Data examples of logarithm Fourier-domain bidirectional deconvolution

Qiang Fu, Yi Shen and Jon Claerbout

ABSTRACT

Time-domain bidirectional deconvolution methods show great promise for overcoming the minimum-phase assumption in blind deconvolution of signals containing a mixed-phase wavelet, such as seismic data. However, usually one time-domain method is slow to converge (the slalom method) and the other one is sensitive to the initial point or preconditioner (the symmetric method). Claerbout proposed a logarithm Fourier-domain method to perform bidirectional deconvolution. In this paper, we test the new logarithm Fourier-domain method on both synthetic data and field data. The results demonstrate that the new method is more stable than previous methods and produces better results.

INTRODUCTION

Usually, a seismic data trace d can be decomposed into a convolution of a wavelet w with a reflectivity series r, as $d = r * w$. Traditionally, seismic blind deconvolution has two assumptions, whiteness and minimum phase. The whiteness assumption supposes that the reflectivity series r is a white spectrum. The minimum-phase assumption supposes that the wavelet w in our problem has minimum-phase. Recently, some new methods have been proposed to avoid or correct these two assumptions in seismic blind deconvolution.

In Zhang and Claerbout (2010a), the authors proposed to use a hyperbolic penalty function introduced in Claerbout (2009) instead of the conventional L2 norm penalty function to solve blind deconvolution problem. With this method, a sparseness assumption takes the place of the traditional whiteness assumption in the deconvolution problem. Subsequently, Zhang and Claerbout (2010b) proposed a new method called "bidirectional deconvolution" in order to overcome the minimum-phase assumption. If the wavelet w is a mixed-phase wavelet, it can be decomposed into a convolution of two parts: $w = w_a * w_b$, where w_a is a minimum-phase wavelet and w_b is a maximum-phase wavelet. We use two deconvolution filters a and b to deal with the two wavelets w_a and w_b. Since Zhang and Claerbout (2010b) solve the two deconvolution filters a and b alternately, we call this method the slalom method. Shen et al. (2011) proposed another method to solve the same problem. They use a linearized approximation to solve the two deconvolution filters simultaneously. We call this method the symmetric method. Fu et al. (2011) proposed a way to choose an initial solution to relieve the

local-minima problem caused by the highly non-linear blind deconvolution problem. **?** discussed a very important aspect of any inversion problem, preconditioning and how it improves bidirectional deconvolution.

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

METHODOLOGY

Claerbout et al. (2011) show the complete derivation of the method, and we do not want to repeat it here. We describe only the major steps of this method. As with any iterative method, we have two issues to solve in one iteration: the update direction and the step length of the update. Now let us see how we can solve these two issues in the logarithm Fourier-domain method.

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

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

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

$$\begin{cases} w_a * a = \delta(n) \\ w_b * b = \delta(n) \end{cases}. \tag{2}$$

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

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

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

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

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

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

Our problem becomes

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

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

$$\begin{cases} \Delta u = r^{\circledast}\mathrm{Hyp}'(r) \\ \Delta r = r * \Delta u \end{cases}, \tag{7}$$

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

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

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

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

$\alpha_j = 0$
Iterate(j)
$\quad \alpha_j = \dfrac{\sum_i \mathrm{Hyp}'(r_i)\Delta r_i}{\sum_i \mathrm{Hyp}''(r_i)\Delta r_i^2}$
$\quad \alpha_{final} = \alpha_{final} + \alpha_j$
$\quad r = r + \alpha_j \Delta r$
$\quad u = u + \alpha_j \Delta u$

With the update directions (both for the unknown u and for the residual r) and the step length α of the update, we have everything we need for an iteration. We can keep updating the unknown until we are satisfied with the convergence.

Trial and error on step length α

When we use the method above to calculate the step length α for some data set, we may get a blow-up problem, caused by over-shooting the solution. The Newton method requires a convex function, but for some field data sets that condition may not be met. There is a solution for avoiding this problem. If we use the method described above to get step length α, we usually overshoot. Therefore, we use trial and error to avoid step length α being too large. If the hyperbolic penalty function on $r = r + \alpha\Delta r$ is greater than on r, the step length α is too large, and we overshoot;

in that case, we just reduce the step length α by half.

