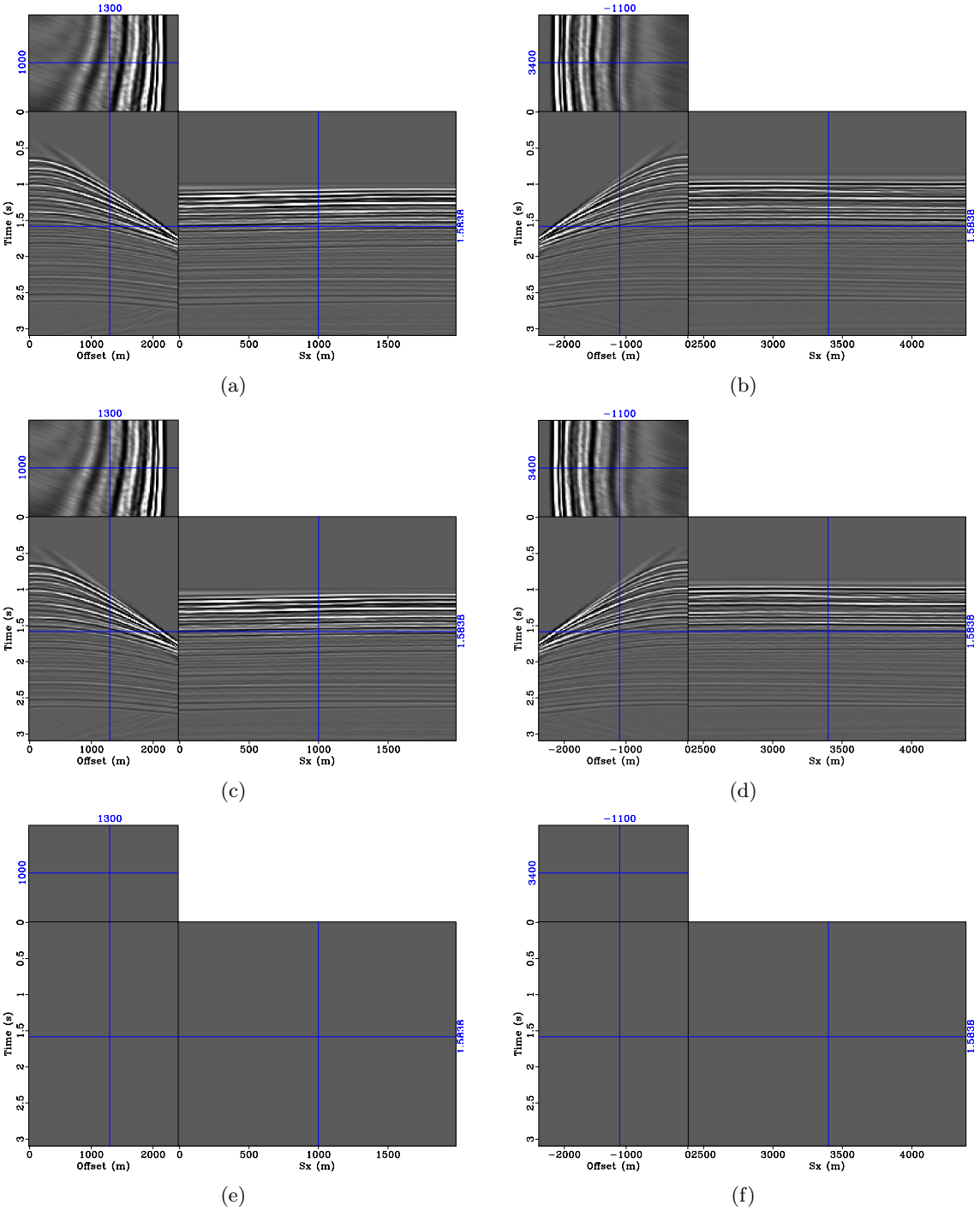


Figure 14: Single-source data that would have been recorded for (a & b) survey 1, and (c & d) survey 2 over a segment of the model in Figure 1(a); and the difference between the two surveys (e & f). The left panel represents data from source 1, whereas the right panel represents source 2. These shot records (a - d) are the components of the two data sets shown in Figure 13. Note that because there is no production-related change between the surveys, the bottom panels are blank. [**CR**] gayeni2/. 4d-1,4d-2,4d-3,4d-4,4d-1-diff,4d-2-diff

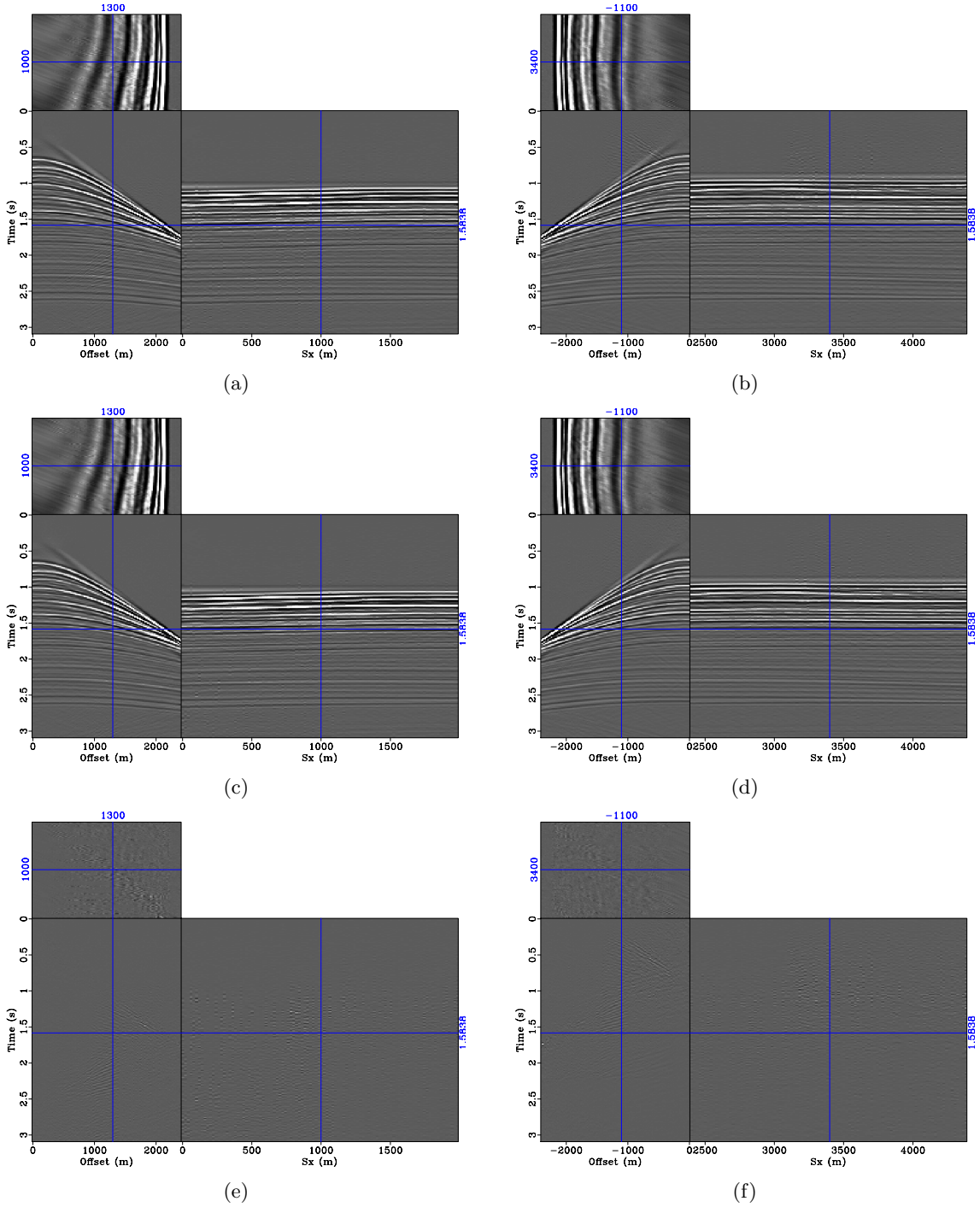


Figure 15: Shot gathers recovered by independent sparse inversion for (a & b) survey 1, and (c & d) survey 2; and the difference between the two surveys (e & f). The left panel represents data from source 1, whereas the right panel represents source 2. Note that the recovered data from the two surveys (top and middle panels) contain several artifacts. Also, note that due to the non-repeatability of the two surveys, the difference data (bottom panels) contain several artifacts. [**CR**] gayeni2/. 4d-hb-1,4d-hb-3,4d-hb-2,4d-hb-4,4d-hb-1-diff,4d-hb-2-diff

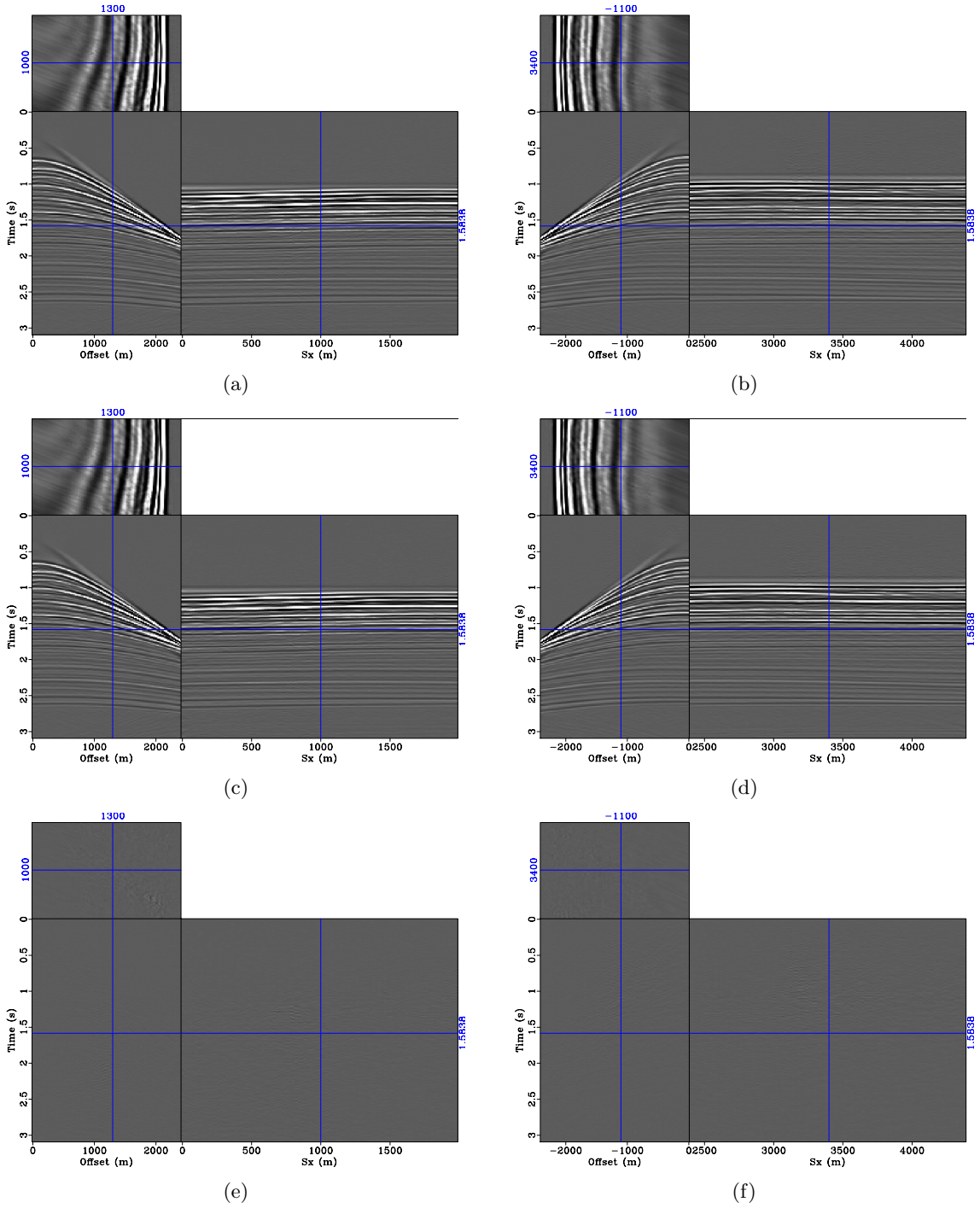


Figure 16: Shot gathers recovered by spatio-temporal constrained sparse inversion for (a & b) survey 1, and (c & d) survey 2; and the difference between the two surveys (e & f). The left panel represents data from source 1, whereas the right panel represents source 2. Note that the residual artifacts present in the unconstrained inversion results, in both the separated and difference data (Figure 15), have been attenuated by STCSI. [**CR**] gayeni2/. 4d-reg-1,4d-reg-3,4d-reg-2,4d-reg-4,4d-reg-1-diff,4d-reg-2-diff

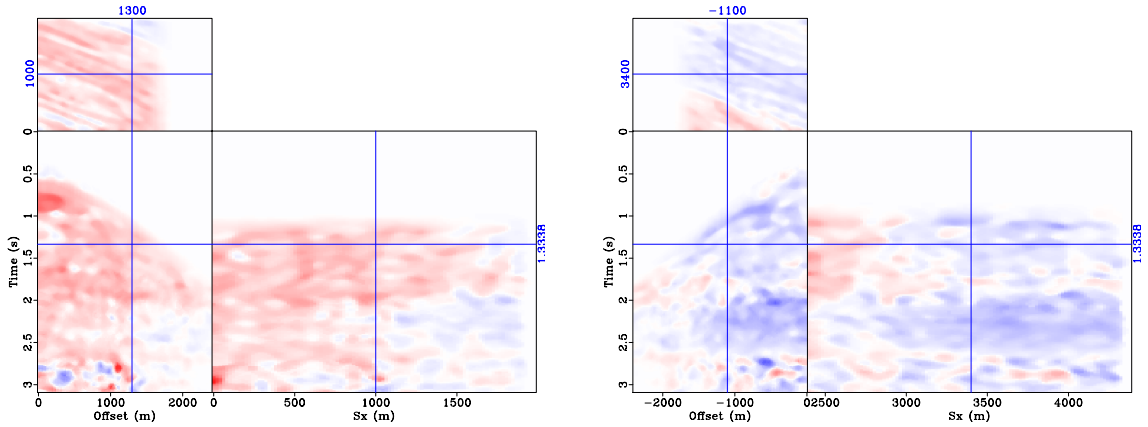


Figure 17: Local dips (common-offset components) for (a) source 1 and (b) source 2 obtained from unconstrained inversion results (Figure 15), and used to obtain the STCSI results (Figure 16). Because there was no change between surveys, each panel was computed as the average of the local dip estimates for the two surveys. [**CR**] gayeni2/. 4d-dip-1,4d-dip-2

CONCLUSIONS

We have demonstrated that simultaneous-source data sets can be separated using an inversion formulation. As shown in the numerical examples from realistic 2D models, our method can be applied in very complex geological environments, to data from multiple seismic sources and to simultaneous-source time-lapse seismic data sets. Incorporating dip information in the inversion procedure through non-stationary directional Laplacians (and in the time-lapse case, temporal constraints) helps to attenuate residual artifacts from the separation process.

REFERENCES

Abma, R., T. Manning, M. Tanis, J. Yu, and M. Foster, 2010, High quality separation of simultaneous sources by sparse inversion: EAGE Expanded Abstracts.

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

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

Beasley, C. J., 2008, Simultaneous sources: A technology whose time has come: SEG Technical Program Expanded Abstracts, **27**, 2796–2800.

Berkhout, A. J. G., G. Blacquière, and E. Verschuur, 2008, From simultaneous shooting to blended acquisition: SEG Technical Program Expanded Abstracts, **27**, 2831–2838.

Claerbout, J. F. and S. Fomel, 2008, Image estimation by example: Geophysical soundings image construction: multidimensional autoregression.

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.

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

Hale, D., 2007, Local dip filtering with directional Laplacians: CWP Project Reiew, **567**, 91–102.

Hampson, G., J. Stefani, and F. Herkenhoff, 2008, Acquisition using simultaneous sources: SEG Technical Program Expanded Abstracts, **27**, 2816–2820.

Howe, D., M. Foster, T. Allen, I. Jack, D. Buddery, A. Choi, R. Abma, T. Manning, and M. Pfister, 2009, Independent simultaneous sweeping in Libya—full scale implementation and new developments: SEG Technical Program Expanded Abstracts, **28**, 109–111.

Huo, S., Y. Luo, and P. Kelamis, 2009, Simultaneous sources separation via multi-directional vector-median filter: SEG Technical Program Expanded Abstracts, **28**, 31–35.

Li, Y., Y. Zhang, and J. Claerbout, 2010, Geophysical applications of a novel and robust L1 solver: SEP Report, **140**.

Moore, I., B. Dragoset, T. Ommundsen, D. Wilson, C. Ward, and D. Eke, 2008, Simultaneous source separation using dithered sources: SEG Technical Program Expanded Abstracts, **27**, 2806–2810.

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

van Mastrigt, P., S. Vaage, M. Dunn, and B. Pramik, 2002, Improvements in 3-d marine acquisition using continuous long offset (clo): The Leading Edge, **21**, 394–399.

Womack, J. E., J. R. Cruz, H. K. Rigdon, and G. M. Hoover, 1990, Encoding techniques for multiple source point seismic data acquisition: Geophysics, **55**, 1389–1396.

A preconditioning scheme for full waveform inversion

Antoine Guitton and Gboyega Ayeni

ABSTRACT

The waveform inversion problem is inherently ill-posed. Traditionally, regularization terms are used to address this issue. For waveform inversion where the model is expected to have many details reflecting the physical properties of the Earth, regularization and data fitting can work in opposite directions: the former smoothing and the later adding details to the model. In this paper, we constrain the velocity model with a model-space preconditioning scheme based on directional Laplacian filters. This preconditioning strategy preserves the details of the velocity model while smoothing the solution along known geological dips. The Laplacian filters have the property to smooth or kill local planar events according to a local dip field. By construction, these filters can be inverted and used in a preconditioned waveform-inversion scheme to yield geologically meaningful models. We illustrate on a 2-D synthetic example how preconditioning with non-stationary directional Laplacian filters outperforms traditional waveform inversion when sparse data are inverted for. We think that preconditioning could benefit waveform inversion of real data where (for instance) irregular geometry, coherent noise and lack of low frequencies are present.

INTRODUCTION

The goal of waveform inversion is to derive physical properties of the Earth, such as P-wave velocity, S-wave velocity, or density. These properties can be related to the presence of hydrocarbons in the subsurface and their estimation is one of the most important goal in seismic processing. In practice, we try to minimize the function

$$f(\mathbf{m}) = \|\mathbf{u}_{obs} - \mathbf{u}_{mod}\|^{norm} \tag{1}$$

where $\mathbf{m}$ is a vector of physical properties (what we are looking for), $\mathbf{u}_{obs}$ the observed and $\mathbf{u}_{mod}$ the modeled data. Note that f is a 1-D function defined by the choice of a norm. In practice, the ℓ^2 norm is often chosen, but the ℓ^1 norm seems to have more practical needs for its robustness to non-Gaussian noise present in the data (Crase et al., 1990). The minimization of $f(\mathbf{m})$ can be achieved using iterative methods. Tarantola (1984) establishes that the model can be updated as follows:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n . \nabla f(\mathbf{m}_n) \tag{2}$$

where $\nabla f(\mathbf{m})$ is the gradient of $f(\mathbf{m})$ and α_n a step-length that needs to be estimated. It turns out that the computation of the gradient is equivalent to a reverse-time migration (Lailly, 1984).

Unfortunately, although a promising approach, waveform inversion is hampered by many issues. The main one is the presence of local minima in f. To circumvent this problem, the

data can be inverted in a multi-scale fashion in the time (Bunks et al., 1995) or frequency domain (Sirgue and Pratt, 2004). Furthermore, time damping of the input data offers opportunities to focus the inversion on different parts of the data, thus reducing the effects of local minima (Brenders et al., 2009).

Traditionally, ill-posed problems can be solved by adding a regularization term to the objective function. Very often, a regularization term that can penalize differences between neighboring points is selected. However, whereas waveform inversion tends to add details to a velocity model, regularization tends to smooth them out, thus working against our primary goal: fitting the data. One way to address these somewhat conflicting goals is to use preconditioning. Here, we show how we can geologically constrain the velocity model by using a non-stationary preconditioning approach. This method requires two ingredients: a dip estimation method and a local dip filtering technique. We use the method of Fomel (2002) for the former and of Hale (2007) for the later.

In this paper we first introduce the waveform inversion approach, with and without preconditioning. We show that preconditioning amounts to a simple change of variable which, in effect, changes the gradient direction. Then, we present our method of local dip filtering, which follows Hale's. Finally, we present synthetic results on a modified version of the Marmousi model. These results demonstrate that non-stationary, preconditioned inversion yields geologically plausible models.

METHOD

In this paper, we use a time domain approach for solving the scalar acoustic wave equation (parametrized in terms of P-wave velocity v_p):

$$\frac{\partial^2 u(\mathbf{x},t)}{\partial t^2} - v_p(\mathbf{x})^2 \nabla^2 u(\mathbf{x},t) = v_p(\mathbf{x})^2 s(\mathbf{x},t). \tag{3}$$

with the source term $s(\mathbf{x},t) = \delta(\mathbf{x}-\mathbf{x}_s)S(t)$ where $S(t)$ is the source function at $\mathbf{x}_s$ and $u(\mathbf{x},t)$ the pressure field. Tarantola (1984) derives the expression of the gradient for the acoustic equation (3) for each component of $\mathbf{m}$ (equal to v_p only in this case).

$$\nabla f(\mathbf{m}_n) = \frac{2}{{\mathbf{v}_{p_n}}^3} \sum_{shots} \sum_t \frac{\partial^2 \overrightarrow{\mathbf{u}_n}}{\partial t^2} \cdot \overleftarrow{\delta \mathbf{u}_n} \tag{4}$$

where $\overleftarrow{\delta \mathbf{u}_n}$ is the *backward* propagated residual at iteration n such that $\delta \mathbf{u}_n = \mathbf{u}_{obs} - \mathbf{u}_n$ and $\overrightarrow{\mathbf{u}_n}$ is the *forward* propagated synthetic source. For our iterative method, we opted for the L-BFGS approach of Nocedal (1980). This quasi-Newton approach computes an estimate of the inverse Hessian iteratively by using a user-defined number of solution and gradient vectors. One of the main benefits of this technique is that because the Hessian is never explicitely formed, there is significant computational and memory savings. W ith the L-BFGS solver, the model is updated as follows:

$$\mathbf{m}_{n+1} = \mathbf{m}_n - \alpha_n \mathbf{H}_n^{-1} \nabla f(\mathbf{m}_n), \tag{5}$$

where $\mathbf{m}_{n+1}$ is the updated solution, α_n the step length computed by a line search that ensures a sufficient decrease of $f(\mathbf{m})$ and $\mathbf{H}_n \approx \nabla^2 f(\mathbf{m}_n)$ the approximate Hessian. To improve chances of not falling into a local minimum, we follow a multi-scale approach (Bunks et al., 1995) where the source and data are bandpassed prior to inversion. We introduce our preconditioning scheme in the following section.

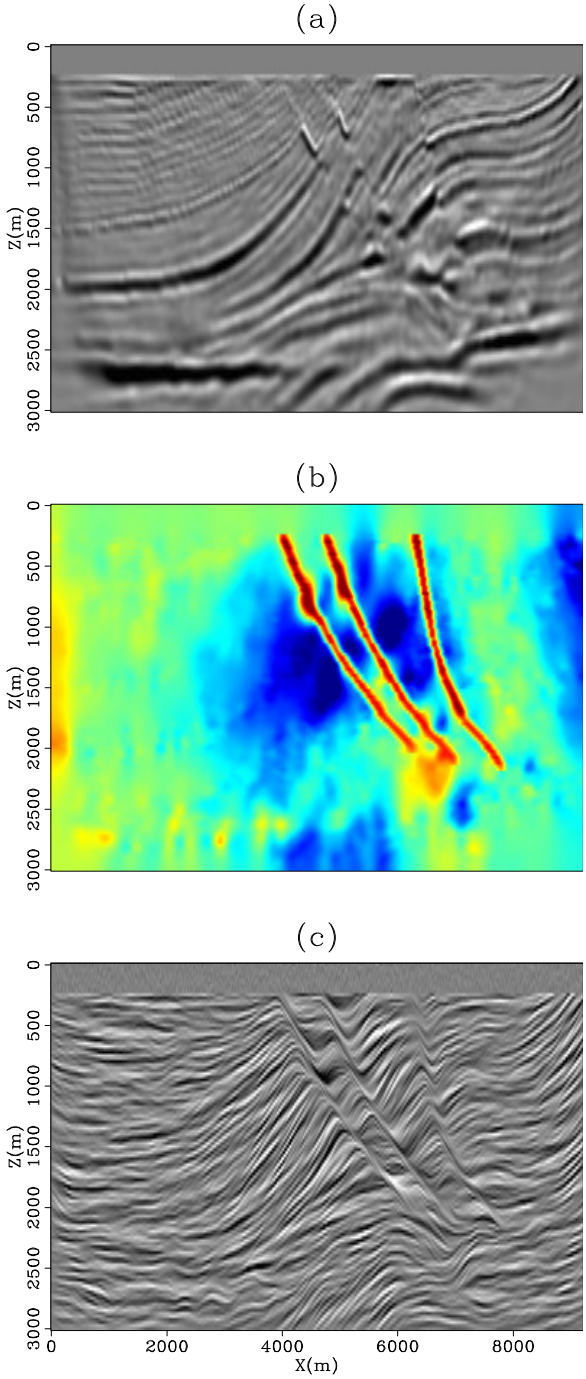


Figure 1: Illustration of the preconditioning operator **S**. We first migrate the synthetic Marmousi dataset to obtain (a). Then, suing the migrated image, we compute the dip field (b). In this example, faults are picked by hand and smoothed horizontally to preserve discontinuities in the velocity model. Applying **S** built from (b) to a random field shows in (c) the texture of the migrated section in (a). Note how the water-column is not smoothed and how the fault locations are clearly visible. [**ER**] antoine1/. MarmLapSEG

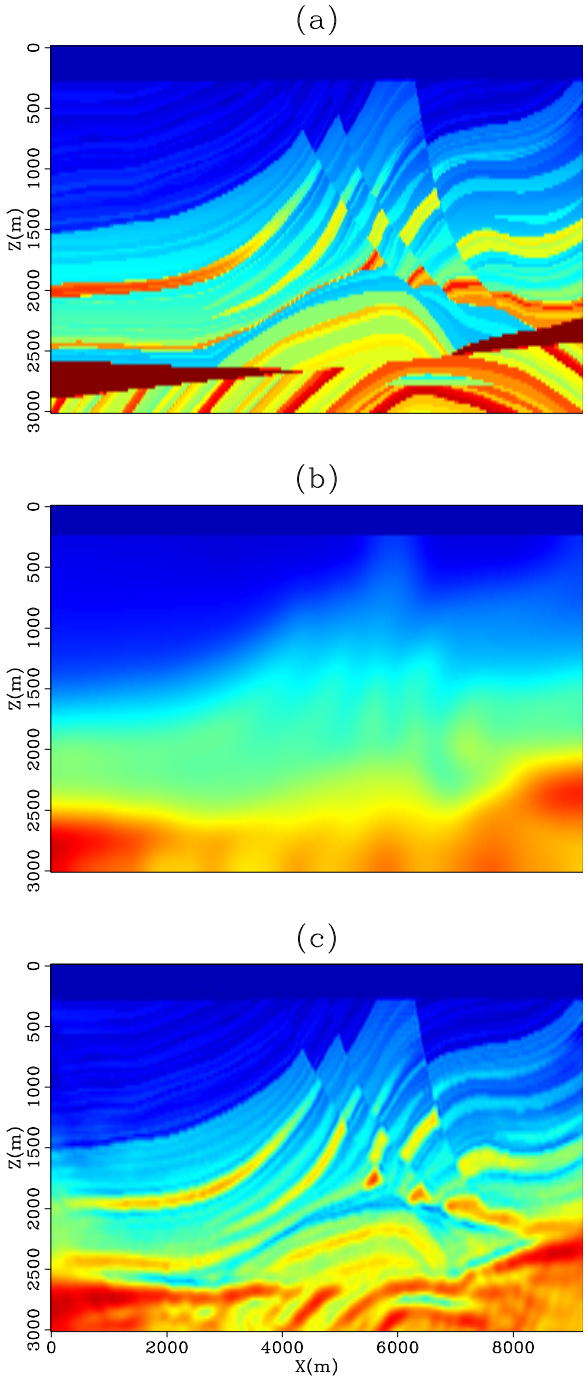


Figure 2: (a) True velocity model used to generate the synthetic dataset. (b) Initial guess obtained by smoothing the true model in (a). (c) Estimated model after waveform inversion. No preconditioning is applied in this case. Four frequency bands were used to bandpass both the source and the data prior to inversion (0-4Hz, 0-8Hz, 0-12Hz, 0-16Hz). The velocity model is recovered very well. [**NR**] antoine1/. MarmWVISEG

PRECONDITIONING

Preconditioning amounts to a change of variable $\mathbf{m} = \mathbf{S}\mathbf{p}$ where $\mathbf{p}$ is a new variable used for the inversion and $\mathbf{S}$ the preconditioning operator. In this paper, we define this operator as non-stationary deconvolution with local dip-filters. Having introduced the new variable $\mathbf{p}$, the iterative scheme in equation (5) becomes:

$$\mathbf{p}_{n+1} = \mathbf{p}_n - \alpha_n \tilde{\mathbf{H}}_n^{-1} \nabla \tilde{f}(\mathbf{p}_n), \tag{6}$$

where

$$\nabla \tilde{f}(\mathbf{p}_n) = \mathbf{S}' \nabla f(\mathbf{m}_n) = \mathbf{S}' \nabla f(\mathbf{S}\mathbf{p}_n) \tag{7}$$

and $\mathbf{S}'$ is the transpose of $\mathbf{S}$. Therefore, with preconditioning, we obtain a new gradient direction. In our scheme, we will opt for a preconditioning operator that steers the gradient toward a geologically constrained solution. Note that in Equation (6) the approximate Hessian in equation (5) is blind to this change of variable as it is built from gradient and solution step vectors only. Assuming that we can estimate $\mathbf{S}$ and compute its adjoint and inverse (to accommodate any starting guess $\mathbf{p}_0 = \mathbf{S}^{-1}\mathbf{m}_0$), preconditioning can be easily introduced in any waveform inversion scheme. Once a solution vector $\mathbf{p}_{sol}$ has been found, the final model is obtained by computing

$$\mathbf{m}_{sol} = \mathbf{S}\mathbf{p}_{sol}. \tag{8}$$

Now we present our choice of preconditioning operator $\mathbf{S}$.

DEFINING THE PRECONDITIONING OPERATOR S

Preconditioning amounts to a change of the gradient direction. For waveform inversion, a gradient that embeds some geological information could result in more meaningful velocity models. To this end, we follow the approach of Hale (2007) for the construction of the operator $\mathbf{S}$. Doing so, this operator becomes a non-stationary deconvolution with directional Laplacian filters.

Directional Laplacian filters are built from small wavekill filters $\mathbf{A}$, similar to those of Claerbout (1995). Wavekill filters have the ability to anihilate local planar-events with a given dip. From these filters, new operators $\mathbf{A}'\mathbf{A}$ are formed by autocorrelation. These new operators are then factorized into minimum-phase filters $\tilde{\mathbf{A}}$ such that $\tilde{\mathbf{A}}'\tilde{\mathbf{A}} \approx \mathbf{A}'\mathbf{A}$. Having minimum-phase filters, we can build a stable non-stationary deconvolution operator $\mathbf{S} = \tilde{\mathbf{A}}^{-1}\tilde{\mathbf{A}}'^{-1}$ and its inverse $\mathbf{S}^{-1} = \tilde{\mathbf{A}}'\tilde{\mathbf{A}}$. Because the wavekill filter $\mathbf{A}$ is dip dependent, the operator $\mathbf{S}$ has the ability to smooth along a given direction as well. Therefore, if we can estimate a dip field that contains some desired geological features, the directional Laplacian filters can make the solution of an inverse problem resemble such features.

In practice, we estimate and use the directional Laplacian filters as follows: first, we estimate a dip field following the approach of Fomel (2002); then we estimate a bank of directional Laplacian filters for a range of angles; finally we apply the appropriate inverse Laplacian filter on each sample according to the local dip. One added feature of our preconditioning scheme is that the strength of the smoothing can be controlled spatially: different areas with similar dips can be smoothed across different distances. These areas are identified by a weighting function which varies from high values (i.e., little smoothing) to low values (i.e., strong smoothing).

To illustrate the preconditioning operator **S**, we show in Figure 1a the migration result of a synthetic dataset based on the 2-D Marmousi model. This result is obtained with Reverse Time Migration (RTM). In real data cases, the dip field could be re-estimated iteratively from a migrated image estimated with the updated velocity field, adding a third outside loop to our waveform inversion algorithm (one for frequency band and one for the muting/masking operator). This possibility is not investigated in this paper.

From the RTM image, we can estimate the local dip field (Figure 1b). This dip field is obtained iteratively with some smoothing using the method of Fomel (2002). We also picked by hand the location of three faults. From these picks, we estimated the dip on the fault and smoothed the local dip horizontally. These three faults were picked to get sharper velocity contrasts. Now, we apply the operator **S** to a random field the size of the migrated image in Figure 1a to obtain Figure 1c. We notice that the "texture" of the original migrated image is recovered and that no smoothing is applied in the water layer: for this example, our weigthing function had only two values separated by the water bottom. Finally, we can clearly identify the fault locations. In the next section, we demonstrate that this operator can be used to constrain the solution of waveform inversion.

EXAMPLES

We illustrate the geologically-constrained waveform inversion method on a synthetic dataset. We modified the Marmousi 2-D velocity model by adding a 250 m. thick water layer (Figure 2a). We created 184 shots 50 m. apart with a fixed receiver array (369 in total) at the surface using the scalar wave equation (no density). The source is a Ricker-2 wavelet with a fundamental frequency of 8Hz. To illustrate that our inversion works (without preconditioning), we show in Figure 2c the result of waveform inversion with four frequency scales (0-4Hz, 0-8Hz, 0-12Hz, and 0-16Hz) using the starting guess in Figure 2b (obtained by smoothing the true model in Figure 2a) and using all the shots. There is a very good match between the inverted and true model. Because all the data was used, little would be gained by using preconditioning.

To make a compelling case, we kept only four shots, 2.5 Km. apart. First, we show in Figure 3 a comparison between the gradient without preconditioning $\nabla f(\mathbf{m}_n)$ and the gradient with preconditioning back in the velocity space $\mathbf{S}\nabla\tilde{f}(\mathbf{p}_n)$. Because only 4 shots are present, the unpreconditioned gradient looks noisy and resemble geology in very few locations only. Some authors suggest attenuating the high wavenumbers in the gradient by smoothing it after each iteration (Ravaut et al., 2004), where the size of the smoothing operator in the horizontal and vertical directions is a function of an average wavelength at a given frequency. This bears a resemblance with our proposed scheme but doesn't allow for directional smoothing. On the contrary, thanks to the preconditioning with directional Laplacian filters, the reprojected gradient in Figure 3b shows the geology captured in the dip field of Figure 1b very well.

Now, we show in Figure 4 the inversion results for the unpreconditioned (Figure 4a) and preconditioned inversion (Figure 4b). Although quite noisy, the unpreconditioned result shows the geology very well: the presence of low frequencies in the data, along with the multi-scale approach, act as regularization operators. This effect will be less pronounced with real data where low frequencies are often missing. The preconditioned inversion result in Figure 4b is much cleaner: the geology at the reservoir level is recovered very well.

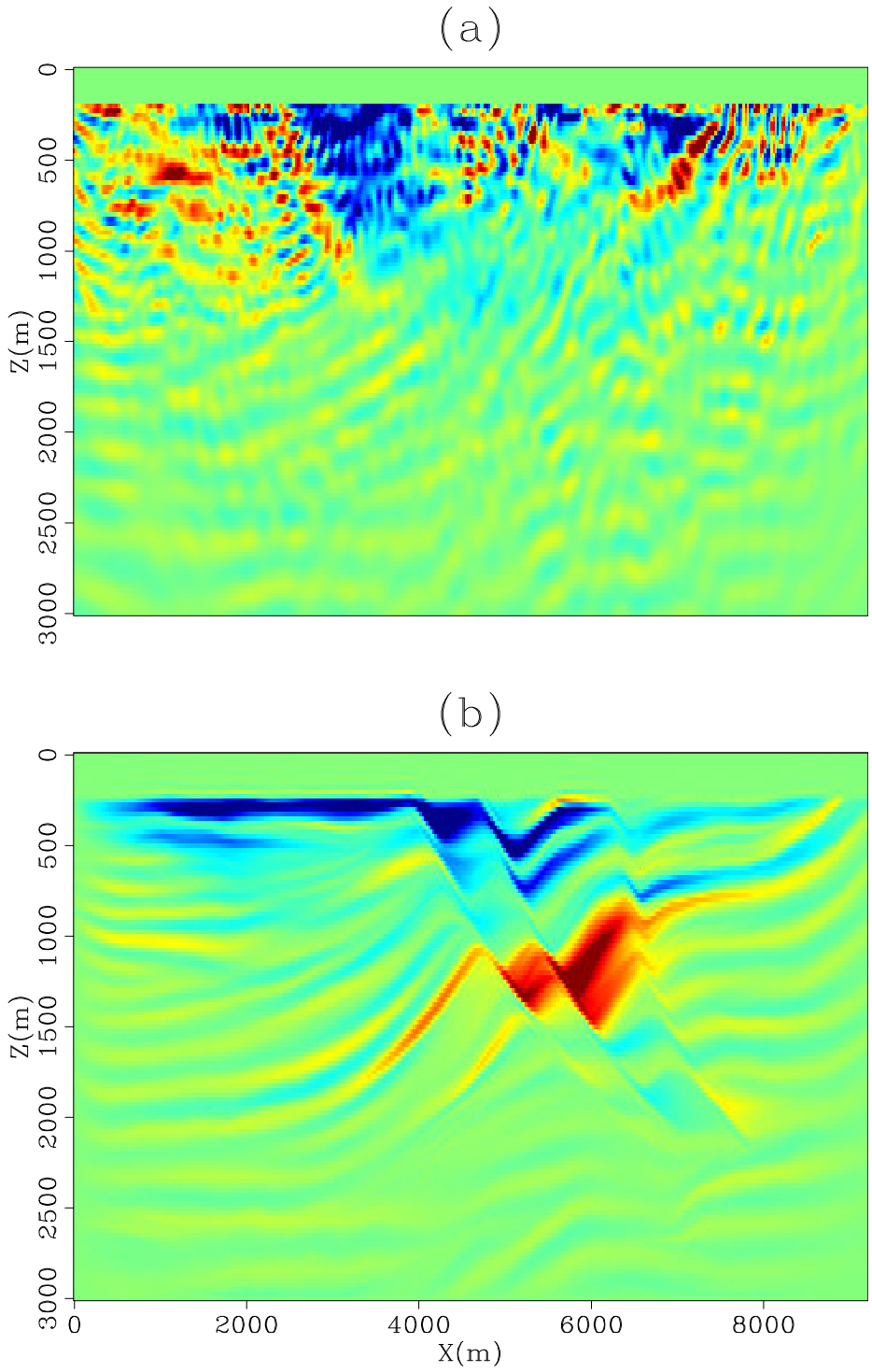


Figure 3: (a) Gradient $\nabla f(\mathbf{m}_n)$ of the unpreconditioned inversion after 4 iterations of the 0-8Hz scale (4 shots, 2.5 Km. apart). (b) Reprojected gradient $\mathbf{S}\nabla\tilde{f}(\mathbf{p}_n)$ of the preconditioned inversion after 4 iterations of the 0-8Hz scale (4 shots, 2.5 Km. apart). With preconditioning, the gradient follows the information captured in the dip field and looks more geologically appealing than in (a). [**NR**] antoine1/. MarmGRADSPARSESEG

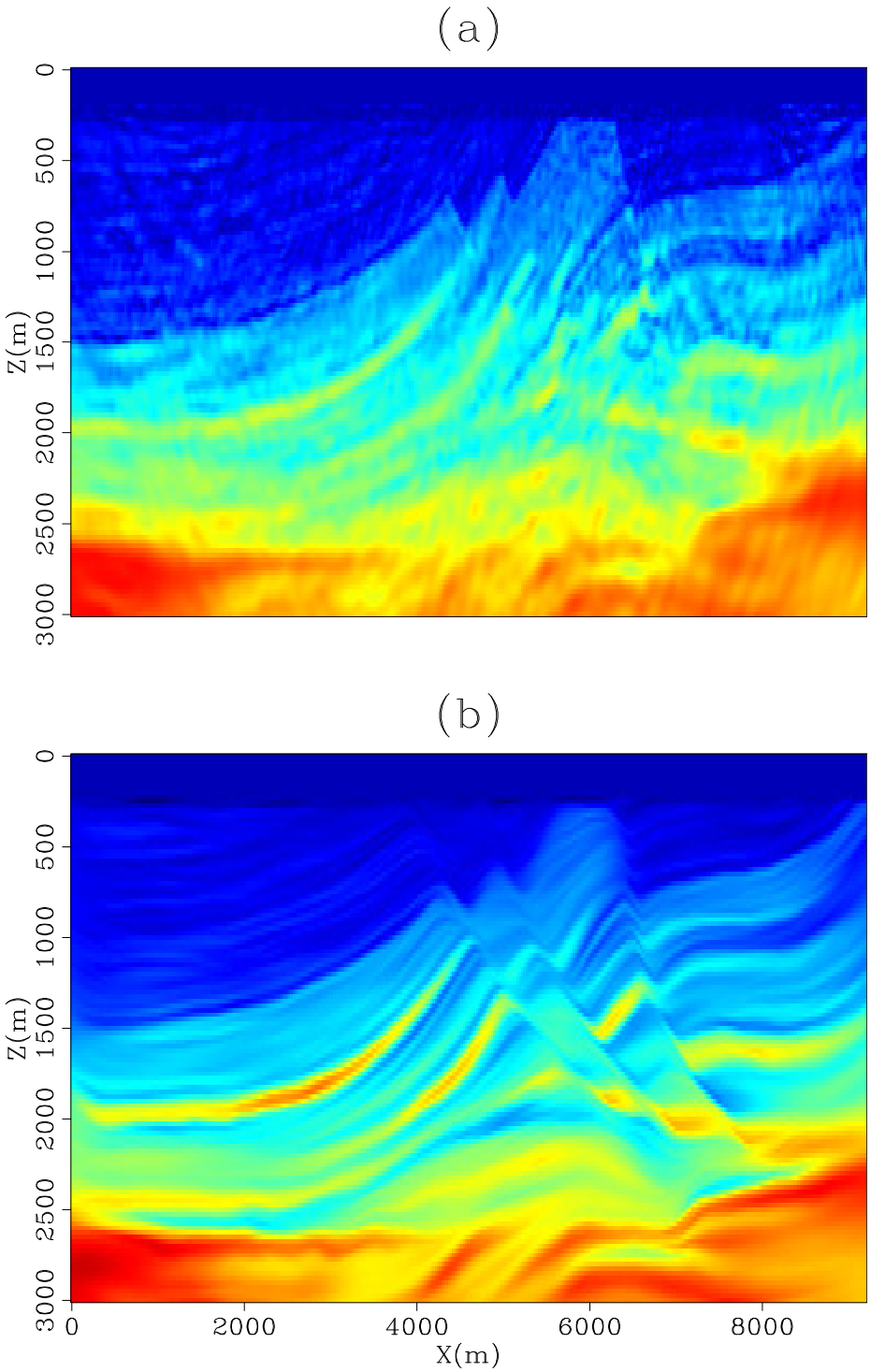


Figure 4: (a) Inversion result for the unpreconditioned inversion. (b) Inversion result for the preconditioned inversion with directional Laplacian filters. The geology at the reservoir level is recovered very well in (b). [**NR**] antoine1/. MarmSPARSESEG

CONCLUSIONS

Preconditioning waveform inversion with non-stationary directional Laplacian filters can yield geologically meaningful velocity models. It can help decrease artifacts due to acquisition geometry or inconsistencies in the data (not shown here). We anticipate that preconditioning can play a bigger role with real data where low frequencies are often lacking, where data are noisy and where the acquisition geometry is irregular.

ACKNOWLEDGMENTS

The authors thank Geoimaging Solutions Inc. and Repsol for permission to publish these results. This paper constitutes a contribution to the Phoenix project.

REFERENCES

Brenders, A., R. Pratt, and S. Charles, 2009, Waveform tomography of 2-d seismic data in the canadian foothills - data preconditioning by exponential time-damping: EAGE expanded abstract, U041.

Bunks, C., F. Saleck, S. Zaleski, and G. Chavent, 1995, Multiscale seismic waveform inversion: Geophysics, **60**, 1457–1473.

Claerbout, J., 1995, Earth Soundings Analysis: Processing Versus Inversion: Blackwell Scientific Publications.

Crase, E., A. Pica, M. Noble, J. McDonald, and A. Tarantola, 1990, Robust elastic nonlinear waveform inversion: Application to real data: Geophysics, **55**, 527–538.

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

Hale, D., 2007, Local dip filtering with directional laplacians: CWP report, 91–102.