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

EXAMPLES

In this section we will test the logarithm Fourier-domain method with three test cases. One is a simple synthetic data set, and other two are field data sets. As prior experience (**?**) shows, the preconditioning is a critical part of the seismic blind deconvolution problem. Here we use Burg PEF as the preconditioner for all tests. We will use one filter for all traces unless otherwise specified.

Synthetic 1D example

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

From this very simple 1D synthetic example, we use 0.1 as the threshold of the hyperbolic penalty function for the logarithm Fourier-domain method and use 95% of the data residual as the threshold for the time-domain method. Using the logarithm Fourier-domain method, we turn the Ricker wavelet into a spike output after about 50 iterations. Using the time-domain symmetric method, even after 30,000 iterations, we get a major spike followed by a minor spike, plus a few additional jitters at the begining of the trace.

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

Field data common-offset gather example

Now we test the logarithm Fourier-domain bidirectional deconvolution with a 2D marine common offset gather data set that is commonly used to test the time-domain bidirectional deconvolution methods. This 2D marine common-offset gather is very popular in papers discussing blind deconvolution using the hyperbolic penalty function. In Zhang and Claerbout (2010a), Zhang and Claerbout (2010b), Fu et al. (2011), Shen et al. (2011) and **?** the authors tested their methods and theories with this data set as a field data example. Hence this is a good choice for comparing this new method with previous ones.

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

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

From figures 4 and 5, we can see that both deconvolution results are good. However we think the logarithm Fourier-domain method (figure 4) is a little better. Within the salt body (in the vicinity of 2.4 s to 2.6 s), the Fourier-domain result looks cleaner. We do not expect to see any features within the salt body, and all events we see in this area in the raw data are the air-gun bubbles. This indicates that the deconvolution in the Fourier domain handles the air-gun bubbles a little better than the time-domain method. However, excluding these differences, the rest of the results are quite similar. We can also find that the estimated wavelets (figures 6(a) and 6(b)) are very similar, except for the tail parts. This is consistent with the deconvolution result.

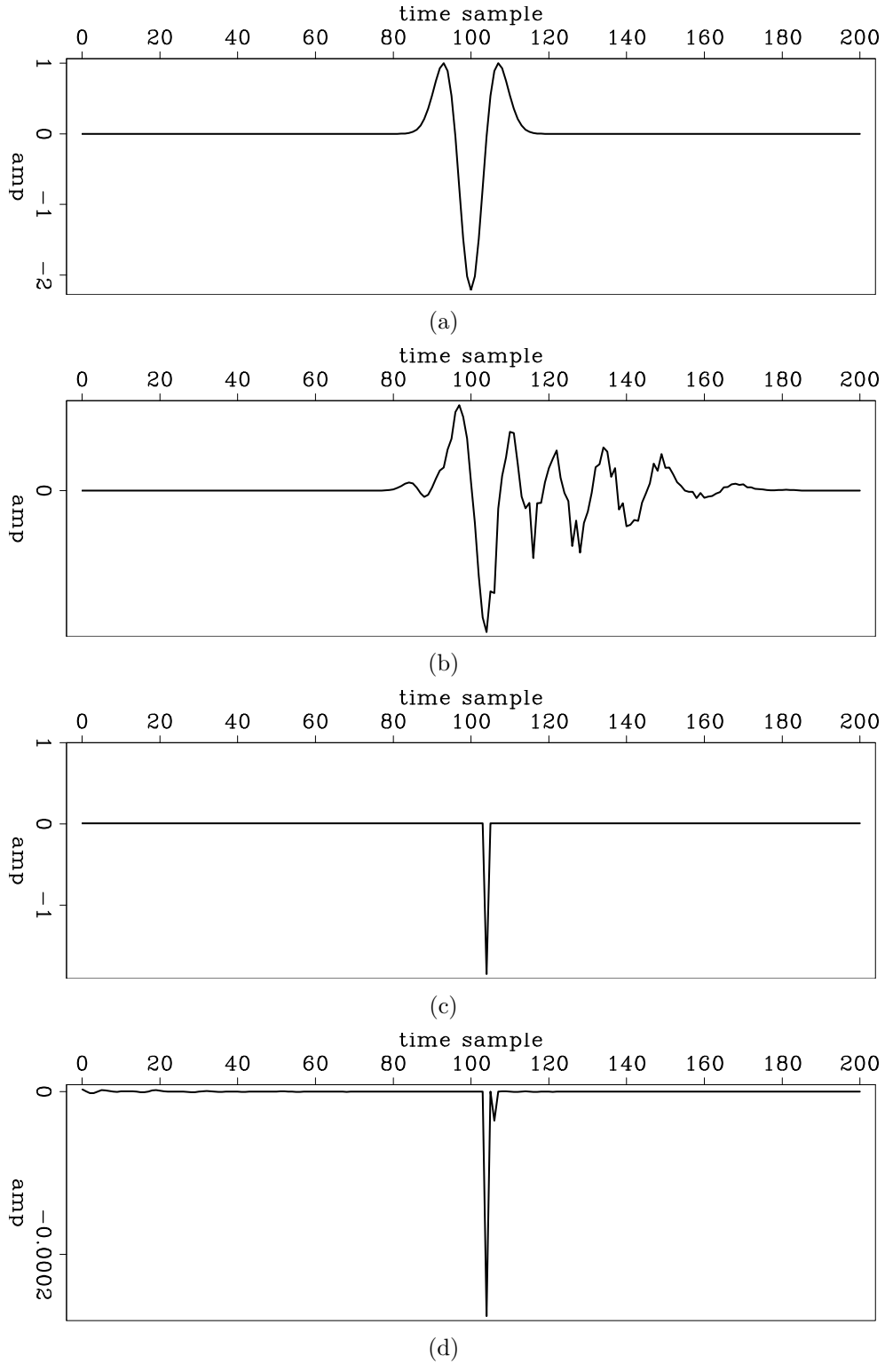

Figure 1: (a) The synthetic Ricker wavelet; (b) The Ricker wavelet after Burg PEF preconditioning; (c) The bidirectional deconvolution result of the logarithm Fourier-domain method after 50 iterations; (d) The bidirectional deconvolution result of the time-domain symmetric method after 30,000 iterations.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

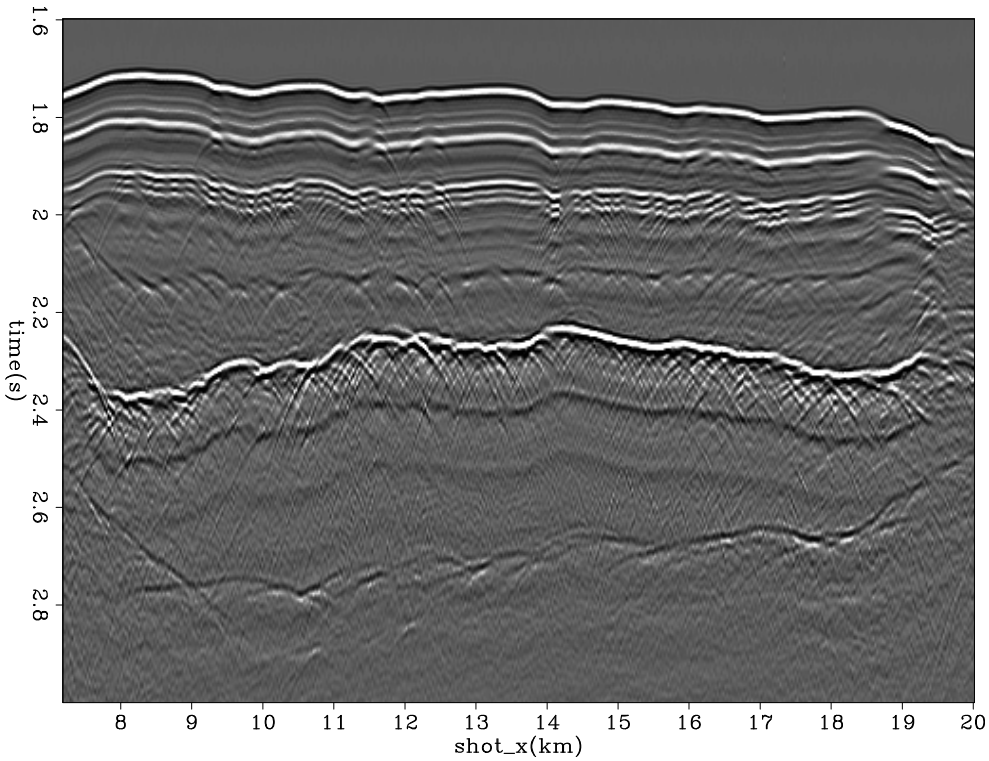

Figure 2: A common-offset section of a marine survey. [**ER**]