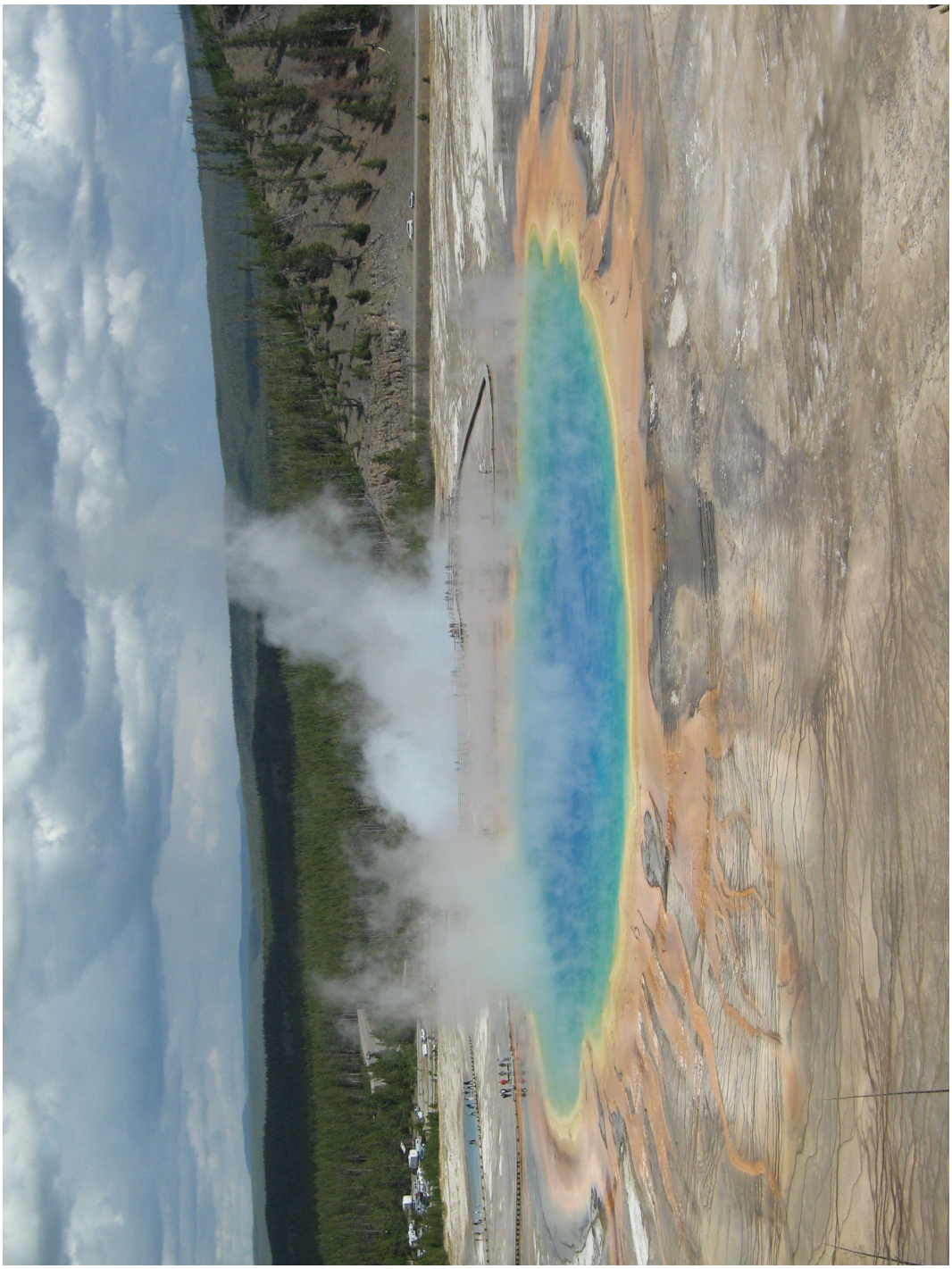
Lailly, P., 1984, The seismic inverse problem as a sequence of before stack migrations: Conference on Inverse Scattering.

Nocedal, J., 1980, Updating quasi-newton matrices with limited storage: Mathematics of Computation, **95**, 339–353.

Ravaut, C., S. Operto, L. Improta, J. Virrieux, A. Herrero, and P. dell'Aversana, 2004, Multi-scale imaging of complex structures from multi-fold wide-aperture seismic data by frequency-domain full-wavefield inversions: Application to a thrust belt: Geophysical Journal International, **159**, 1032–1056.

Sirgue, L. and R. Pratt, 2004, Efficient waveform inversion and imaging: A strategy for selecting temporal frequencies: Geophysics, **69**, 231–248.

Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, **49**, 1259–1266.



Hybrid-norm and Fortran 2003: Separating the physics from the solver

Robert G. Clapp

ABSTRACT

Object-oriented approaches allow a separation between solvers and operators. An abstract vector class is created with a limited set of methods. Solvers are written in terms of this abstract vector class and operators act on vectors inherited from the abstract class. Ideally, this separation allows the geophysicist to leverage the work of the mathematician without needing to understand the implementation details of the optimization method. The minimal set of object-oriented features of Fortran95 and its predecessors limited the potential separation between the physics and the solver. 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 2003, a more complete separation between solvers and operators can be achieved. By expanding the vector class definition, approaches such as the hybrid norm can be implemented.

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 estimating problems. Iterative solvers can become quite complex, quite quickly, 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 viewed 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 recent introduction of 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 2003, which is more complete object-oriented language.

In this paper, I show how to implement an abstract solver class in Fortran 2003. I begin by describing one method to implement an abstract operator-based iterative solver. I describe how the definition of a vector needs to be extended to support the hybrid norm. I then show how the abstract operator and vector classes can be implemented in Fortran2003.

OPERATOR-BASED OBJECT-ORIENTED SOLVERS

There are at least two different approaches one can take to solving the typical geophysical inversion problem. Harlan (1996), among others, takes a Bayesian approach to inversion. The Bayesian approach allows a natural inclusion of *a priori* statistical properties of the model. SEP (Claerbout, 1999) has traditionally taken an approach which is described as either classical, traditional, or deterministic. 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 $\mathbf{rr}$. Finally, we find the scaling factor of $\mathbf{rr}$ that will make $\mathbf{r} + \mathbf{rr}$ 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 it's 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 $\mathbf{rr}$, we need to calculate a dot product. In order to update the model and the residual, we will need to scale $\mathbf{rr}$ 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 class `vec` 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 class `vec`. 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.

There are several other functions that can be useful in a generic vector class. The ability to check that two vectors are from the same vector space can avoid many bugs. The ability to fill the vector with random numbers makes it easy to test whether or not an operator passes the dot product. Scaling and adding are often done together by defining a scale-add feature,

$$\mathbf{v} = \mathbf{a}\mathbf{v} + \mathbf{b}\mathbf{w}, \tag{2}$$

that updates a vector $\mathbf{v}$ by scaling it with a scaled version of a second vector $\mathbf{w}$ we can often improve performance. Finally, having the ability to make a copy of only the space a vector sits in rather than making copy of all elements can often improve performance and reduce storage requirements.

Operators

The base operator class contains the ability to perform a mapping from the vector-space of $\mathbf{m}$, its domain, to the vector space of $\mathbf{d}$, the operator's range (the forward), and vice versa. 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 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 $\mathbf{L}$ is actually the cascade of two operator $\mathbf{A}$ and $\mathbf{B}$,

$$\mathbf{L} = \mathbf{AB} \tag{3}$$

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

Combining operators

The number of different ways that 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 the row operation described above. 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 a problem that can be described as simply as equation 1. Achieving this simple form is often 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 finding the initial residual 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 $\mathbf{W_0}, \mathbf{W_1}$.

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

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

$$\begin{bmatrix} \mathbf{r_d} \\ \mathbf{r_m} \end{bmatrix} = \begin{bmatrix} \mathbf{W_1 L_1} \\ \mathbf{W_2 L_2} \end{bmatrix} \mathbf{S x}, \tag{8}$$

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

$$\mathbf{m} = \mathbf{m_0 S x}. \tag{9}$$

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.

SUPPORTING THE HYBRID NORM

The L2-norm is often a non-optimal choice both because of the non-Gaussian nature of noise in data and it's tendency to create smooth models with traditional regularization schemes. To improve model estimation, authors have suggested using either direct L1-norm approaches, or more commonly, Iteratively Reweighed Least-Squares (IRLS) to approximate different norms. IRLS attempts to find the best model at a non-L2 norm by a series of linearization s. Each non-linear iteration updates the weighting $\mathbf{W}$ of the residual. For example we can achieve L1 like behavior by updating the weighting operator through,

$$w(i) = \frac{1}{|r(i)|}. \tag{10}$$

IRLS methods tend to be difficult to use because the user must choose carefully the number of relinearizations and the numbers of steps between relinearizations carefully. (Claerbout, 2009) suggested an alternate approach that dynamically changes the weighting function every iteration and uses a Taylor expansion of the standard conjugate direction algorithm to update the solution. Further he suggests a different norm, the hybrid norm, that creates a smooth transition between the standard L2 problem and a L1 problem. Given an error function E and a residual vector $r(i)$ the hybrid norm is defined

$$E = \sum_i R^2 (\sqrt{1 + r^2(i)/R^2} - 1), \tag{11}$$

where **R** is a user supplied bad-data percentile.

Creating an inversion framework that supports a Taylor expansion approach to conjugate directions requires adding two additional features to our vector class description. First, we need to associate a norm to each vector. Second we need to be able to multiply a vector by another vector, element by element. Adding support for the hybrid norm requires more changes. A vector must now have a bad-data percentage associated with it, it must be able to find its *ith* percentile value, and create a vector with this value.

IMPLEMENTATION IN FORTRAN 2003

The Fortran 2003 standard makes Fortran a nearly complete object-oriented language. Fortunately (because it improves the compilers ability to optimize) and unfortunately, the object-oriented language components are not described in a very compact manner. The basic object-oriented construct ia a `type`. In Fortran 90, types could not contain function pointers and there was no inheritance concept. In Fortran 2003, a type can contain a reference to a function pointer. A type now is broken into two parts separated by `contains` statement. Above the `contains` statement variables are defined, below procedure pointers. For example, we can create a type `vec` which contains a function that can add one vector to another. The type is declared by:

```
type vec
  real, allocatable :: vals(:)
  contains
   procedure, pass :: add=>add_me
 end type
```

The `=>` keyword indicates that to access the function we should use the name on the left but the name of the procedure is found on the right. The `pass` keyword will be described later. Within the module that contains the `vec` definition, we need to define the `add_me` function:

```
 subroutine add_me(vec1,vec2)
   class(vec) :: vec1
   type(vec)  :: vec2
   vec2%vals=vec2%vals+vec1%vals
 end subroutine
```

Note that `vec1` is declared using the `class` keyword rather than the `type` keyword. The `class` keyword indicates that anything of type `vec` or anything that inherits from `vec` can call this function. We can access the `add` function through the standard Fortran `call` keyword:

```
 type(vec) :: vec1,vec2
 call vec1%add(vec2)
```

Note how the function definition has two arguments while the call description has a single argument. The `pass` keyword is the reason for the discrepancy. The `pass` keyword indicates that the type itself should be passed as the first argument.

An abstract type can also be constructed. An abstract type contains one or more function pointers that haven't been assigned. The abstract type can never be declared. Only types that inherit from it can be declared in a functional unit and only if all function pointers have been assigned. An abstract type is declared with the `abstract` keyword. The keyword `deferred` is used for all functions that will be assigned by inherited objects. In addition, you can define the interface of each function using the `abstract interface` construct. Below is an example of using these features:

```
type,abstract :: vector
  contains
  procedure(add_dec), pass, deferred :: add
end type

abstract interface
subroutine add_dec(v1,v2)
  class(vector) :: v1,v2
end subroutine
end abstract interface
```

We can declare a type that inherits from this `vector` class using the `extends` construct:

```
type, extends(vector) :: vec_real
 real :: vals(:)
 contains
 procedure, pass :: add=>add_real
end type
```

The function `vec_real` must have the same interface as the type it extends from with the exception of the first argument which now must be of type `vec_real`. Note how the abstract interface for the `add` function defines `v2` as a `class` object. To add two vectors we need them to be the same `type`. We can check the `type` of a class object using the `select type` construct:

```
subroutine add_real(v1,v2)
 class(vector) :: v2
 class(vector_real) ::v1
 select type(v2)
   type is(vector_real)
      v1%vals=v1%vals+v2%vals
   end select
end subroutine
```

Within the `type is` code block, `v2` is assumed to be of type `vector_real` and all components of `vector_real` can be accessed.

Fortran 2003 also provides a cleanup feature. The `finalize` keyword is called when an object is no longer needed (for example, when leaving a subroutine where it has been declared). In the example below, the `deleteit` function is used remove any memory associated with an object.

```
type, extends(vector) :: vec_real
 real :: vals(:)
 contains
 procedure, pass :: add=>add_real
 final ::  deleteit
end type
```

Finally, we need the ability to clone an abstract type. The `allocate` function now takes a keyword argument `source`. For example, we can create an object `v2` of the same type as `v1` even though `v1` is of an abstract rather than concrete type.

```
subroutine cloneit(v1,v2)
  class(vector),pointer :: v1,v2
  allocate (v2,source=v1)
end subroutine
```

With these extensions to the Fortran language, it is possible to completely separate operator writing from solver writing.

Current compiler limitations

Unfortunately, most Fortran compilers are still not Fortran 2003 compliant. Specifically, they lack the ability to copy from source (needed for cloning) and have yet to implement the `final` construct. By the end of 2010, both Intel and PGI promise to fully support these features.

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 hybrid-norm approach requires some modification from the standard vector class description but can easily be accommodated. The Fortran 2003 standard contains all of the object-oriented features needed to write an inversion library.

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.

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

Harlan, W. S., 1996, Promising research topics: http://sepwww.stanford.edu/oldsep/harlan /papers/seg96review.ps.gz.

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.

A new bidirectional deconvolution method that overcomes the minimum phase assumption

Yang Zhang and Jon Claerbout

ABSTRACT

Traditionally blind deconvolution makes the assumption that the reflectivity spike series is white. Earlier we dropped that assumption and adopted the assumption that the output spike series is sparse under a hyperbolic penalty function. This approach now here allows us to take a step further and drop the assumption of minimum phase. In this new method (what we called Bidirectional Sparse Deconvolution), We solve explicitly for the maximum phase part of the source. Results on both synthetic data and field data show clear improvements.

INTRODUCTION

In the previous report (Zhang and Claerbout, 2010), we introduced the spiking deconvolution problem using the hybrid norm solver (Claerbout, 2009a). Synthetic examples (Zhang and Claerbout, 2010) showed that given a minimum-phase wavelet, it retrieved the sparse reflectivity model almost perfectly even with a reflection series that is far from white, while conventional L2 deconvolution did a poor job. However, if the assumption of a minimum-phase wavelet was removed, the hybrid norm spiking deconvolution failed quickly and gave a poor result similar to the conventional L2 deconvolution.

In this paper, we still rely on the hybrid norm solver to retrieve the sparse model, but we use a slightly more complex formulation that avoids the minimum-phase wavelet constraint.

We start by realizing that any (mixed-phase) wavelet $C(Z)$ can be decomposed into a minimum-phase part $A(Z)$ and a maximum-phase part $B(1/Z)$ plus a certain time shift:

$$C(Z) = A(Z)B(1/Z)Z^k, \tag{1}$$

where $B(Z)$ is also a minimum-phase wavelet (therefore $B(1/Z)Z^k$ is a maximum-phase wavelet) and the exponent k is the order of $B(Z)$. This Z^k term makes the wavelet $C(Z)$ causal. In the time domain, (1) can be written as

$$c = a * b^r * \delta(n-k), \tag{2}$$

where b^r stands for the time reverse of series b.

Our original spiking deconvolution can find only a minimum-phase wavelet which has the same spectrum of real wavelet c. It can be defined as an inverse problem as follows:

$$[d]f_c = r, \tag{3}$$

where $[d]$ is the data convolution operator, and f_c is the unknown filter. In this formulation, the filter is the only unknown, the hybrid norm is applied on the residual term r to enforce

the sparseness constraint. In theory, the residual r itself is the reflectivity model. Such a method requires the wavelet in the data to be minimum-phase because only a minimum-phase wavelet has a causal stable inverse.

The following bidirectional deconvolution formulation utilizes a pair of conventional deconvolutions, trying to invert components a and b separately:

$$\begin{aligned} [(d * f_b^r)]\, f_a &= r_a, \\ [(d * f_a)^r]\, f_b &= r_b, \end{aligned} \tag{4}$$

in which f_a and f_b are the corresponding filters that corresponds to the inverses of a and b denoted above, the superscript r means time-reverse. The operator in each equation is the convolution operator. Again the hybrid norm is applied to r_a and r_b, and the reflectivity model is simply r_a plus a time shift. Notice that this is a non-linear inversion, since the operator itself depends on the unknown f_a and f_b. In practice we have to solve these two inversions alternately and therefore iteratively.

To understand the meaning of (4), let

$$d = m * c = m * a * b^r * \delta(n-k), \tag{5}$$

where m is the reflectivity model and the δ term is just a time shift. Assume f_a and f_b are perfectly known in the operators (which is not true in reality), i.e.

$$f_a * a = \delta(n), f_b * b = \delta(n)$$

Substituting (5) into (4), since

$$d * f_b^r = m * \delta(n-k) * a, \tag{6}$$

$$(d * f_a)^r = (m * b^r * \delta(n-k))^r = m^r * \delta(n+k) * b, \tag{7}$$

we have

$$\begin{aligned} [(m * \delta(n-k)) * a]\, f_a &= r_a, \\ [(m^r * \delta(n+k)) * b]\, f_b &= r_b. \end{aligned} \tag{8}$$

From (8) it is easier to see what is behind the bidirectional deconvolution formulation (4): It tries to separate the two parts of the wavelet, turning each one into a traditional deconvolution problem in which the wavelet (a, b) is always minimum-phase.

As with all non-linear estimation, iteration is required. Convergence is assured if the starting solution is close enough. We expect the traditional PEF for a and an impulse function for b to be a pretty good first guess. The following section shows several examples (complexity varies from low to high) illustrating the effectiveness and limitations of the method.

DATA EXAMPLES

Inverting a single wavelet

To verify the bidirectional deconvolution's ability to handle mixed-phase wavelets, we first set the input data to be a single wavelet, to see whether the data can be compressed to a single spike. We choose three types of wavelets as inputs:

1. a minimum-phase wavelet used in the previous report (Zhang and Claerbout, 2010), referred to as wavelet 1.

2. a wavelet that deviates slightly from minimum-phase: it models a simple marine ghost – a low frequency function passing through a time derivative at the source and another at the receiver. The low frequency function chosen is the convolution of two one-sided triangles.

3. a zero-phase wavelet created by convolving the minimum-phase with its own time-reverse wavelet. Such wavelet has identical a and b components, referred to as wavelet 3.

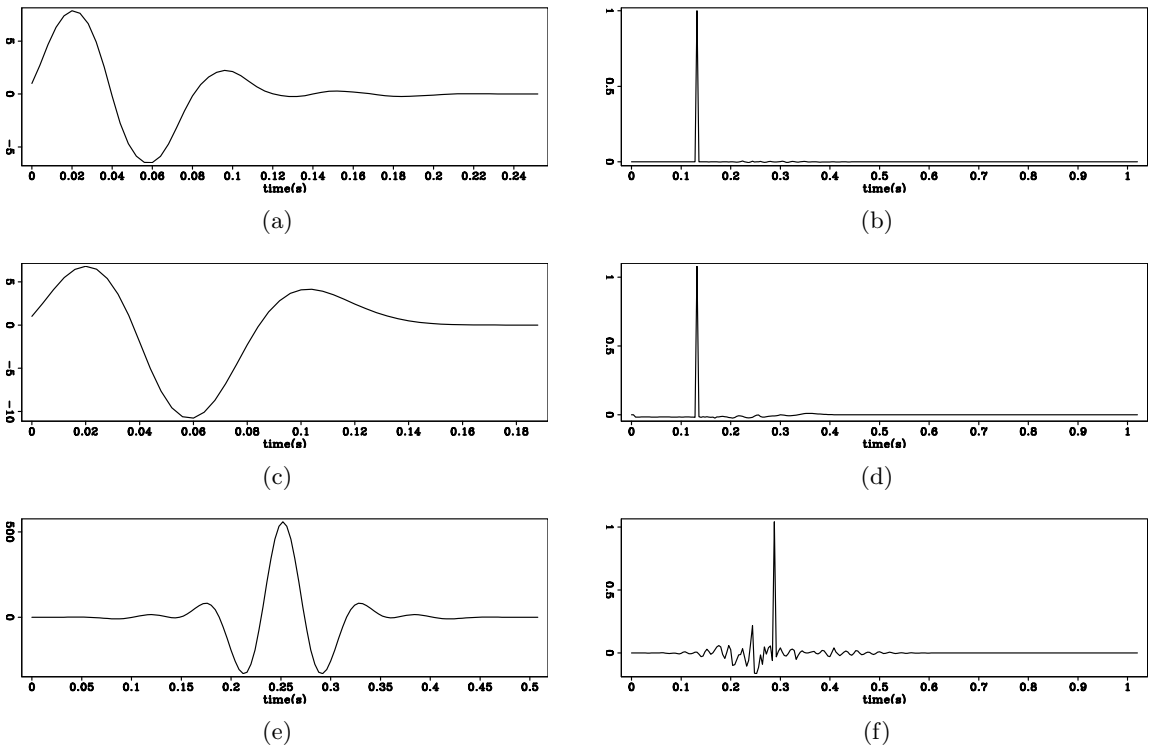


Figure 1: (a) Input wavelet 1 and (b) its deconvolution result. (c) Input wavelet 2 and (b) its deconvolution result. (e) Input wavelet 3 and (f) its deconvolution result. [**ER**] yang1/. minwavlet,mod-minwavlet,jonwavlet,mod-jonwavlet,symwavlet,mod-symwavlet

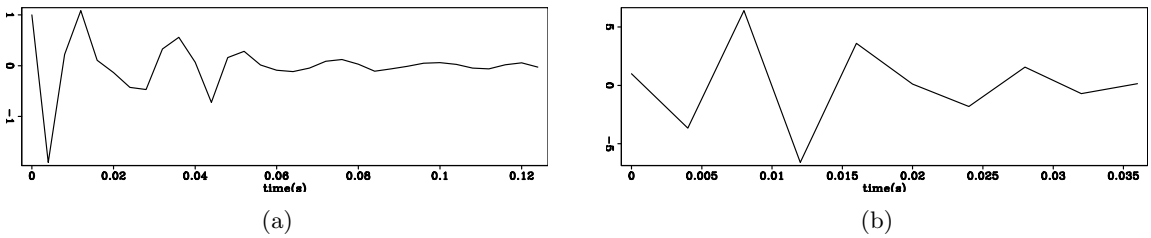


Figure 2: For the wavelet 3 inversion, (a) filter f_a; (b) filter f_b. [**ER**] yang1/. fita-symwavlet,fitb-symwavlet

Figure 1(a) 1(b), figure 1(c) 1(d) and figure 1(e) 1(f) show wavelets 1,2,3, and the results of reflectivity models respectively. In all 3 cases, our bidirectional deconvolution method is able to compress the wavelet into a spike.

Figure 2 shows the retrieved filters f_a and f_b from wavelet 3's inversion. Notice that f_a and f_b given by the inversion are different from each other, while ideally they should be the same, since a and b are the same when we create wavelet 3. This observation indicates that the solutions f_a and f_b of this method do not necessarily converge to the inverse of the initial a and b.