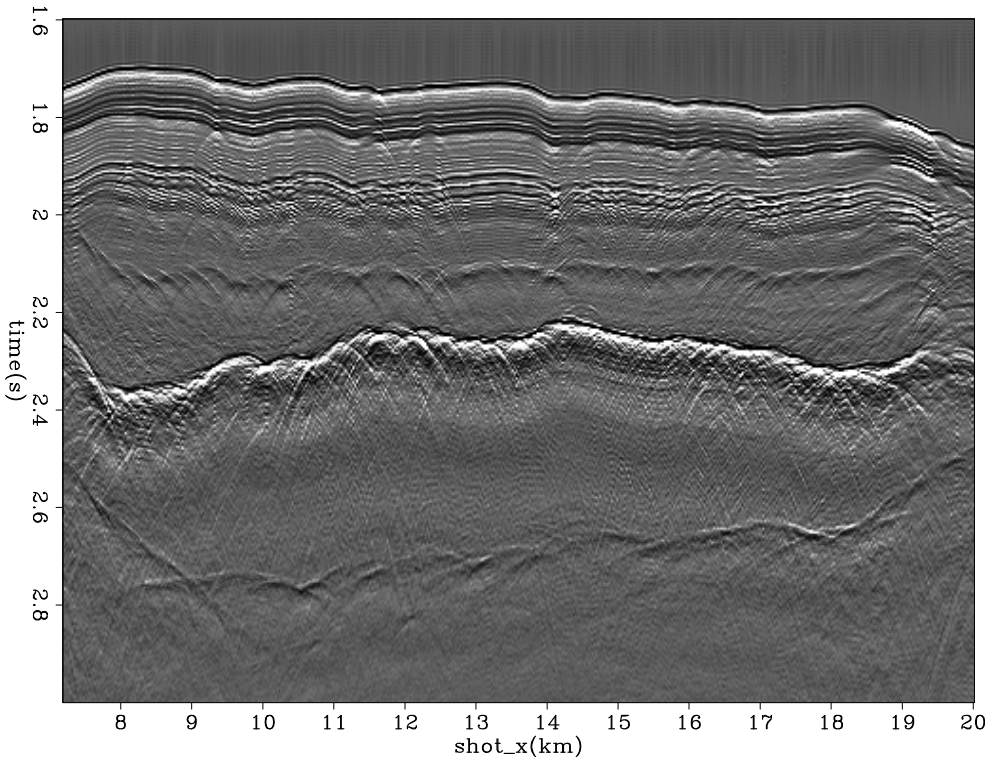

Figure 3: The common-offset section after applying the Burg PEF preconditioner. [**ER**]