Inverting a synthetic trace

Next we try a more complex example where the data is generated by convolving each type of wavelet with a sparse reflectivity series. Figure 3 shows the reflectivity series.

Figure 4(a) 4(b), figure 4(c) 4(d) and figure 4(e) 4(f) show the data created using wavelets 1,2,3, and the recovered reflectivity models respectively. In all 3 cases, the reflectivity model is well recovered; however the polarity of the reflectivity model from wavelet 3 case is opposite to that of the real reflectivity model; this unexpected change of polarity shows the uncertainty of the convergence point in our non-linear formulation. We think this polarity change is not an issue in our blind deconvolution scenario.

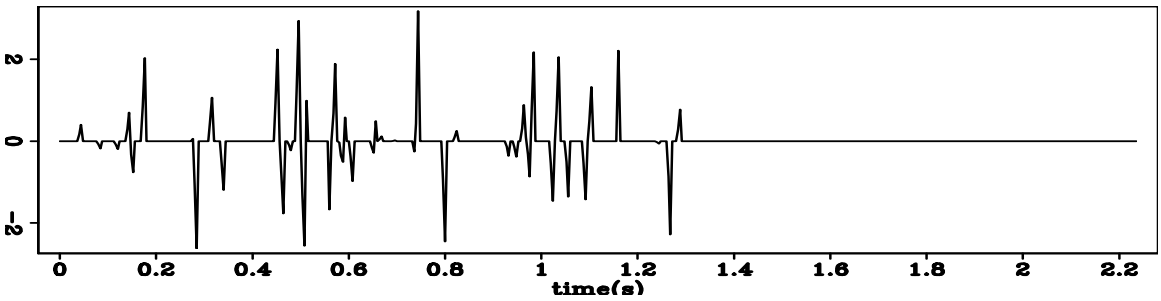


Figure 3: reflectivity model trace. [**ER**] yang1/. refl-trc

Inverting a 2D synthetic section

As in the previous report (Zhang and Claerbout, 2010), we use a 2D synthetic reflectivity model from Claerbout (2009b). Figure 5(a) shows the starting reflectivity model. Figure 5(b) shows the data generated by convolving the reflectivity model with wavelet 3. All traces use the same wavelet when generating the data, and all traces share the same wavelet when we are doing the deconvolution.

Previously the traditional sparse deconvolution failed on this example because of the symmetric wavelet; therefore, here we compare the old method and the bidirectional deconvolution method. Figure 6(a) shows the result using the old method (equation (3)). Figure 6(b) shows the result using the bidrectional deconvolution method (equation (4)). Comparing to the given model, the bidirectional deconvolution result is a spectacular improvement over the old one. Bidirectional deconvolution is a big improvement.

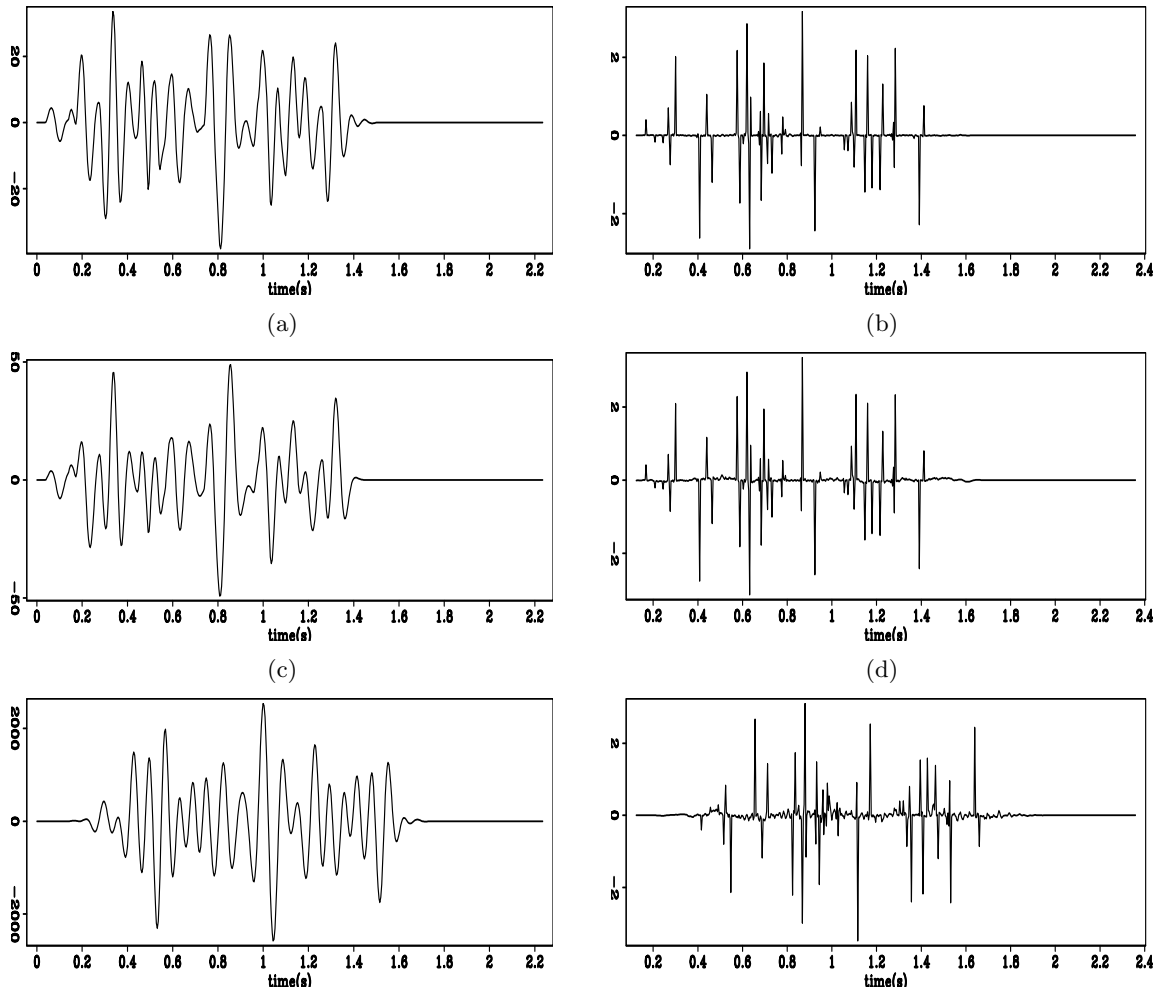


Figure 4: (a) The data trace generated using wavelet 1; (b) the recovered reflectivity model of (a). (c) The data trace generated using wavelet 2; (d) the recovered reflectivity model of (c). (e) The data trace generated using wavelet 3; (f) the recovered reflectivity model of (e). [**ER**] yang1/. data-mintrc,mod-mintrc,data-jontrc,mod-jontrc,data-symtrc,mod-symtrc

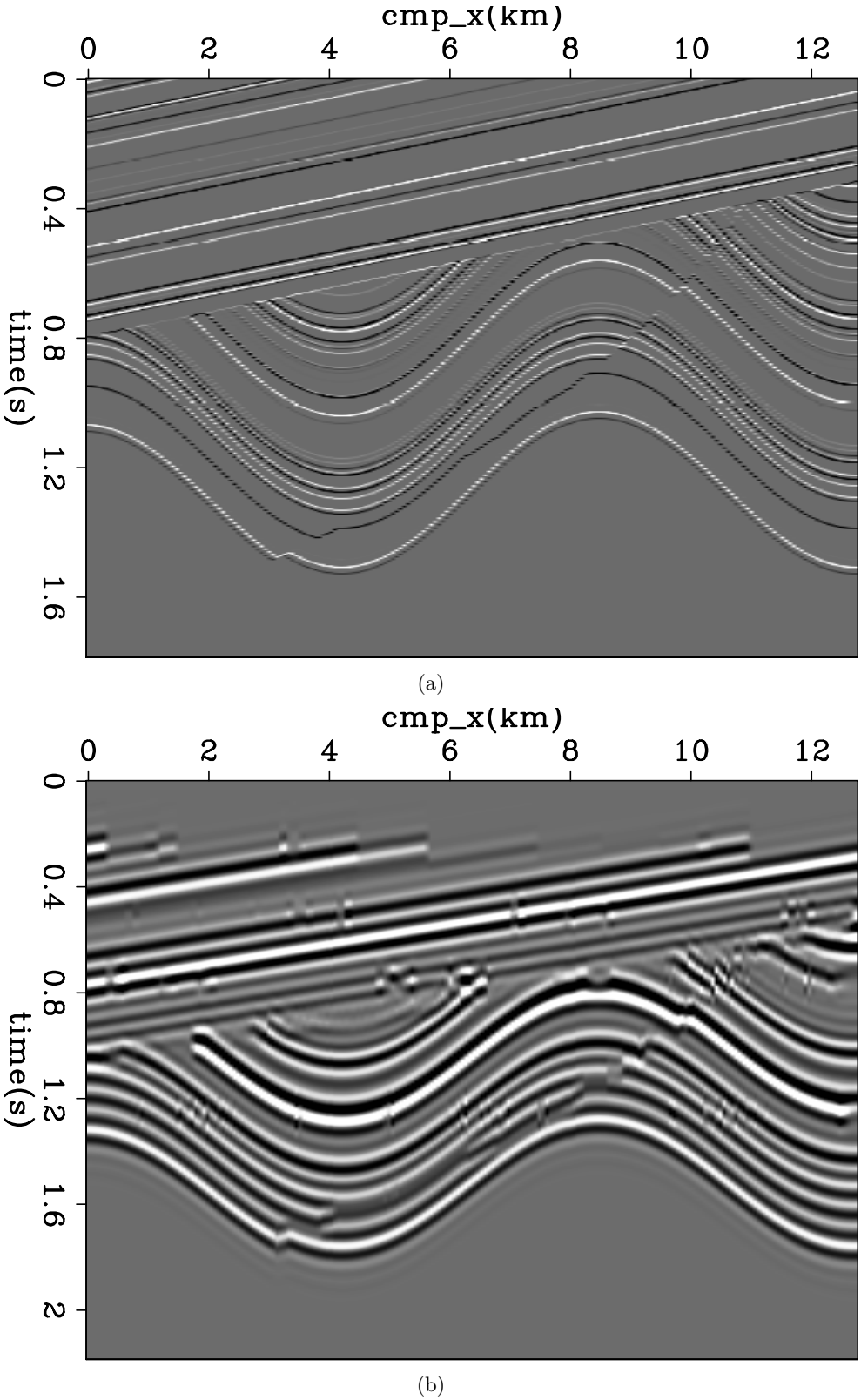


Figure 5: (a) The 2D synthetic reflectivity model; (b) the synthetic data generated using wavelet 3. [**ER**] yang1/. refl-jon,data-syn-sym-2d

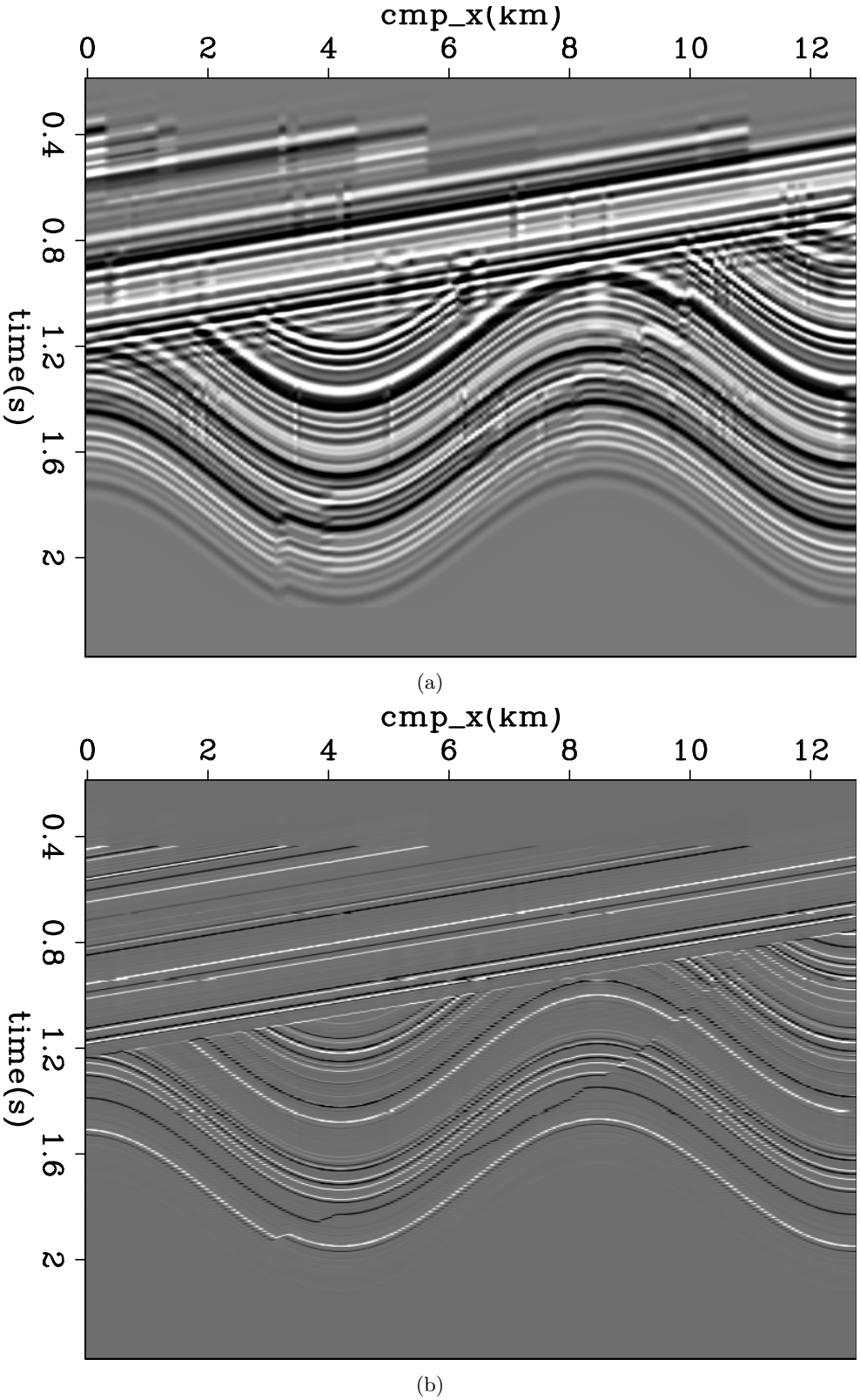


Figure 6: (a) reflectivity model retrieved from the original method; (b) reflectivity model retrieved from the bidirectional deconvolution method. [**ER**] yang1/. mod-sym-2d-old,mod-sym-2d

Inverting a 2D field section

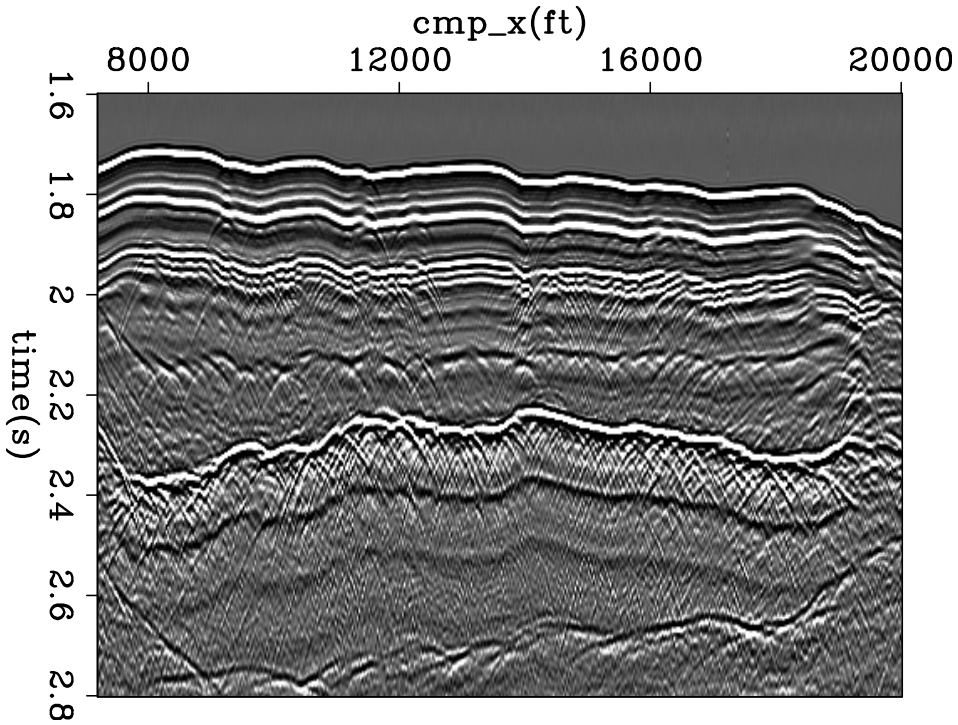
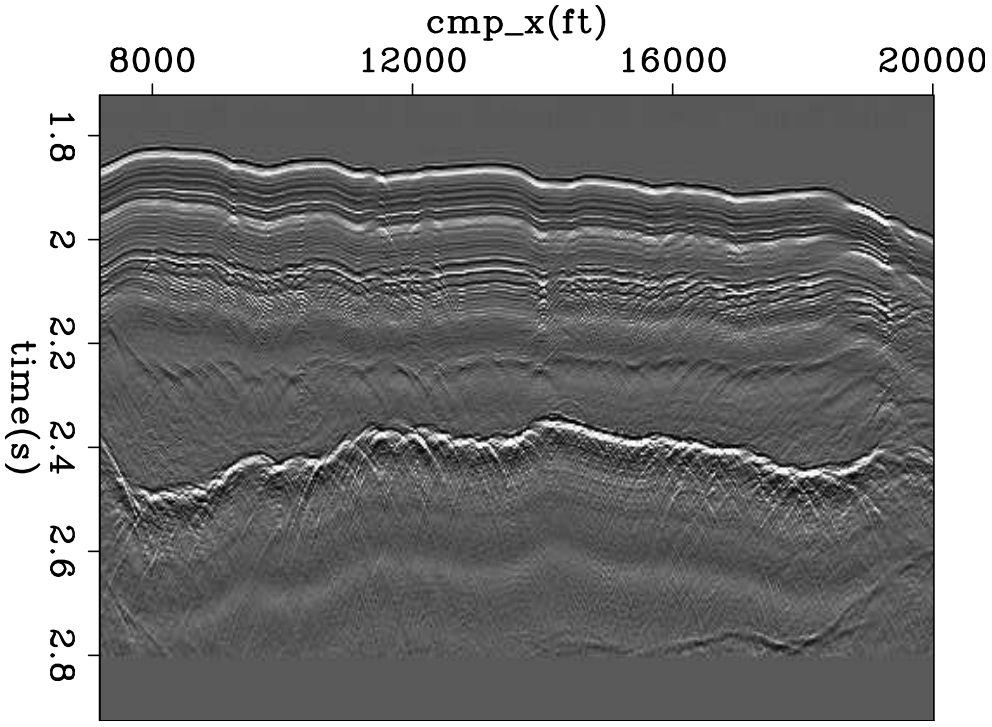


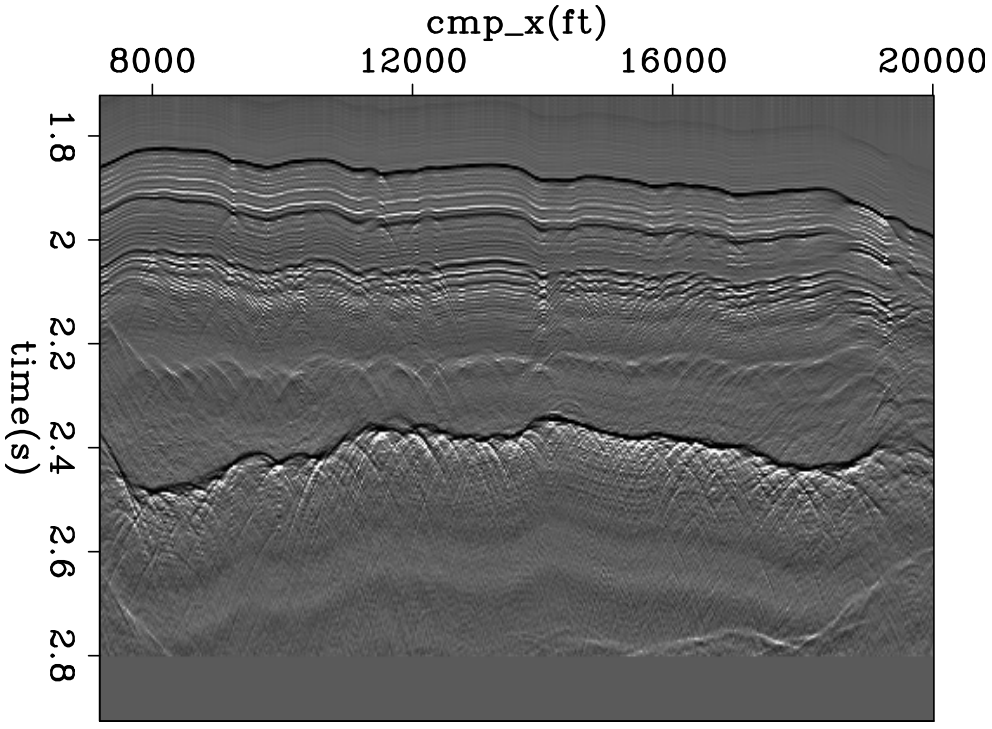
Figure 7: Input Common Offset data. [**ER**] yang1/. data-COF

The second example is a common-offset section of marine field data. Figure 7 shows the input data. Figure 8(a) shows the result using the old method. Figure 8(b) shows the result using the bidirectional deconvolution method.

The raw data in Figure 7 shows strong events like a double ghost (black, white, black). The traditional PEF result in Figure 8(a) shows strong events like doublets (black, white). The bidirectional deconvolution result in Figure 8(b) shows strong events like singlets (white). Examining Figure 8(b) we notice events at about 1.85s (black), 1.95s (black), 2.3s (white), 2.4s(black), 2.5s (mixed), and 2.8s (white). The unipolarity of individual suggests that a causal integration would produce the step functions we associate with impedence in a blocky model. Figure 9 is a first attempt to compute the impedence from the reflectivity in Figure 8(b). This was done by causal integration and some horizontal smoothing. Ideally, Figure 8(b) has only isolated white events and black events defining geologic boundaries. Time integrating these impulsive events should yield positive rectangle functions. Actually, the result we see in Figure 9 looks more like leaky integration of Figure 8(b). The small events present in Figure 8(b) apparently contains low frequency energy at the opposite polarity of that of the isolated impulses. We could thus regard Figure 9 as a failure. Instead we regard it as an inverse problem that we have not yet correctly posed. The failure arises because the raw data fails to contain the required low frequencies. Were we to replace small values in Figure 8(b) by zeros, we might have obtained a result more to our liking. We need to formalize the inverse problem and reduce it to the usual situation which is how much to regard the data as perfect, and how to allow imperfection to be overcome by methodology that tends us to blocky models.