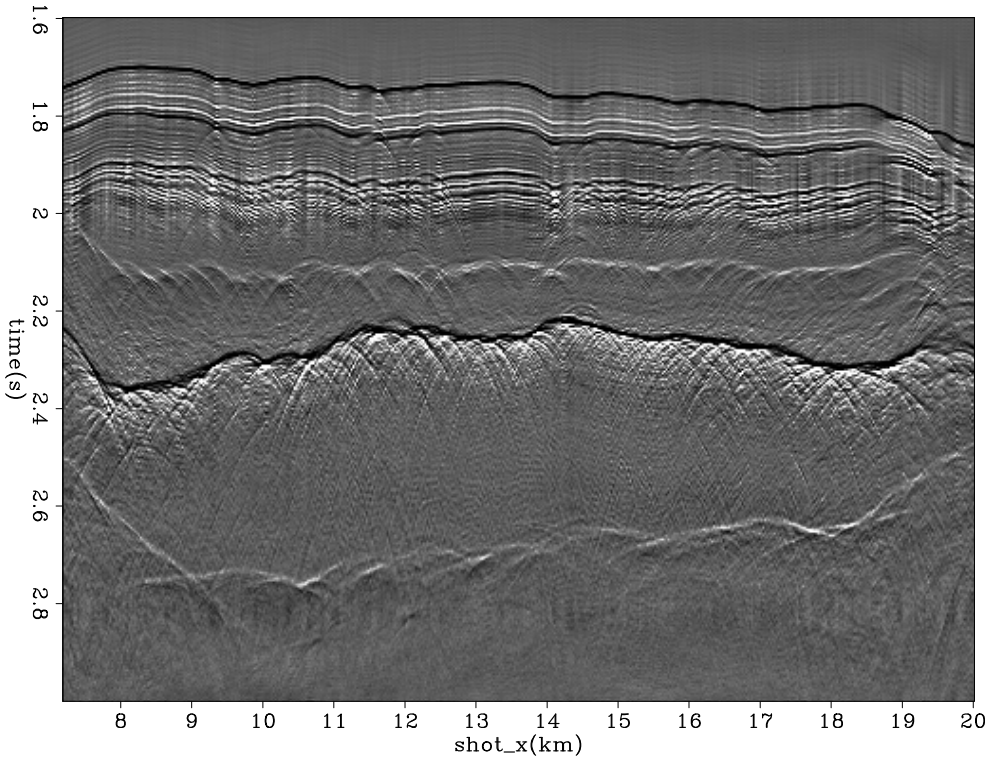

Figure 4: Logarithm Fourier-domain bidirectional deconvolution result. [**ER**]

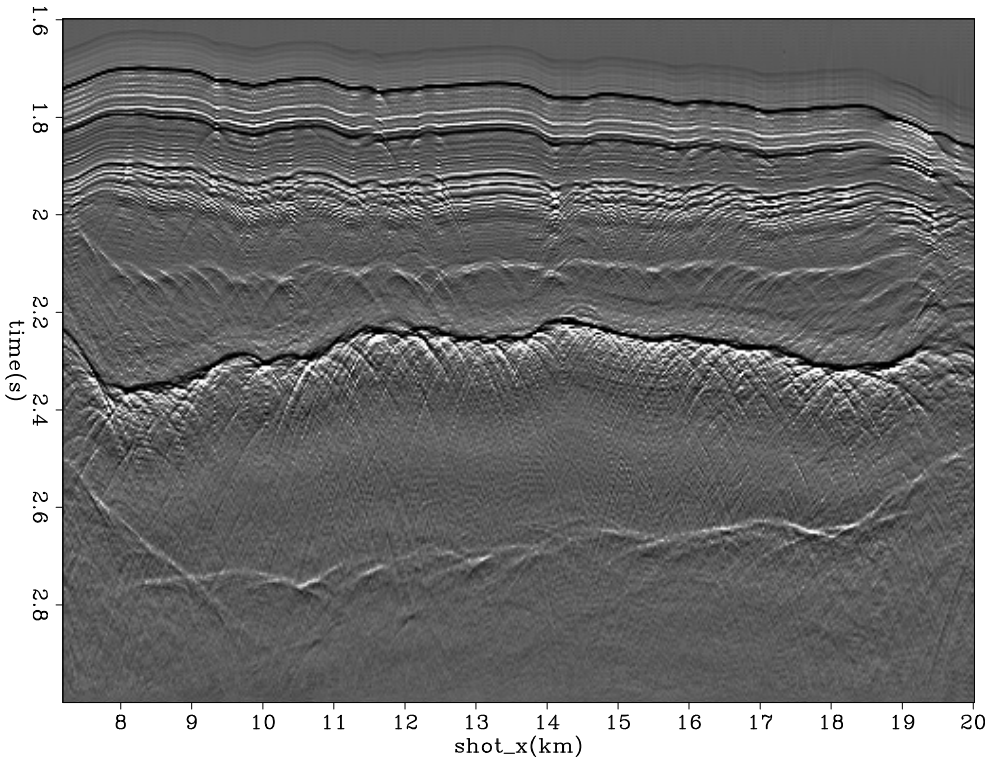

Figure 5: Time-domain (symmetric) bidirectional deconvolution result. [**ER**]