(a)



(b)

Figure 8: Given the common offset data in Figure 7, (a): reflectivity model retrieved from the original method; (b): reflectivity model retrieved from the bidirectional deconvolution method. [**ER**] yang1/. mod-COF-decon-old,mod-COF-decon

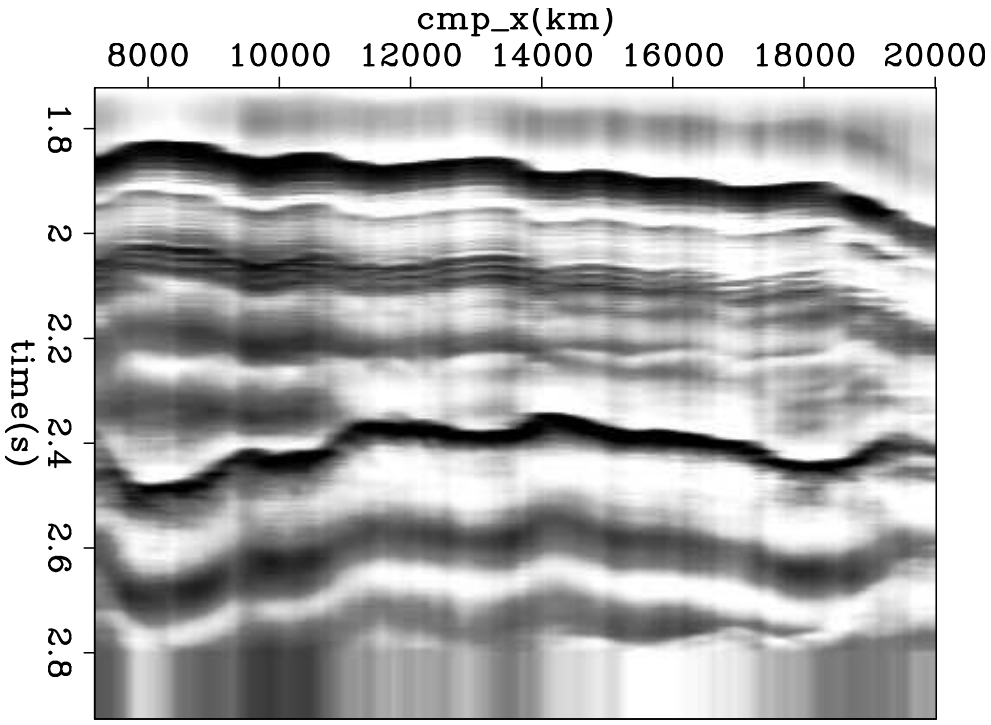


Figure 9: Causal time integration of the reflectivity in figure 8(b). This should be the impedence. [**ER**] yang1/. sm-integ-mod-COF-decon

CONCLUSION

We demonstrate what we anticipated theoretically that we can overcome the minimum phase assumption in blind deconvolution. Our process is non-linear, but (we claim) not extremely so. To be successful it does require a non-Gaussian distribution of impulses. Likewise, the iteration has a few adjustable parameters which makes its use a little more difficult, but we do not anticipate serious difficulties in practice. One interesting phenomenon about the bidirectional deconvolution (Figure 1(e) 1(f) and figure 2) is that it was able to compress a mixed-phase wavelet to a spike but without obtaining the correct causal and anti-causal parts. We do not yet understand this. In addition, it is more costly because it requires multiple iterations.

FUTURE WORK

Having had good fortune here introducing the anti-causal PEF and earlier explicitly estimating a portion of the data not fitting the convolutional model (Zhang and Claerbout, 2010) , it is natural to try introducing both at the same time. That takes into account the fact that a part of the input data does not fit the convolution model:

$$\left\{ \begin{bmatrix} (d * f_b^r) & -\mathbf{I} \\ \mathbf{0} & \epsilon\mathbf{I} \end{bmatrix} \begin{bmatrix} f_a \\ m_a \end{bmatrix} \approx \begin{bmatrix} r_{da} \\ r_{ma} \end{bmatrix}, \right. \tag{9}$$

$$\left\{ \begin{bmatrix} (d * f_a)^r & -\mathbf{I} \\ \mathbf{0} & \epsilon\mathbf{I} \end{bmatrix} \begin{bmatrix} f_b \\ m_b \end{bmatrix} \approx \begin{bmatrix} r_{db} \\ r_{mb} \end{bmatrix}, \tag{10}$$

in both matrices on the upper left is the data convolution operator, f_a and f_b are the filters, and m_a and m_b are the reflectivity models. The parameter ϵ indicates the strength of the regularization. We apply the hybrid norm on model residuals r_{ma} and r_{mb} to enforce sparseness. Although the extra parameter tuning (ϵ) is undesirable, we expect to get more successful result using this more advanced formulation.

ACKNOWLEDGMENTS

The authors thank the sponsors of Stanford Exploration Project for the financial support, and also thank Bob Clapp, Antoine Guitton and Luis Canales for fruitful discussions.

REFERENCES

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

Claerbout, J. F., 2009b, Basic earth imaging.

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

Short note: Three dimensional deconvolution of helioseismic data

Chris Leader, Jon Claerbout, and Antoine Guitton

ABSTRACT

This is a short note on helioseismic deconvolution. Herein results are presented by deconvolving helioseismic data with a calculated impulse response in 3D to help determine source information in the shallow solar interior. Tentatively it can be concluded that there the solar acoustic energy is close to uniformaly distributed throughout the convective envelope.

INTRODUCTION

Deconvolution is a technique that the majority of geophysicists are very familiar with, since the presence of a source function inherent within recorded seismic data is an issue that permeates all areas of seismology. Deconvolution simply seeks to remove this source function from the recorded data, leaving the response of the Earth (and any other instrument responses from the recording system.) The main problem within deconvolution is the estimation of the source function, since often these are non-stationary and non-minimum phase, and as such typical 'batch processing' methods such as spiking and predictive deconvolution will make these assumptions, along with several others.

Helioseismology attempts to determine certain aspects of the solar interior by studying acoustic oscillations and disturbances on the surface of the sun. The source function for these oscillations is attributed to turbulence within the convective envelope of the outer sun, and as such is stochastic. Generally helioseismologists work with these source functions by cross-correlating passively observed seismograms and calculating time-distance curves by picking lags.

Rickett and Claerbout (2001) showed that is possible to estimate the solar impulse response by using spectral factorisation techniques, notably by applying one-dimensional Kolmogorov spectral factorisation theory to the 3D oberserved data by using helical boundary conditions (Claerbout, 1998). This short note is an extension of Rickett's work, whereby the postulation is that if we deconvolve the raw data with the three dimensional solar impulse response information about the location and separation of these solar source regions may be revealed.

The helioseismic data used was the SOHO/MDI dataset. The data was transformed to Cartesian coordinates by projecting the high resolution data from an approximate 18 degree square onto a tangent plane. Time sampling for this survey is 60 seconds, and after pre-processing the sampling in space over the solar surface is largely regular and set at 824,800 m.

INTRODUCTION TO HELIOSEISMOLOGY

Helioseismology is a subset of seismology and is unique in the fact that the recorded data is not terrestrial. By recording surface solar oscillations and accounting for Doppler shifts it is possible to constrain information - such as velocities and distances - about the shallow solar interior. This problem is more complex than most terrestrial surveys since there are now problems such as a spherical target, unknown source function and location, and the fact the sun is moving at significant rate (giving a Doppler shift.) However, by using passive data techniques such as cross-correlation and spectral factorisation it is possible to gain valuable insight into solar properties.

Often large scale trends are delineated by decomposing the stochastic wavefields into spherical harmonics (Harvey, 1995), which works well for studying macro solar trends. However to describe small scale events, harmonics of a high order need to be computed, such as is done with cosmic microwave background (CMB) studies, however focusing on smaller areas using this technique is inefficient. What is often done instead is to cross correlate oscillatory dopplergram traces, since the lags acquired from doing this can give information, such as velocity, about ray paths travelling between the two trace locations (Duvall, 1993).

ACQUIRING THE SOLAR IMPULSE RESPONSE

The method used to acquire the solar response is described in detail in Rickett and Claerbout (2001). The premise is that Kolmogorov spectral factorisation is used since this is an efficient method of constructing a minimum phase time domain function from a given power spectrum (Kolmogorov, 1939). The theory is 1D, however as shown in Claerbout (2001) by applying helical boundary conditions it is possible to model the dataset as a long 1D trace by applying a sequence of lags, and then the factorisation can be applied and the 3D data reconstructed. The raw data and the factorised impulse response can be seen in Figure 1 and Figure 2 respectively.

SOLAR DECONVOLUTION IN 3D

Deconvolution in seismology typically seeks to remove the source signature from the recorded data. For 3D data this can be done in multiple ways: as a three way convolution integral in time, as a three way multiplication in the Fourier domain, or as 1D Fourier multiplication in helical coordinates. In this case the solar impulse response has been estimated using spectral factorisation, and by deconvolving the raw data with this response we can find information about the source - namely signature and location.

In frequency space the stochastic oscillation model can be described as the following multiplication

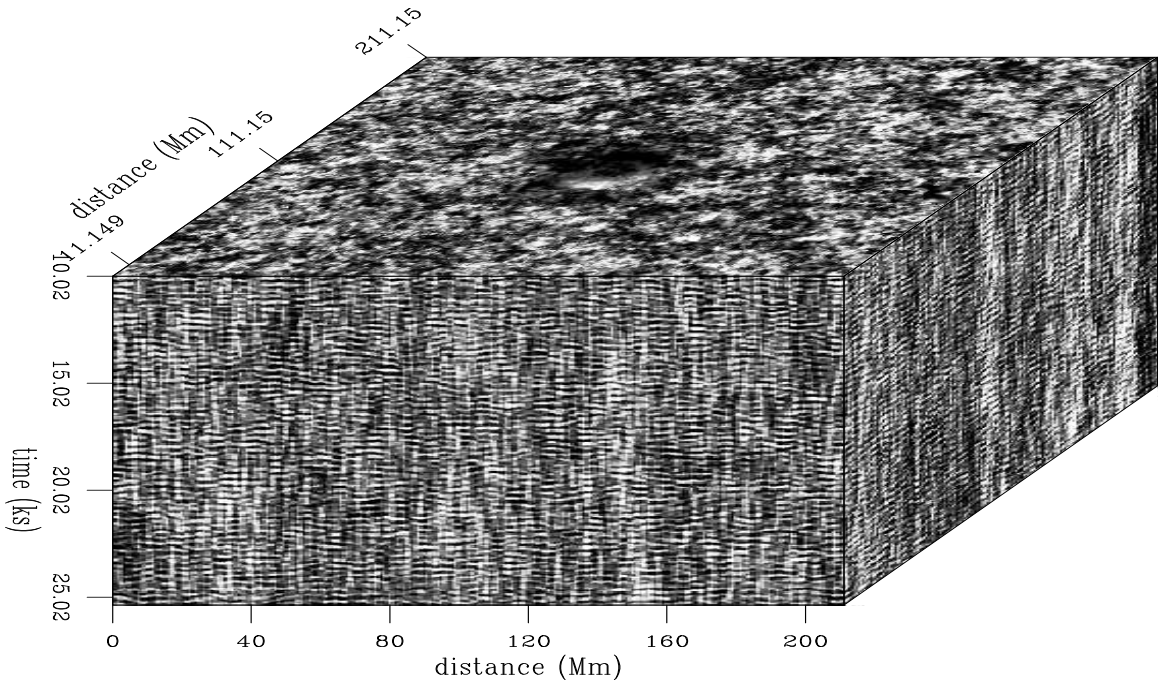


Figure 1: The raw helioseismic data, with a sun spot in the centre. [**ER**] chris1/. shortcube

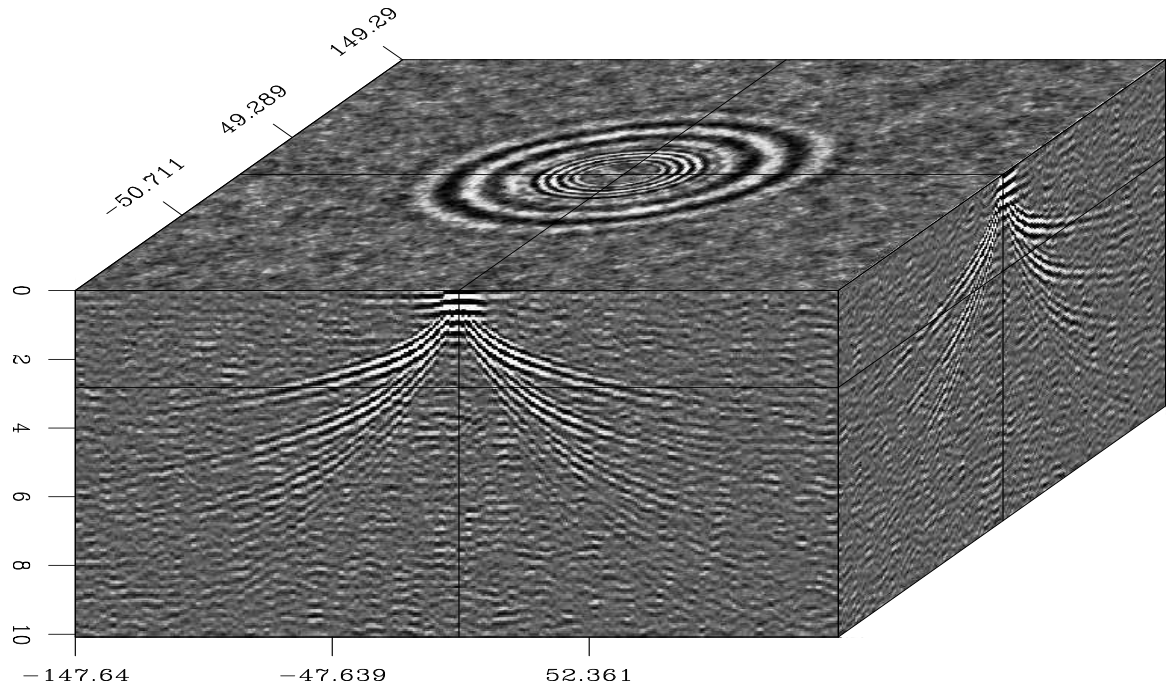


Figure 2: The solar impulse response. [**ER**] chris1/. kolcube

$$D(k_x, k_y, \omega) \ = S(k_x, k_y, \omega) G(k_x, k_y, \omega), \tag{1}$$

where D is the raw data, S the source function and G the impulse response. The 3D deconvolution can then be applied as a division in frequency, and then transformed to time (Rickett, 2001).

This method suffers from the fact that by dividing the input data D by G, any small or zero values in G will cause large perturbations in the solution for S. This is a problem addressed many, many times in geophysics, and one solution to helping to constrain the estimation is to add a small amount of white noise (a constant in frequency space) to the denominator, ensuring a maximum possible value in the output (Claerbout, 2001).

$$S(k_x, k_y, \omega) \ = G(k_x, k_y, \omega) / (D(k_x, k_y, \omega) + \epsilon), \tag{2}$$

Part of the usual challenge of deconvolution is choosing an appropriate value for ϵ such that the final image has not been overly steered (Claerbout, 2001). To ensure the 3D Fourier deconvolution was working correctly a synthetic model was produced, convolved with the impulse response and transformed to the time domain. This was then deconvolved with the impulse response, and the initial model was recovered clearly, with the exception of some Gibbs' artifacts due to the domain transformations and truncation of the impulse response. Subsequently the deconvolution part of this process was applied to the raw solar data, and a 3D volume acquired.

RESULTS AND FUTURE WORK

When using too low an ϵ value (or none at all) then a lot of low frequency noise was visible in the deconvolved image and this energy dominated any smaller events. When using $\epsilon = 0.01$ Figure 3 is produced. This initial deconvolved image is noisy and no discernible areas of high amplitude contrast are noticeable, with the exception of the sun spot, as was visible in the raw data. A potential conclusion from this is that source regions are too poorly separated in time and space to be visible, or that all shallow source regions operate at a similar power giving no conclusive separations after deconvolution.

The next stage for this concept will be to further tune ϵ and experiment with smaller sections of the data and also to test with balancing the data and the impulse response. One way could be to window the impulse response and use a smaller section for the deconvolution, and also to focus on smaller time windows.

REFERENCES

Claerbout, J., 1998, Multidimensional recursive filters via a helix: Geophysics, **63**, 1532–1541.

———, 2001, Image estimation by example: Society of Exploration Geophysicists.

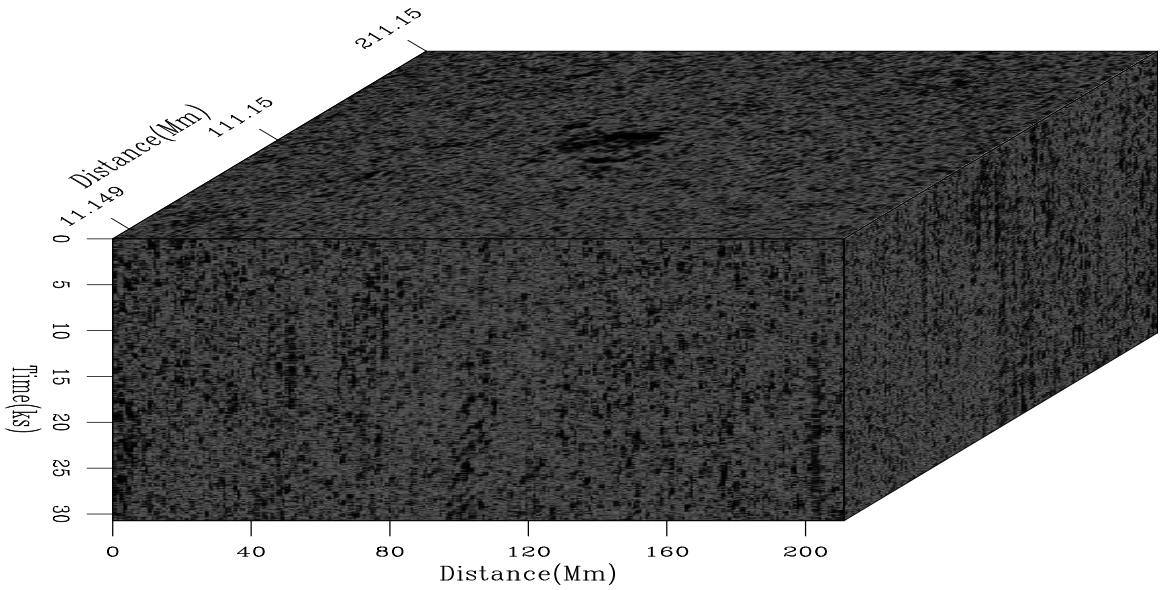


Figure 3: The deconvolved data. [**ER**] chris1/. decon

Duvall, T. e. a., 1993, Time distance helioseismology: Nature, **362**, 430–432.

Harvey, J., Oct. 1995, Sounding out the sun: Physics today, 32–38.

Kolmogorov, A., 1939, Sur l'interpretation et l'extrapolation des suites stationnaires: C.R Acad. Sci., **208**, 2043–2045.

Rickett, J., 2001, Spectral factorisation of wavefields and wave operators: Stanford University.

Rickett, J. and J. Claerbout, 2001, Calculation of the acoustic solar impulse response by multi-dimensional spectral factorization: Solar Physics, **192**, 203–210.

Implementing implicit finite-difference in the time-space domain using spectral factorization and helical deconvolution

Ohad Barak

ABSTRACT

The method of modeling wavefield propagation with an implicit finite-difference approximation to the two-way acoustic isotropic wave equation, using spectral factorization and helical deconvolution, exhibits instability of the propagating wavefield as the time step is increased. In this study, I test several potential sources of the instability problem: the implicit finite-difference scheme itself, the precision of the floating point representation of the filter coefficients, the number of filter coefficients, and the spectral factorization method. None of these issues is the cause for the apparent instability.

INTRODUCTION

Implicit finite-difference methods are inherently more stable than explicit ones. This attribute enables us to increase the time step size (and consequently decrease computation time) while retaining stability of the wavefield. In the previous SEP report (Barak, 2010) I showed that by using spectral factorization and the helix transform, the propagation of a wavefield using an implicit finite-difference approximation of the two-way acoustic wave equation can be achieved by a set of deconvolution operations of filter coefficients applied to the wavefield. Through testing, I have found that despite the theoretical stability advantage of the implicit finite-difference scheme which I used for propagation, the resulting wavefield becomes more dispersive as the time step increases (to the point that the wavefield is no longer useful), and also that beyond a certain time step size - the wavefield diverges.

The increased dispersion of the implicit finite-difference scheme in comparison to an explicit scheme is an attribute of the scheme itself. This is not a fundamental problem, since some of this dispersion can be alleviated simply by using a higher order approximation. However, the causes of the instability of the wavefield beyond a certain time step size remained unclear. In order to understand the reasons behind the instability, I tested several hypotheses for its causes. These were:

1. The implicit finite-difference approximation itself.
2. The precision of the floating point representation of the filter coefficients.
3. The number of filter coefficients.
4. The spectral factorization method.

First I will review the method by which wave propagation can be done by deconvolutions with spectrally factorized filters of a finite-difference approximation, and then I will go over the various tests I carried out to try and determine the causes for the instability problem.

REVIEW OF METHODOLOGY

The two-way acoustic wave equation in one dimension reads:

$$\frac{\partial^2 P}{\partial t^2} = C^2 \frac{\partial^2 P}{\partial x^2}. \tag{1}$$

The central implicit finite-difference approximation I used for the propagation tests was 2nd order in time and 2nd order in space:

$$\begin{aligned} \frac{P_x^{t+1} - 2P_x^t + P_x^{t-1}}{\Delta t^2} &= \frac{C^2}{4\Delta x^2}[\left(P_{x+1}^{t+1} - 2P_x^{t+1} + P_{x-1}^{t+1}\right) \\ &+ 2\left(P_{x+1}^t - 2P_x^t + P_{x-1}^t\right) + \left(P_{x+1}^{t-1} - 2P_x^{t-1} + P_{x-1}^{t-1}\right)], \end{aligned} \tag{2}$$

where P is the pressure wavefield, t and x are the time and space coordinate indices, and Δt and Δx are the temporal and spatial step sizes. Note that this approximation is based on the Crank-Nicolson method, and so the spatial derivative is balanced between the three time steps: $t-1$, t, and $t+1$, where the central time index t has twice the weight of the other two time indices.

In order to propagate the wavefield, the pressure values at time $t+1$ must be equated to the values at times t and $t-1$. The linear system which must then be solved has the form:

$$\begin{aligned} \begin{pmatrix} U_0 & U_1 & 0 & 0 \\ U_1 & U_0 & U_1 & 0 \\ 0 & U_1 & U_0 & U_1 \\ 0 & 0 & U_1 & U_0 \end{pmatrix} \begin{pmatrix} P_1^{t+1} \\ P_2^{t+1} \\ P_3^{t+1} \\ P_4^{t+1} \end{pmatrix} &= \begin{pmatrix} V_0 & V_1 & 0 & 0 \\ V_1 & V_0 & V_1 & 0 \\ 0 & V_1 & V_0 & V_1 \\ 0 & 0 & V_1 & V_0 \end{pmatrix} \begin{pmatrix} P_1^t \\ P_2^t \\ P_3^t \\ P_4^t \end{pmatrix} \\ &+ \begin{pmatrix} W_0 & W_1 & 0 & 0 \\ W_1 & W_0 & W_1 & 0 \\ 0 & W_1 & W_0 & W_1 \\ 0 & 0 & W_1 & W_0 \end{pmatrix} \begin{pmatrix} P_1^{t-1} \\ P_2^{t-1} \\ P_3^{t-1} \\ P_4^{t-1} \end{pmatrix}. \end{aligned} \tag{3}$$

For simplicity, we can combine all the constants into one: $\alpha = \frac{C^2 \Delta t^2}{4 \Delta x^2}$. The matrix coefficients in equation 3 (the finite-difference weights) are then:
$U_0 = 1 + 2\alpha, \qquad U_1 = -\alpha;$
$V_0 = 2 - 4\alpha, \qquad V_1 = 2\alpha;$
$W_0 = -1 - 2\alpha, \qquad W_1 = \alpha.$

In shorter notation, equation 3 reads:

$$UP^{t+1} = VP^t + WP^{t-1}. \tag{4}$$

The solution of this linear system is:

$$P^{t+1} = U^{-1} \left(VP^t + WP^{t-1} \right). \tag{5}$$

To solve this system, we must perform polynomial division. The system is tridiagonal (and easily solvable) only for one dimension. For multiple dimensions, matrix U is block diagonal. Additional non-zero elements appear at a certain offset from the diagonal, making the solution process more complicated. However, using spectral factorization, the finite-difference weights of matrix U (which pertain to time $t+1$) can be factorized into a set of causal filter coefficients u and its time reverse $u^{'}$. Using the helical approach to deconvolution, equation 5 can be recast as:

$$P^{t+1} = (u^{'} u)^{-1} \left(VP^t + WP^{t-1} \right); \tag{6}$$

$$P^{t+1} = u^{-1} (u^{'})^{-1} \left(VP^t + WP^{t-1} \right). \tag{7}$$

Polynomial division is comparable to deconvolution. This means that the polynomial division in equation 5 can be achieved by a set of two deconvolutions of the data by the spectrally factorized coefficients u of matrix U. One deconvolution is done along the data in the reverse direction (application of the adjoint of the filter):

$$y_k = x_k - \sum_{i=1}^{N_u} u'_i y_{k-i}, \tag{8}$$

where u' is the time reversed filter coefficients of u. The other deconvolution is done in the forward direction:

$$x_k = y_k - \sum_{i=1}^{N_u} u_i x_{k-i}. \tag{9}$$

I used the SEPlib module `polydiv`, which uses the helical coordinates to perform the deconvolutions (the polynomial division) in equations 8 and 9.
The wavefield propogation is done by the following sequence:

1. Spectrally factorize the coefficients of matrix U.
2. Multiply the saved wavefield at time $t-1$ by the coefficients of matrix W.
3. Multiply the saved wavefield at time t by the coefficients of matrix V.
4. Sum the results of the previous 2 steps into a result vector.
5. Deconvolve the result vector by the time-reversed factorized filter coefficients $u^{'}$ (eq. 8).
6. Deconvolve the result vector by the factorized filter coefficients u (eq. 9).

Steps 2 - 6 are repeated for each time step. The inputs of the spectral factorization are the finite-difference weights of the matrix U (in Eq. 3), and the outputs are coefficients of a minumum phase filter u. Since I used a constant velocity in all propagation tests, the finite-difference weights are constant, and the filter is stationary.

IMPLEMENTATION OF METHODOLOGY WITH INCREASING TIME STEP SIZE

Figures 1(a)-1(f) show how wave propagation in one dimension using the implicit scheme from Eq. 2 and spectral factorization fails when the time step size is increased beyond a certain limit. The horizontal axis is time, and the vertical is distance. On the left the propagation is done using a linear equation system solver, and on the right is the result of deconvolving the wavefield with the filter coefficients obtained from spectral factorization. At smaller time steps, the two solutions are similar. The increasing dispersion with increasing time step size is apparent in both solutions. However, once the time step exceeds $5ms$, the wavefield propagated by deconvolution diverges, whereas the wavefield propagated by the "standard" linear system solver exhibits additional dispersion, yet remains stable.

Figures 2(a)-2(f) show the same kind of comparison as Figures 1(a)-1(f), except that here a small ϵ value was added to the central finite-difference weight (U_0) which was sent as an input to the spectral factorizer:

$$U_0 = 1 + 2\alpha + \epsilon, \qquad U_1 = -\alpha.$$

This results in a filter with slightly different coefficients, and with this filter the propagation is stable (with added dispersion). The value of ϵ required to maintain stability increases as the time step size increases. So far I've been unable to determine the relation between the value of the time step and the value of ϵ, but I know it is not arbitrary. If ϵ is too large, the result is a low-frequency dispersion which seems to initially precede the wavefield, as shown in Figure 3. Afterwards, the wavefield loses amplitude until eventually it disappears altogether.

While the addition of some ϵ value does stabilize the wavefield, the flip side is that it causes wrong propagation kinematics. This is a direct result of the artificial increase of the central finite weight. The incorrect kinematics can be seen in Figure 2(d) when looking at the wavelet as it reaches the edge at the 4 second mark. The arrival time of the wavelet is retarded as ϵ increases.

A similar phenomena occurs in 2D. In Figure 4 the effect of increasing the time step size from $\Delta t = 5ms$ to $10ms$ is shown. The increase causes the wavefield to diverge. Adding $\epsilon = 0.005$ to the central finite-difference weight, as in Figure 5, alters the filter coefficients obtained by spectral factorization, and enables stable propagation, with a slight time retardation of the wavefront. If ϵ is too large, then an unusual dispersion pattern appears. As the time step is increased further (Figure 6), the value of ϵ required for stable propagation increases as well, as does the time retardation of the wavefront. With too large an ϵ value the unusual dispersion pattern appears.

Summary of current implementation

An increase in the time step size causes the wavefield to diverge after a certain number of propagation steps. This divergence can be avoided - by artificially increasing the value of the central finite-difference weight. This correspondingly increases the zero-lag coefficient of the factorized filter, making it more dominant in comparison to the other filter coefficients. The result is stable propagation, albeit with much dispersion owing to the finite-difference approximation itself. The exact minimum value required for ϵ which ensures stable propagation is difficult to ascertain. Too large a value and an odd dispersion pattern unlike that of standard numerical dispersion begins to appear. The addition of ϵ to the central finite-difference weight also has the rather unfortunate effect of ruining the propagation kinematics.

EFFECT OF FLOATING POINT PRECISION

As a result of the tests shown in the previous section, I concluded that the reason for the unstable propagation at large time step sizes had to do with the spectral factorization, and not with the finite-difference approximation. One of the characteristics of the filter coefficients produced by the spectral factorization algorithm I used (SEPlib module `wilson`) is that they are very small. The smaller ones can reach 10^{-20}. Biondi and Clapp (pers. comm., 2010) suggested that I attempt to use double precision variables instead of single precision, in order to see whether the precision of the representation of the filter coefficients is indeed an issue. Ronen (pers. comm., 2010) also suggested trying to do the opposite - reduce the precision and see whether that would have a degrading effect on the propagation.

To achieve double precision, both the spectral factorization algorithm and the helical deconvolution module had to be rewritten to include double precision variables. The wavefield itself was also composed of double precision variables. Furthermore, I used a 4th order in space, 2nd order in time approximation for this test. Figures 7(a)-7(d) show the comparison between propagation with single precision (left) and double precision (right). For the top Figures I used $\epsilon = 0$, and for the bottom ones $\epsilon = 0.01$. The time step Δt was $10ms$ for all Figures. The results for single and double precision are identical for this time step, and

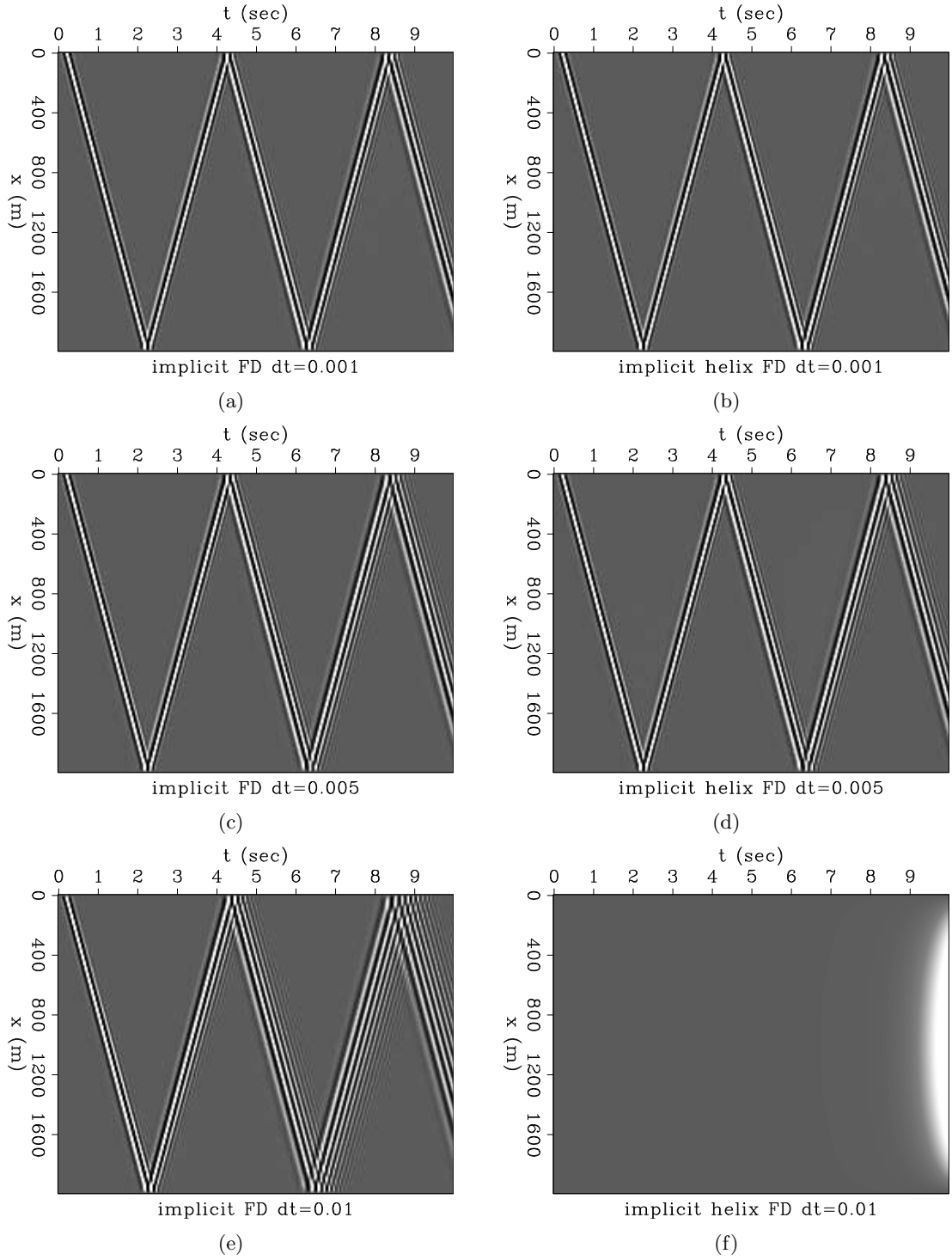


Figure 1: 1D Implicit (left) vs. Helical Implicit (right) finite-difference with constant velocity = $1000m/s$. Horizontal axis is time, and the vertical axis is distance. Source is a Ricker wavelet with central frequency = $12.5Hz$. The time step size is $\Delta t = 1ms$ for the top Figures, $5ms$ for the center Figures, and $10ms$ for the bottom Figures. $\Delta x = 10m$.[**ER**] ohad1/. impA-a,impA-b,impA-c,impA-d,impA-e,impA-f

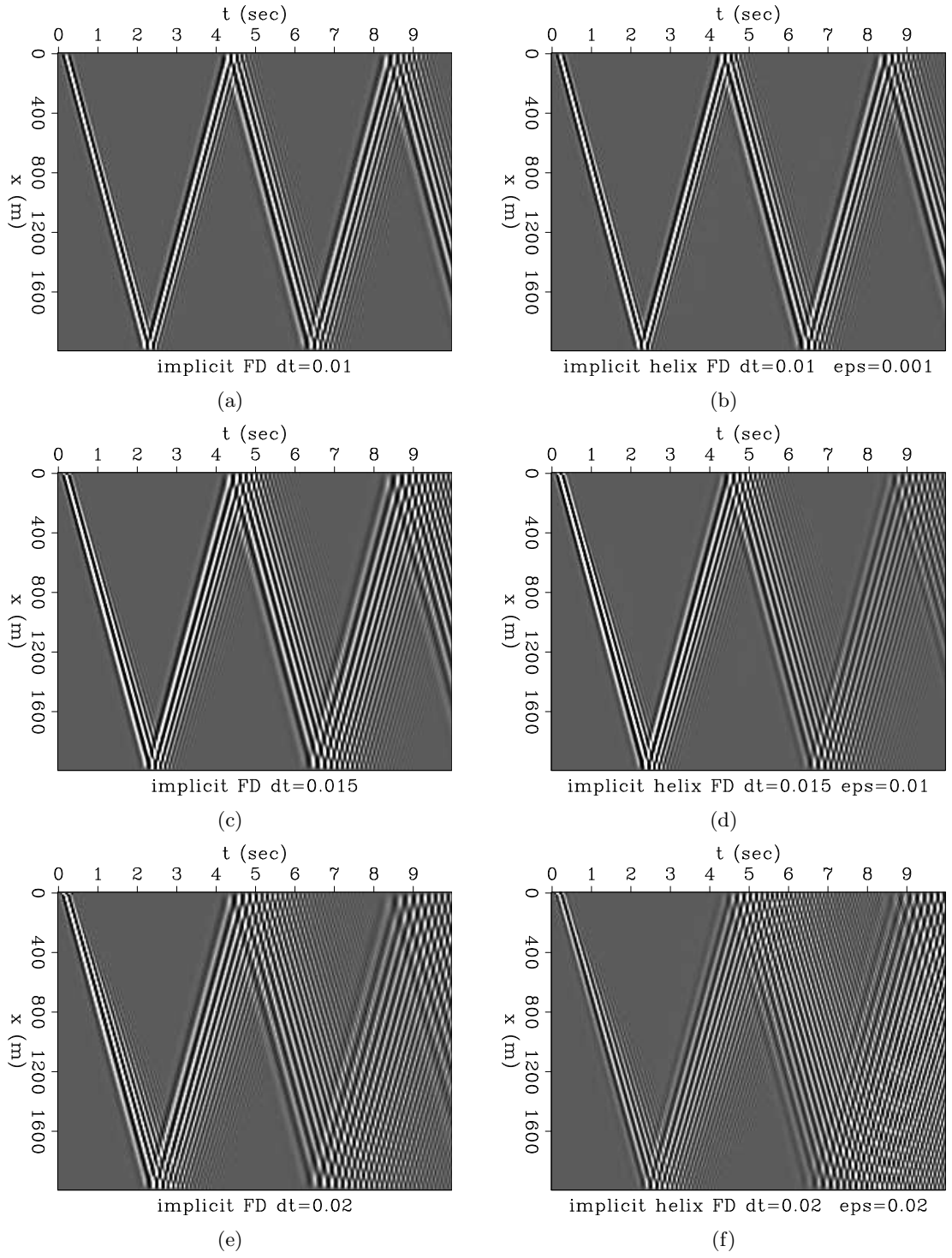


Figure 2: 1D Implicit (left) vs. Helical Implicit (right) finite-difference with constant velocity = $1000m/s$. Horizontal axis is time, and the vertical axis is distance. Source is a Ricker wavelet with central frequency = $12.5Hz$. The time step size is $\Delta t = 10ms$ for the top Figures, $15ms$ for the center Figures, and $20ms$ for the bottom Figures. Top right $\epsilon = 0.001$; center right $\epsilon = 0.01$; bottom right $\epsilon = 0.02$. $\Delta x = 10m$.[**ER**] ohad1/. impB-a,impB-b,impB-c,impB-d,impB-e,impB-f

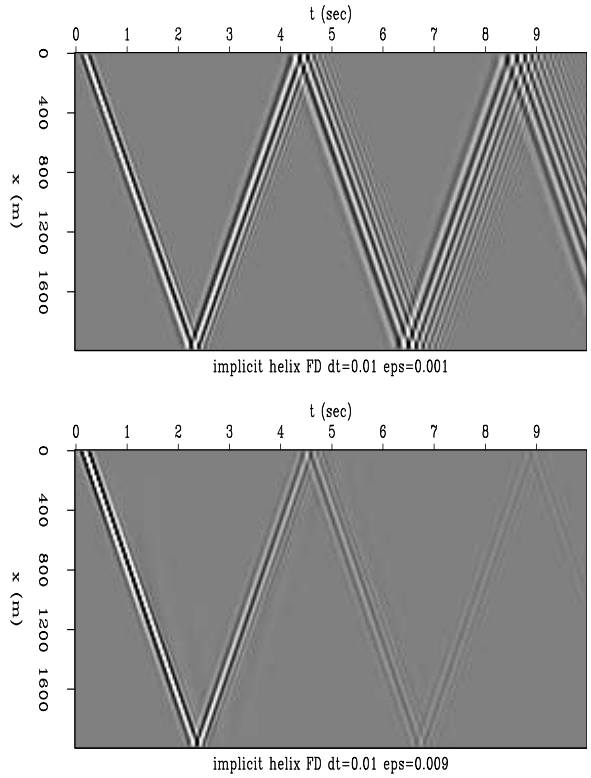


Figure 3: 1D Helical implicit finite-difference propagation with constant velocity = $1000m/s$. The time step size is $\Delta t = 10ms$. $\epsilon = 0.001$ for the top Figure, $\epsilon = 0.009$ for the bottom Figure.[**ER**] ohad1/. imp-vs-helimp-1d-bigeps

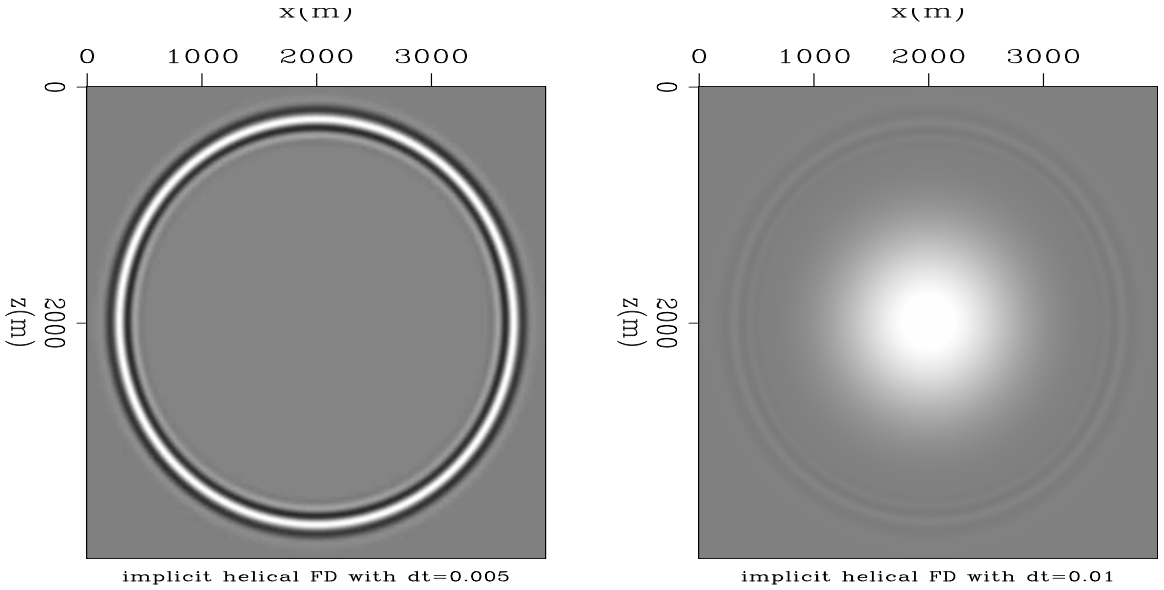


Figure 4: 2D helical implicit finite-difference with constant velocity = $1000m/s$. Wavefields are after 2 seconds of propagation. Source is a Ricker wavelet with central frequency = $12.5Hz$. The time step size is $\Delta t = 5ms$ for the left Figure, and $10ms$ for the right Figure. $\Delta x = \Delta z = 10m$.[**ER**] ohad1/. helimp-2d

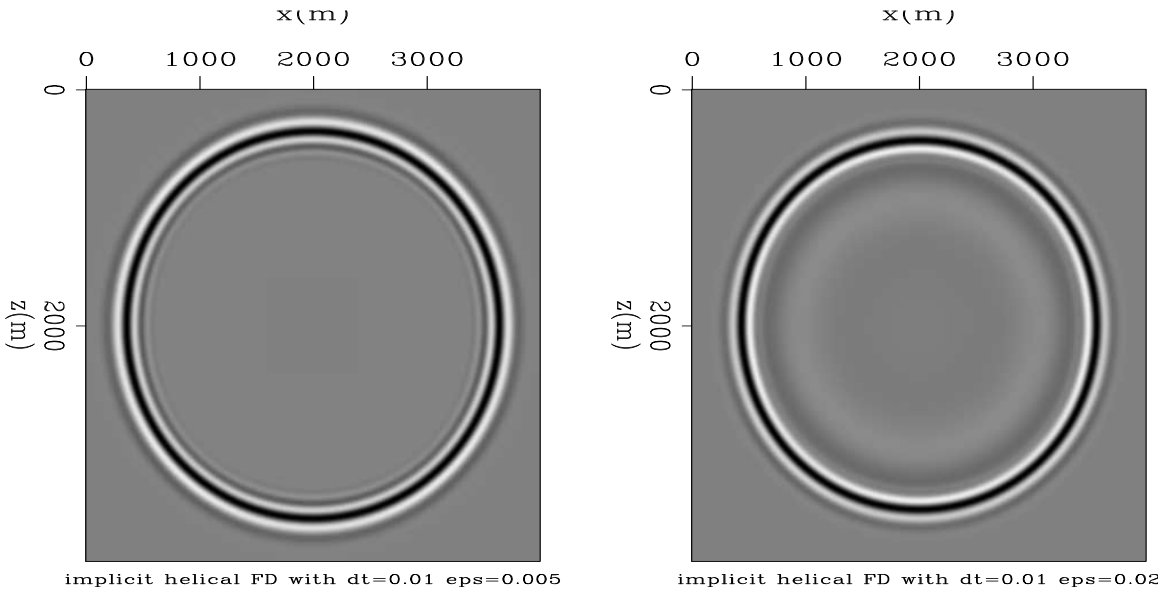


Figure 5: 2D helical implicit finite-difference with constant velocity $= 1000m/s$. Wavefields are after 2 seconds of propagation. Source is a Ricker wavelet with central frequency $= 12.5Hz$. The time step size is $\Delta t = 10ms$. $\epsilon = 0.005$ for the left Figure, and $\epsilon = 0.02$ for the right Figure. $\Delta x = \Delta z = 10m$.[**ER**] ohad1/. helimp-2d-eps

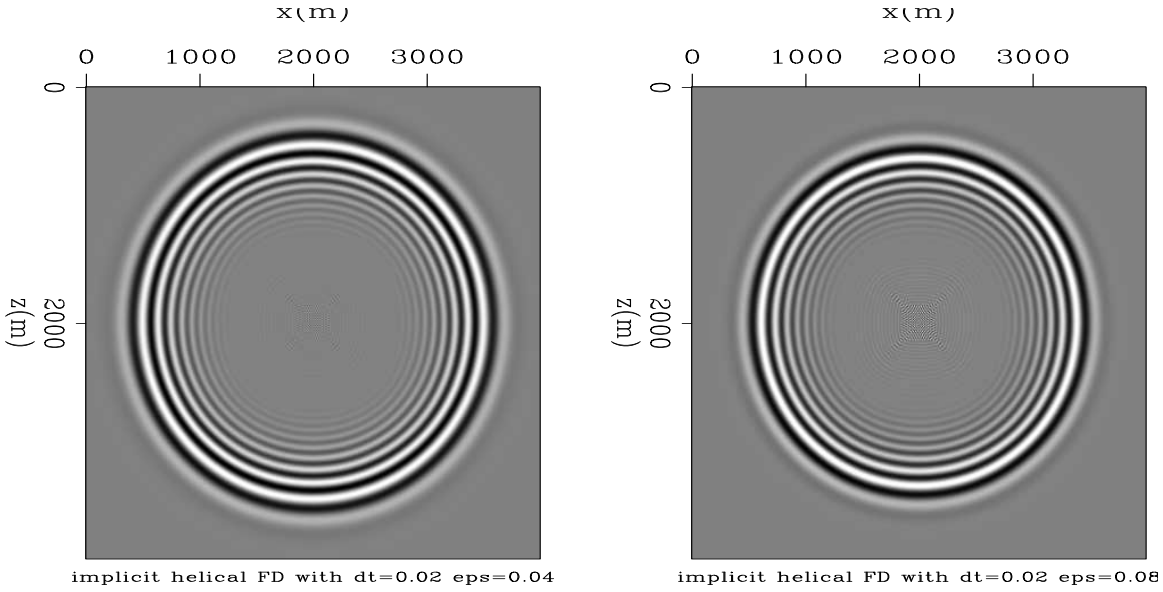


Figure 6: 2D helical implicit finite-difference with constant velocity $= 1000m/s$. Wavefields are after 2 seconds of propagation. Source is a Ricker wavelet with central frequency $= 12.5Hz$. The time step size is $\Delta t = 20ms$. $\epsilon = 0.04$ for the left Figure, and $\epsilon = 0.08$ for the right Figure. $\Delta x = \Delta z = 10m$.[**ER**] ohad1/. helimp-2d-eps2

from other tests with many different time step sizes I can say that the behaviour is always identical, and so is the response to varying value of ϵ. The similarity in the wavefield values extends to the statistics of the wavefields - the mean, average, RMS and min/max values are nearly identical as well. In summary - I could not find a set of parameters for which propagation with double precision variables is better (or at all different) than propagation with single precision.

The next step was to attempt to reduce the precision of the spectrally factorized coefficients one decimal point at a time, and see when propagation with a certain set of parameters is destroyed as a result of this loss of precision. This should give an indication as to how important the floating point precision actually is for stable propagation. The precision reduction was done by running the regular `wilson` spectral factorization subroutine, and then reducing precision by the following two lines of code:
noindent
```
IntFilter = CutFactor * FloatFilter
FloatFilter = IntFilter / CutFactor
```

`CutFactor` is a power of 10. Multiplying by this factor and then casting to integer effectiveley removes decimal precision from the filter coefficients. Example:
```
10000 * 1.23456 = 12345
12345 / 10000 = 1.2345
```

The purpose of this test was to see how many decimal precision digits can be removed from the filter coefficients before wavefield propagation using those coefficients is altered, in comparison to propagation with standard floating point precision. Results can be seen in Figures 8(a)-8(c). On the left is the result of propagation with single precision coefficients, with parameters which have shown stability ($\Delta t = 5ms, \epsilon = 0$). The center Figure shows propagation with coefficients which have had their precision truncated to 3 decimal points only. The wavefield exhibits a phase shift in comparison to the single precision wavefield, and yet it remains stable. Only when precision is truncated to 2 decimal points (right) is propagation severely affected.
The wavefields in Figures 7(a)-7(d) and 8(a)-8(c) were generated using factorization of the finite-difference weights of the 4th spatial order approximation (A-1). The values of these weights when using the specific set of propagation parameters were:
$U_0 = 1.3125, \qquad U_1 = -8.3333343E-02, \qquad U_2 = 5.2083340E-03.$

Since $\Delta x = \Delta z$, the weights are identical for both dimensions. These weights are fed to the spectral factorization routine, which is supposed to produce a causal set of filter coefficients, such that their cross-correlation will reproduce the finite-difference weights (Claerbout, 1997). This suggests that one way of testing the sensitivity of propagation to the floating point precision of the filter coefficients is to correlate the filter coefficients and compare the result to the finite-difference weights.

I used 21 filter coefficients to produce Figures 8(a)-8(c). For single precision propagation, the values of the correlation of the filter coefficients were (Only the first four values of the correlation are displayed. The rest are in A-2 to A-4):

$$1.312500 \quad -8.3333343E-02 \quad 5.2083335E-03 \quad 2.9154580E-12$$

For the propagation where precision was reduced to 3 decimal points only, the correlation was:

$$1.312500 \quad -8.3329208E-02 \quad 5.2077500E-03 \quad 0.0000000E+00$$

For the propagation where precision was reduced to 2 decimal points only, the correlation was:

$$1.312500 \quad -7.8187048E-02 \quad 0.0000000E+00 \quad 0.0000000E+00$$

Note that the correlation products are arranged in order of lags, so that the first coefficient corresponds to the central finite-difference weight U_0, the second to U_1, and the third to U_2. Note also that only after reducing precision to the 2nd decimal point, the weight U_2 is effectively erased, and the weight U_1 is considerably altered.

This comparison proves that the wavefield divergence shown in the previous sections is not the result of inadequate representation of the filter coefficient's floating point values when using single precision. If it were, then propagation with reduced precision would not have been possible. However, this result raises another question: If propagation is stable with so little precision, how come a small value of ϵ added to the central finite-difference weight (and by that also to the zero-lag filter coefficient) causes the wavefield to stabilize, when the effect that this slight addition has on the filter's correlation is so much less pronounced than the precision reduction?

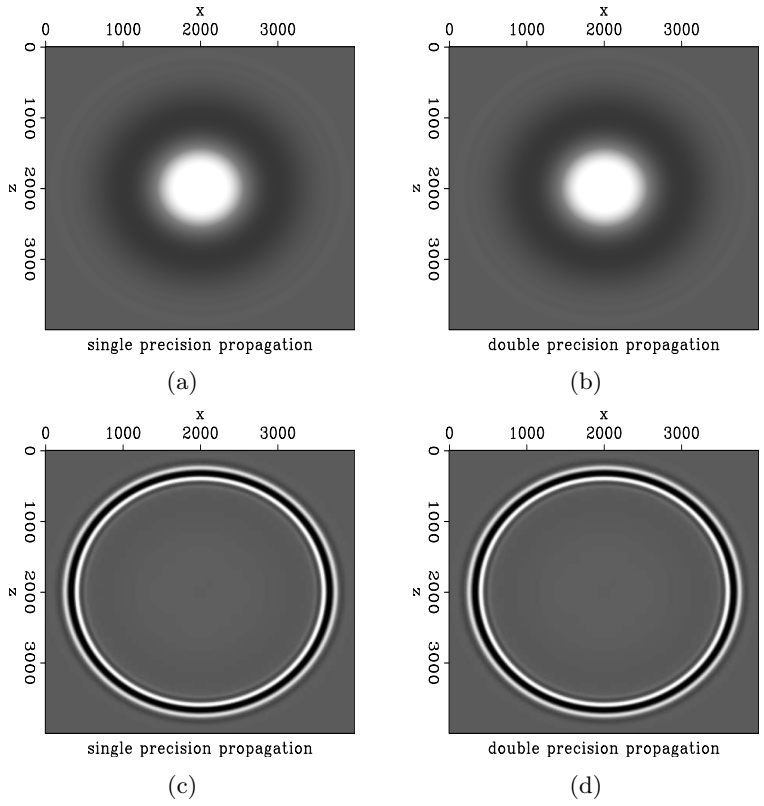


Figure 7: 2D helical implicit finite-difference using single (left) and double (right) precision. Velocity = $1000m/s$. Wavefields are after 2 seconds of propagation. Source is a Ricker wavelet with central frequency = $12.5Hz$. The time step size is $\Delta t = 10ms$. $\epsilon = 0$ for the top Figures, and $\epsilon = 0.01$ for the bottom Figures. $\Delta x = \Delta z = 10m$.[**ER**] ohad1/. double-vs-single1-a,double-vs-single1-b,double-vs-single1-c,double-vs-single1-d

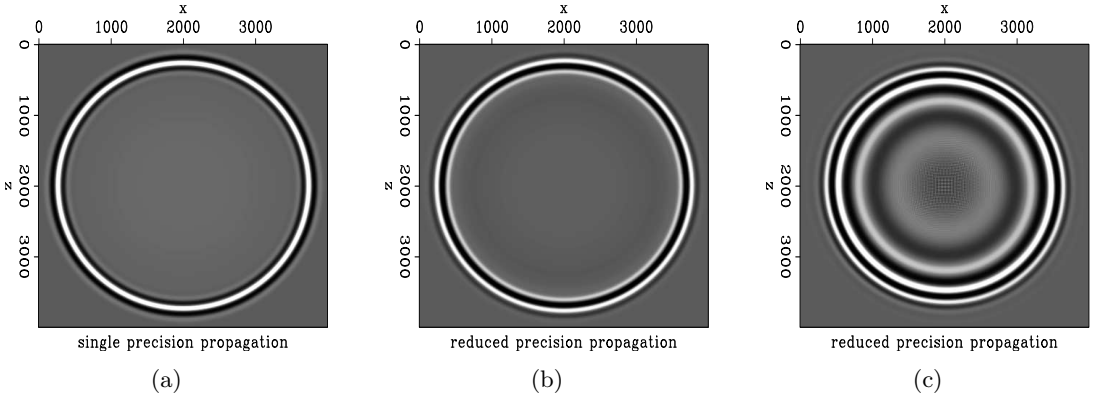


Figure 8: 2D helical implicit finite-difference using single precision (left), precision reduced to 3 decimal points (center), and precision reduced to 2 decimal points (right). Velocity = $1000m/s$, $\Delta t = 5ms$, $\epsilon = 0$, $\Delta x = \Delta z = 10m$.[**ER**] ohad1/. single-vs-half-a,single-vs-half-b,single-vs-half-c

EFFECT OF NUMBER OF SPECTRALLY FACTORIZED COEFFICIENTS

I had initially assumed that the number of filter coefficients would be the most dominant factor in determining the accuracy of the propagation. I had supposed that the more filter coefficients used in the spectral factorization, the closer would be the value of their correlation to the finite-difference weights. Indeed, the instinctive response I had to the divergence problem was to increase the number of coefficients in the spectral factorization parameters. This, unfortunately, had no effect. Furthermore, the correlation of the filter coefficients created by the spectral factorizer (the Wilson-Burg algorithm) produced accurate finite-difference weights even when very few filter coefficients were present.

An example of the lack of the effect of number of coefficients on the propagation is shown in Figure 9. This 1D example shows how propagation using 2 spectrally factorized filter coefficients is basically identical to propagation when using 50 filter coefficients. Another indication comes from observing the filter coefficients themselves. This example was produced by a 2nd order scheme, which means that there are only 2 finite-difference weights. When factorizing using only 2 filter coefficients, the Wilson-Burg algorithm (for the propagation parameters used in Figure 9) yielded:

$$1.000000 \quad -5.5728100E-02.$$

The coefficients are displayed in order of lags, so the 1.0 is the zero-lag filter coefficient. Correlating these coefficients, we get:

$$1.125 \quad -6.2500007E-02$$

at lag 0 and lag 1, which are equal to the floating point representations of the finite-difference weights for Figure 9.

Factorizing using 50 filter coefficients produced (only the first four coefficients are shown, the rest are in A-5 and A-6):

$$1.000000 \quad -5.5728100E-02 \quad -2.7755576E-17 \quad 1.7347235E-18.$$

After lag 20, the coefficients are all zeros. Note that the first two coefficients are identical to the ones produced by the factorizer when requesting only two coefficients.The correlation of this filter is:

$$1.125000 \quad -6.2500007E-02 \quad -3.1236769E-17 \quad 1.9455218E-18$$

This correlation again shows the accurate representation of the finite-difference coefficients at lag 0 and lag 1. In addition, the correlation produces a set of other values at later lags,

which are much smaller than the weights themselves.

The fact that two filter coefficients were sufficient to produce the finite-difference weights by correlation was interesting, but what is more important is to test what effect the change in the number of coefficients might have on the deconvolution process. Correlating the coefficients is like convolving them over a spike, once in the forward direction and once in reverse order. In order to test the exact effect that a change in the number of coefficients had on the deconvolution, I tested the result of deconvolving the coefficients over a spike. Here as well, the result was identical. I shall spare displaying the numbers themselves for this case.

In summary, I could not a find a combination of parameters (of propagation or of factorization) for which wavefield propagation was more stable if more filter coefficients were used.

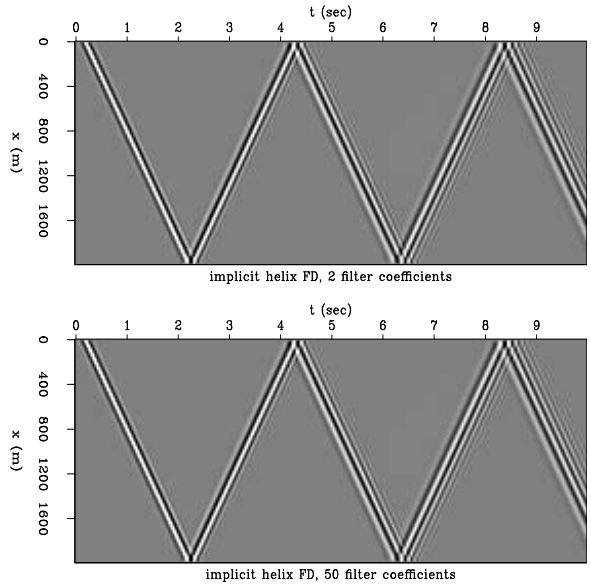


Figure 9: 1D helical implicit finite-difference with 2 spectrally factorized filter coefficients (top), and 50 coefficients (bottom). Velocity = $1000m/s$, $\Delta t = 5ms$, $\epsilon = 0$, $\Delta x = 10m$.[**ER**] ohad1/. helimp-ncoeffs

WILSON-BURG VS. KOLMOGOROFF SPECTRAL FACTORIZATION

Last on the checklist was the spectral factorization algorithm itself. In Rickett (2001) the Kolmogoroff spectral factorization method is shown to be successful for modeling seismic activity on the surface of the Sun. I used the SEPlib `ccrosskolmog` module, and compared wavefield propagation when the spectral factorization was done by the Kolmogoroff method vs. the Wilson-Burg method. The comparison is shown in Figure 10. On the left are wavefields propagated with Wilson factorization, and on the right - Kolmogoroff. The time step is $\Delta t = .5ms$ in the top Figures. When the time step is increased to $1ms$, the propagation with Kolmogoroff coefficients diverges. However, if $\epsilon = 10^{-4}$ is added to the central finite-difference coefficient prior to factorization (bottom right), propagation is successful and appears similar to propagation by Wilson factorized coefficients.

This result indicates that the Kolmogoroff factorization method is even less suitable than the Wilson method for this finite-difference scheme, since the addition of a small value to the central FD coefficient when using Wilson is necessary only at greater time step sizes.

The ten Wilson filter coefficients used to create the center panels in Figure 10 were:

$$\begin{array}{cccc} 1.0 & -2.4875777E-03 & 0.0000000E+00 & -6.7762636E-21 \\ -2.6469780E-23 & 5.1698788E-26 & 0.0000000E+00 & -3.9443045E-31 \\ 0.0000000E+00 & -1.5046328E-36 & & . \end{array}$$

The Kolmogoroff coefficients were:

$$\begin{array}{cccc} 1.002494 & -2.4938183E-03 & 4.7695384E-08 & 1.3291222E-08 \\ -3.6223135E-08 & 4.7327675E-09 & 8.5023713E-09 & 8.8970848E-09 \\ -9.8089106E-12 & 4.9380566E-09 & & . \end{array}$$

Other than the zero-lag coefficient not being equal to 1, a striking difference is that the Kolmogoroff coefficients do not drop off quickly as do the Wilson coefficients. This has a degrading effect on the filter correlation. The Wilson filter's correlation is:

$$\begin{array}{cccc} 1.005 & -2.5000004E-03 & 1.6940662E-23 & -6.8100374E-21 \\ -2.6602097E-23 & 5.1956968E-26 & 9.8607629E-34 & -3.9640020E-31 \\ 0.0000000E+00 & -1.5121468E-36 & & \end{array}$$

The Kolmogoroff's filter correlation is:

$$\begin{array}{cccc} 1.005 & -2.4999434E-03 & 8.8091141E-08 & 7.9071558E-09 \\ -2.9466412E-08 & -1.4710333E-10 & 3.0997090E-11 & 3.9199342E-09 \\ -6.6459863E-12 & -2.8234270E-09 & & \end{array}$$

The finite-difference coefficients for the parameter set of the wavefields in Figure 10 are $U_0 = 1.005, U_1 = -2.5E-03$. The Wilson filter's correlation recreates these weights precisely, while the Kolmogoroff filter's correlation does not. Also, the drop-off in the magnitude of the filter correlation at lags which do not represent finite-difference weights (i.e. not lag 0 or lag 1) is much better for the Wilson filter.

CONCLUSION AND FUTURE WORK

After conducting the aforementioned tests, I still cannot say why wavefield propagation by the proposed methodology does not function beyond a certain time step size. I can only conclude that for some reason the spectral factorization fails when the finite-difference weights, which I wish to factorize, are not dominated by the central finite-difference weight. Since the entire purpose of attempting to use the combination of implicit finite-difference and spectral factorization for propagation was to increase the time step size (thereby decreasing the total computation time, but also decreasing dominance of the central finite-difference weight), this failure makes the method unuseful. At the time step sizes for which this method does work, explicit methods will function better and faster.

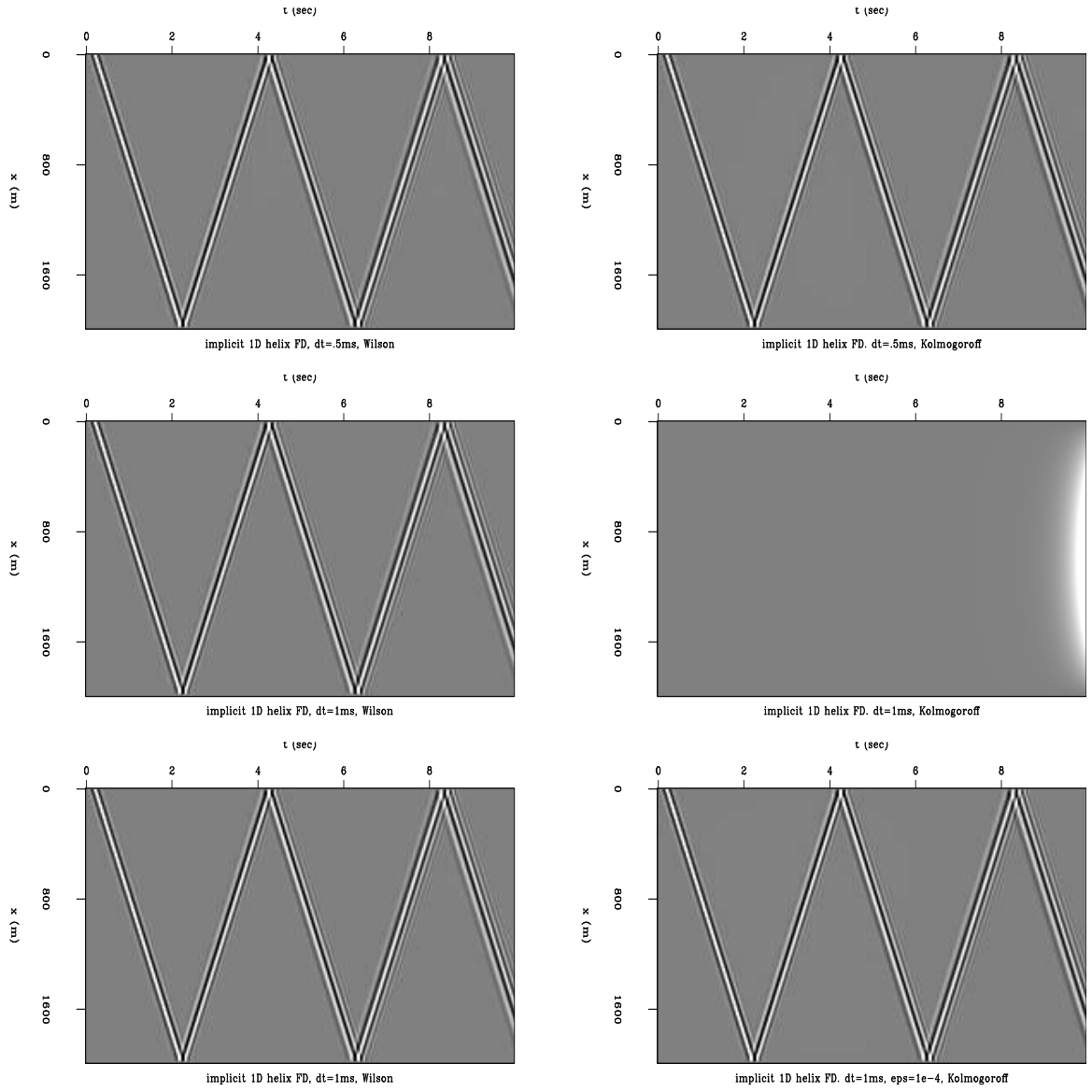


Figure 10: 1D helical implicit finite-difference with Wilson-Burg spectral factorization (left), and Kolmogoroff spectral factorization (right). $\Delta t = .5ms$ (top), $1ms$ (center and bottom). $\epsilon = 1e^{-4}$ only on the bottom right Figure, otherwise $\epsilon = 0$. Velocity $= 1000m/s$, $\Delta x = 10m$.[**ER**] ohad1/. wil-vs-kol

There is one possible avenue in which to continue research of this method. The central weight of the finite-difference scheme which I used does decrease in dominance as the time step size is increased, but I am not bound to use this scheme only. It is possible that an alternate implicit finite-difference scheme will not have this attribute, and will thus be more amenable to factorization when the time step size is increased.

One source of such a scheme could be the pseudo-Laplacian discussed in Etgen and Bransdsberg-Dahl (2009).

REFERENCES

Barak, O., 2010, Implicit finite difference in time-space domain with the helix transform: Stanford Exploration Project, **Report 140**, 103–118.

Claerbout, J. F., 1997, Multidimensional recursive filters via a helix: Stanford Exploration Project, **Report 95**, 1–13.

Etgen, J. T. and S. Bransdsberg-Dahl, 2009, The pseudo-analytical method: application of pseudo-laplacians to acoustic and acoustic anisotropic wave propagation: SEG Expanded Abstracts, 2552–2556.

Rickett, J., 2001, Spectral factorization of wavefields and wave operators: PhD thesis, Stanford University.

APPENDIX A

The 4th order in space and 2nd order in time implicit finite difference scheme used to create Figures 7(a) and 8(a) was:

$$\begin{aligned}\frac{P_x^{t+1}-2P_x^t+P_x^{t-1}}{\Delta t^2} &= \frac{C^2}{4\Delta x^2}[\left(\frac{-1}{12}(P_{x+2}^{t+1}+P_{x-2}^{t+1})-\frac{16}{12}(P_{x+1}^{t+1}+P_{x-1}^{t+1})-\frac{30}{12}P_x^{t+1}\right)\\ &+ 2\left(\frac{-1}{12}(P_{x+2}^{t}+P_{x-2}^{t})-\frac{16}{12}(P_{x+1}^{t}+P_{x-1}^{t})-\frac{30}{12}P_x^{t}\right)\\ &+ \left(\frac{-1}{12}(P_{x+2}^{t-1}+P_{x-2}^{t-1})-\frac{16}{12}(P_{x+1}^{t-1}+P_{x-1}^{t-1})-\frac{30}{12}P_x^{t-1}\right)]. \qquad \text{(A-1)}\end{aligned}$$

The correlation of the 21 filter coefficients used to create Figure 8(a) for single precision propagation was (I apologize for having the temerity to show raw numbers, but I couldn't find a suitable graphic representation):

$$\begin{array}{cccc} 1.312500 & -8.3333343E-02 & 5.2083335E-03 & 2.9154580E-12 \\ 5.4817577E-09 & -4.2199644E-09 & -2.5573333E-12 & 2.8602476E-13 \\ 3.4990573E-11 & -4.1297177E-10 & -8.3333343E-02 & -2.3679786E-12 \\ 2.3124791E-15 & -1.7574841E-13 & -3.4801828E-10 & 3.0937783E-10 \\ 5.6366680E-15 & -1.0001472E-13 & 0.0000000E+00 & -3.7887658E-11 \\ 5.2083340E-03 & & & . \end{array} \qquad \text{(A-2)}$$

For propagation where precision was reduced to 3 decimal points only, the correlation was:

$$\begin{array}{cccc} 1.312500 & -8.3329208E-02 & 5.2077500E-03 & 0.0000000E+00 \\ 0.0000000E+00 & 0.0000000E+00 & 0.0000000E+00 & -2.0831001E-05 \\ 5.2077517E-06 & 4.1662315E-05 & -8.2350150E-02 & -2.0831001E-05 \\ 0.0000000E+00 & 0.0000000E+00 & 0.0000000E+00 & 0.0000000E+00 \\ 0.0000000E+00 & 0.0000000E+00 & 2.0831001E-05 & -3.3329602E-04 \\ 5.2077500E-03 & & & . \end{array} \tag{A-3}$$

For propagation where precision was reduced to 2 decimal points only, the correlation was:

$$\begin{array}{cccc} 1.312500 & -7.8187048E-02 & 0.0000000E+00 & 0.0000000E+00 \\ 0.0000000E+00 & 0.0000000E+00 & 0.0000000E+00 & 0.0000000E+00 \\ 0.0000000E+00 & 4.6912231E-03 & -7.8187048E-02 & 0.0000000E+00 \\ 0.0000000E+00 & 0.0000000E+00 & 0.0000000E+00 & 0.0000000E+00 \\ 0.0000000E+00 & 0.0000000E+00 & 0.0000000E+00 & 0.0000000E+00 \\ 0.0000000E+00 & & & . \end{array} \tag{A-4}$$

The correlation products are arranged in order of lags. Since the finite-difference operator is two dimensional, the weights for U_1 and U_2 reappear at lags corresponding to the wrap-around of the 1D filter around the edges of the 2D grid (in helical coordinates). Therefore the 11th coefficient is equal to the 2nd coefficient, and the 21st is equal to the 3rd.

The 50 filter coefficients used to produce Figure 9:

$$\begin{array}{cccc} 1.000000 & -5.5728100E-02 & -2.7755576E-17 & 1.7347235E-18 \\ 0.0000000E+00 & -6.7497938E-21 & 8.2718061E-25 & 7.7548182E-26 \\ 8.2718061E-25 & 2.6315262E-28 & 1.4603365E-29 & 2.0273725E-28 \\ 6.7288147E-32 & 3.7618776E-33 & -2.4442825E-32 & 1.1651518E-35 \\ 6.4931696E-37 & 3.5264831E-38 & 0.0000000E+00 & -2.3509887E-38 \\ 0.0000000E+00 & .. & .. & .. \end{array} \tag{A-5}$$

After lag 20, the coefficients were all zeros. The correlation of these coefficients was:

$$\begin{array}{cccc} 1.125000 & -6.2500007E-02 & -3.1236769E-17 & 1.9455218E-18 \\ 4.2186216E-22 & -7.5700597E-21 & 9.2285031E-25 & 3.5272806E-26 \\ 9.2768060E-25 & 2.9421741E-28 & 3.7068419E-30 & 2.2736905E-28 \\ 7.5229681E-32 & 5.7466863E-33 & -2.7413771E-32 & 1.3026792E-35 \\ 7.2821997E-37 & 3.9550105E-38 & 0.0000000E+00 & -2.6366737E-38 \\ 0.0000000E+00 & .. & .. & .. \end{array} \tag{A-6}$$

Short note: GPU accelerated 3D wave propagation and continuous coil shooting

Chris Leader, Robert Clapp, and Biondo Biondi

ABSTRACT

This short note discusses how continuous coil shooting for a synthetic VSP survey could lead towards more azimuth rich data whilst keeping the survey time below that of a conventional towed streamer survey.

INTRODUCTION

Throughout the history of reflection seismology, particularly since the large scale inception of 3D surveys, there has been a simultaneous desire to both improve acquisition geometries and also to reduce survey cost. At first glance these goals appear to be contradictory, and for the most case they are; however, this short note postulates that recent innovations in simultaneous source acquisition and in coil shooting could lead toward these stipulations.

Areas that exhibit complex salt geology, such as the North sea, the western Gulf of Mexico and offshore West Africa, are extremely petroleum-rich making the production of detailed, high resolution 3D images of these areas key in understanding and appraising such fields. Over the last 10 years there has been a proliferation of survey techniques that differ from or augment the usual single source streamer towed cable geometry. Methods such as multiple azimuth surveys (MAZ) (Manning, 2007) and wide azimuth surveys (WATS) (Verwest and Lin, 2007) are common. When compared to equivalent, traditional narrow azimuth surveys these images exhibit better illumination, more consistent amplitudes along reflectors and sub-salt coherency. However, whilst these methods improve image illumination they also greatly increase the cost the survey.

More recently, coil shooting has gained interest as a technique of acquiring more azimuth rich data without increasing survey costs as much as MAZ and WATS. It has been shown with synthetic data that over complex geologies coil shooting can provide more illumination and fill in gaps in 3D angle gathers (Buia (2009); Moldoveanu and Kapoor (2009)). Also recently the concept of simultaneous shooting (Aaron and Fromyr (2009); Ayeni and Biondi (2009); Tang and Biondi (2009)) within surveys has gained momentum, since this technqiue reduces acquisition time, which is generally considered to be 80% of a survey's cost. The problem with the latter technique is that when cable towed streamers acquire simultaneously shot data, the waves recorded tend to be conical, rather than spherical, due to the motion of the source vessel. This gives reduced angular illumination compared to an equivalent, conventionally shot survey. We suggest that combining continuous shooting with coil shooting will help to both reduce acquisition costs and fill in illumination holes

seen in 3D angle gathers.

PROPOSED METHOD

In order to simulate this problem synthetic data was produced using a 3D subsection of the SEAM velocity model. Initially a 2D VSP dataset was simulated using a parallelised, variable-density, two-way modelling CPU code. A VSP survey was chosen since reciprocity can be used to simulate the relevant surface geometries and this would require far fewer shot simulations. A total of 32 shots were simulated. To extend this to 3D modelling a constant density 3D GPU propagation kernel was written, and is in the process of being adapted to simulate a dataset of this size at an acceptable speed

SEAM DATASET AND 2D DATA

The model used to create the data was the latest iteration of the SEAM model, and for the 2D case a variable density two-way wave propagation algorithm was used. For the 3D modelling a constant density 3D GPU code was used, with the view of extending this to variable density. The SEAM model itself is extremely large, with over 20 billion samples, and so a small section of the data was windowed. A representative view of this windowed section can be seen in Figure 1.

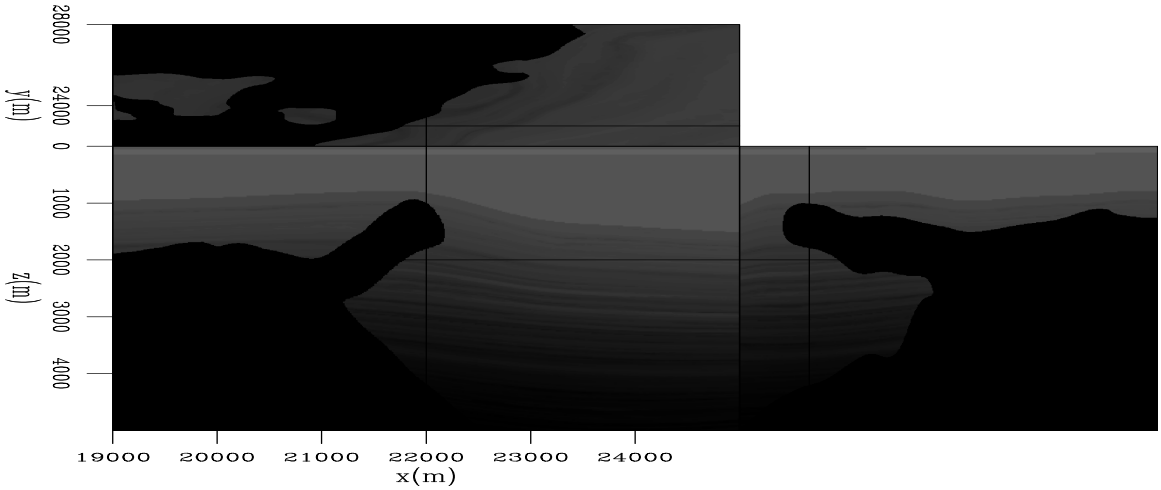


Figure 1: A cubeplot of the windowed section of the SEAM data. [**ER**] chris2/. 3dslice

Within this section, a salt body with an allochthonous arm extends into the sediments. Above the salt there is a carbonate layer, and the sediments are finely layered with an increasing velocity with depth.

Figure 2 shows an example of a simulated shot from the SEAM dataset.

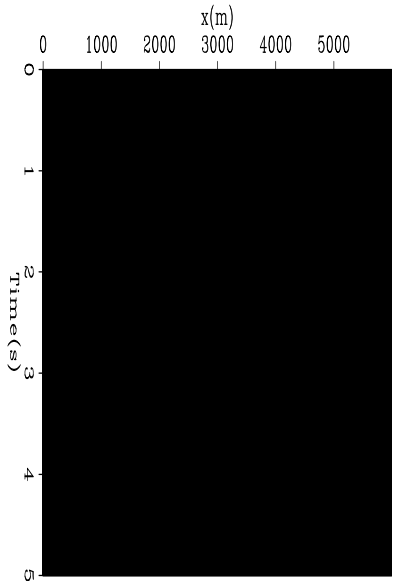


Figure 2: A subsurface shot at 1km depth. [**ER**] chris2/. 2dshot

3D GPU MODELLING AND FUTURE WORK

The next step in this concept is to extend the modelling to 3D. Thirty-two VSP shots are in the process of being modelled, with the entire surface wavefield saved every several time steps (such that the survey time sampling is 4ms.) From these densely sampled wavefields both continuous linear towed streamer and continuous coil shot streamer surveys will be modelled. These data can then be migrated (reverse time) and their images and 3D angle gathers analysed to see if better offset sampling per azimuth is observed, and if the survey type changes are noticeable in the final images.

REFERENCES

Aaron, P, v. B. R. and E. Fromyr, 2009, Simultaneous sources: A controlled experiment on different source configurations: SEG Expanded Abstracts, **28**, 1177–1181.

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

Buia, M. e. a., 2009, Full azimuth circular survey in indonesia: Survey design, onboard illumination qc and preliminary processing results: SEG Expanded Abstracts, **28**, 71–75.

Manning, T. e. a., 2007, Quantifying and increasing the value of multi-azimuth seismic: The Leading Edge, **26**, 510–520.

Moldoveanu, N. and J. Kapoor, 2009, What is the next step after waz for exploration in the gulf of mexico?: SEG Expanded Abstracts, **28**, 41–45.

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

Verwest, B. and D. Lin, 2007, Modelling the impact of wide-azimuth acquisition on subsalt imaging: Geophysics, **72**, 241–250.

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	723-3187	qiang
Guitton, Antoine	–	antoine
Halpert, Adam	723-6006	adam
Lau, Diane	723-1703	diane
Leader, Chris	723-0463	chris1
Li, Elita	723-9282	myfusan
Shen, Xukai	723-0463	xukai
Shen, Yi	723-3187	yishen
Tang, Yaxun	723-1250	tang
Taweesintananon, Kittinat	723-3187	kittinat
Wong, Mandy	723-9282	mandyman
Zhang, Yang	723-1250	yang

SEP fax number: (650) 723-0683

E-MAIL

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

WORLD-WIDE WEB SERVER INFORMATION

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

STEERING COMMITTEE MEMBERS, 2010-2011

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
Luca Cazzola (Co-chair, 1st year)	Eni E&P	–	luca.cazzola@eni.com
Jon Claerbout	SEP	(650)723-3717	jon@sep.stanford.edu
Richard Cook	Shell	(713)245-7195	richard.cook@shell.com
Stewart A. Levin	Halliburton	(303)488-3062	stewart.levin@halliburton.com
Yi Luo	Saudi Aramco	–	yi.luo@aramco.com
Marta Woodward (Co-chair, 2nd year)	WesternGeco	(832)798-6204	martajo@slb.com

Research Personnel

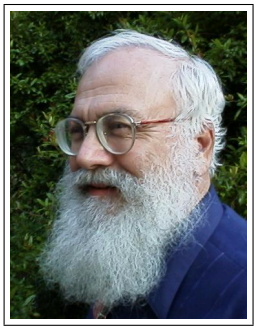
Ali Almomin graduated from Texas A&M University in 2007 with a BS in Geophysics. Then, he joined Saudi Aramco and worked in several exploration and research departments with a focus on 3D seismic processing and near surface imaging. He joined Stanford Exploration Project in 2009 to pursue a PhD in Geophysics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

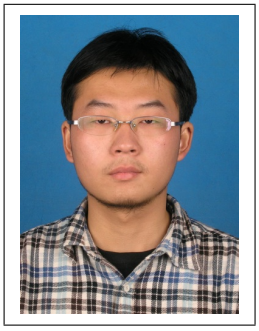
Antoine Guitton received a M.Sc. in geophysics from Universite de Strasbourg, France in 1996 and from Stanford University in 2000. He received his Ph.D. from Stanford University in 2005. He was awarded Best Student Paper from the SEG in 1999 and received the EAGE Van Weelden award in 2004. He was research assistant at the Institut Francais du Petrole (Paris-1996/97) working on well seismic imaging and research geophysicist at CGG Houston (1997-98) working on multiples attenuation. He is now Senior Geophysicist for 3DGeo Inc. in Santa Clara, CA. His research interests include interpolation, noise attenuation, inversion and novel seismic-interpretation techniques. He is a member of the SEG and EAGE.

Chris Leader graduated from Oxford Univeristy in 2008 with a BA in Physics (with concentration on Astrophysics and Condensed Matter physics) and then from Imperial College London in 2009 with an MSc in Petroleum Geophysics (Distinction). He is currently a first year student in the Stanford Exploration Project on the PhD program working on Fourier methods of regularisation. Work experience involves 3D seismic processing for a Rio Tinto acquired dataset over summer 2009. He is a member of SEG, EAGE, PESGB and IOP.

Dave Nichols received a B.A. in physics from Cambridge University in 1982; a M.Sc. in geophysics from Imperial College, London in 1983; and a Ph.D. in geophysics from Stanford University in 1994. From 1983 to 1987, he was employed by Western Geophysical in London and joined SEP in 1987. During the summer of 1991, he worked for Chevron Oil Field Research Company in La Habra, California. He currently works for Schlumberger in the Schlumberger Reservoir Completions Center in Rosharon, Texas.

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.

Yang Zhang graduated from Tsinghua University in July 2007 with a B.E. in Electrical Engineering. He took an internship in Microsoft Research Asia during 2007-2008. He joined SEP in 2009, and is currently pursuing a Ph.D. in Geophysics.



SEP ARTICLES PUBLISHED OR IN PRESS

Ayeni, G., A. Huck, and P. de Groot, 2008, Extending reservoir property prediction with pseudo-wells: First Break, **26**, 11–15.

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** (Accepted).

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.

Biondi, B., 2010, Velocity estimation by image focusing analysis: Geophysics, in press.

Biondi, B., 2010, Velocity estimation by image focusing analysis: SEG Technical Program Expanded Abstracts **29** (Accepted).

de Ridder, S., Prieto, G.A., 2008, Seismic interferometry and the spatial auto-correlation method on the Regional Coda of the non-proliferation experiment, Eos Trans. AGU, 89(53), Fall Meet. Suppl., Abstract S31A-1885.

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** (Accepted).

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.

Halpert, A., Clapp, R.G., Lomask, J., and B. Biondi, 2008, Image segmentation for velocity model construction and updating: SEG Technical Program Expanded Abstracts **27**, 3088–3092.

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** (Accepted).

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

Maysami, M., and F. J. Herrmann, 2008, Lithology constraints from seismic waveforms: application to opal-A to opal-CT transition: SEG Technical Program Expanded Abstracts **27**, 2011-2015.

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

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

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

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.

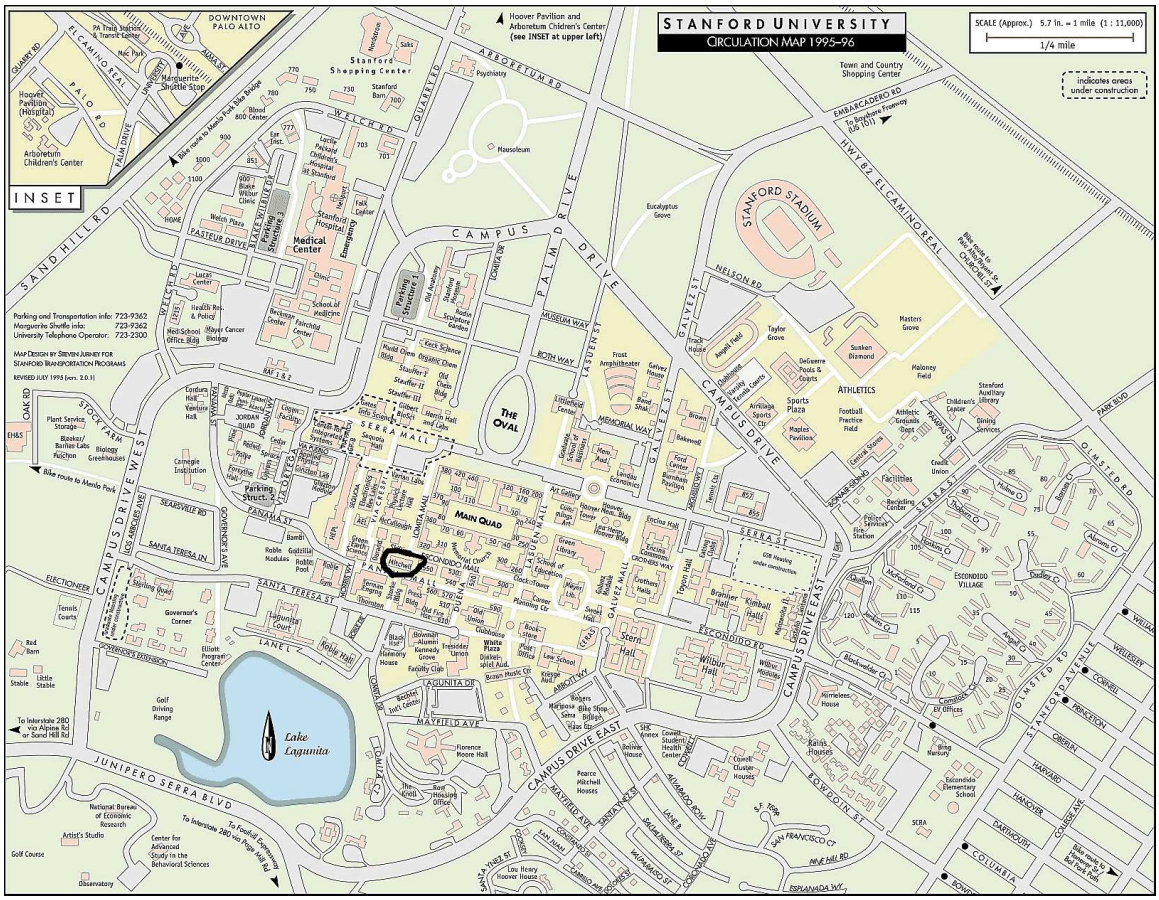
Tang, Y. and S. Lee, 2010, Preconditioning full waveform inversion with phase-encoded Hessian: SEG Technical Program Expanded Abstracts **29** (Accepted).

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

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** (Accepted).

Wong, M., Biondi L. B., and Ronen, S., 2010, Joint inversion of up- and down-going signal for ocean bottom data: SEG Technical Program Expanded Abstractsi **29** (Accepted).



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 Chidren'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
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
Medical Center
Stanford Hospital
Mausoleum
Eucalyptus Grove
STANFORD STADIUM
ATHLETICS
Sports Plaza
Maples Pavilion
THE OVAL
MAIN QUAD
Memorial Church
Green Library
Hoover Tower
Mitchell
Tresidder Union
White Plaza
Stern Hall
Wilbur Hall
ESCONDIDO VILLAGE
Lake Lagunita
CAMPUS DRIVE
CAMPUS DRIVE WEST
CAMPUS DRIVE EAST
PALM DRIVE
SERRA MALL
JUNIPERO SERRA BLVD
SAND HILL RD
HWY 82 EL CAMINO REAL
EMBARCADERO RD
STANFORD AVENUE
To Interstate 280 via Alpine Rd or Sand Hill Rd
To Interstate 280 via Page Mill Rd
Bike route to Menlo Park