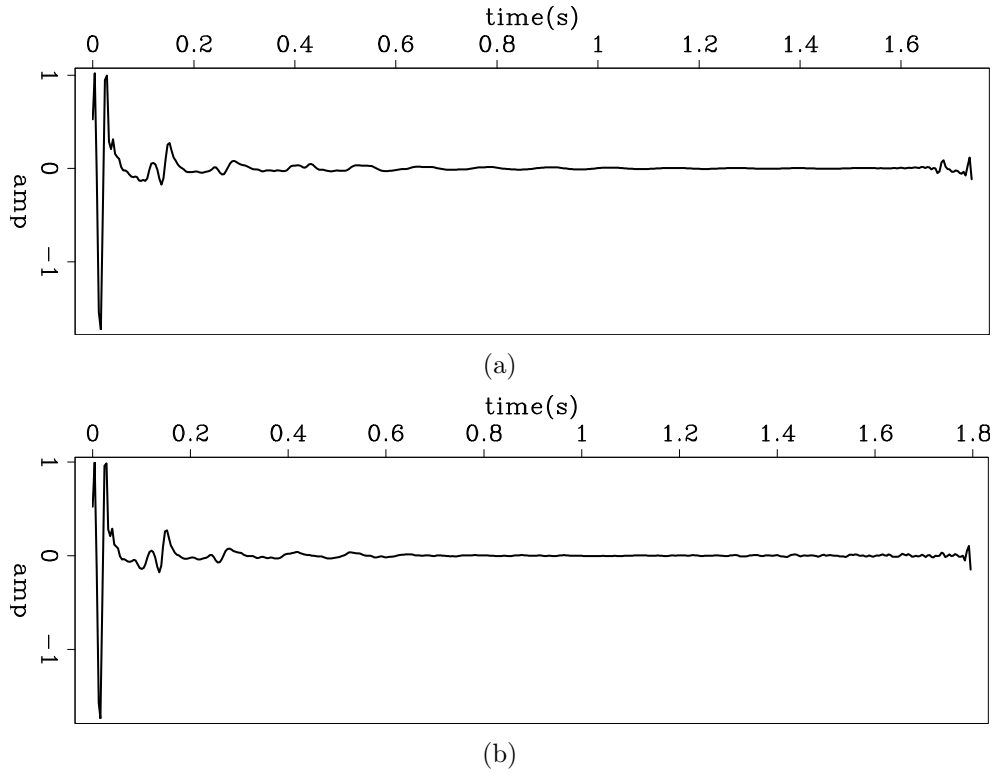

Figure 6: Estimated wavelet from (a) logarithm Fourier-domain and (b) time-domain (symmetric) bidirectional deconvolution. The estimation wavelets are the inverse deconvolution filters, calculated by inverting the frequency spectrum of the filter in the Fourier domain. Therefore the wavelet waveform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

Field data pre-stack shot gathers example

The previous common-offset data set is extracted from a pre-stack survey line. Figure 7 shows the pre-stack shot gathers for the whole survey line. In order to see clearly the deep feature, we use a gain function of t^3 on figure 7. This gain function is only applied here for display purposes and will never be used in later deconvolution procedures. Figure 8 shows two shot gathers in a marine survey. No gain is applied on these two shot gathers.

We find that the previous common-offset gather windowed the data not only in space but also in time. We want to get the data within the same time window as the previous common offset gather. But now we are working on the prestack gathers, and simply cutting a horizontal window from 1.6 s to 3 s will lose the far-offset part of the deep event, due to the moveout. We do not preform a Normal Move Out correction (NMO) to correct the moveout, because we do not want the stretch caused by NMO to damage the shot waveform. Instead, we perform a Linear Move Out correction (LMO), which shifts each trace with a constant time shift. We use a major event in the desired time window, which is the reflection from the top of the salt body, as a reference to calculate the shift time. Figure 9(a) shows the time-shift function to flatten the gather and two gathers after flattening.

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

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

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

From the deconvolution result of the shot gathers, we find the wavelet differs greatly between the near and far offsets. We decided to test different offset ranges within one shot gather. Figure 13 shows the whole offset range divided into three equal parts, displayed in three panels. The head wave of the salt body top event happens to start at the boundary between the near- and middle-offset panels.

Figure 14 shows the three offset range panels after applying Burg PEF preconditioning. Different PEF filters are estimated from each panel and applied separately. We combined the three panels for display here, but the boundaries among them are still evident. That means the wavelets of the three offset ranges are different.

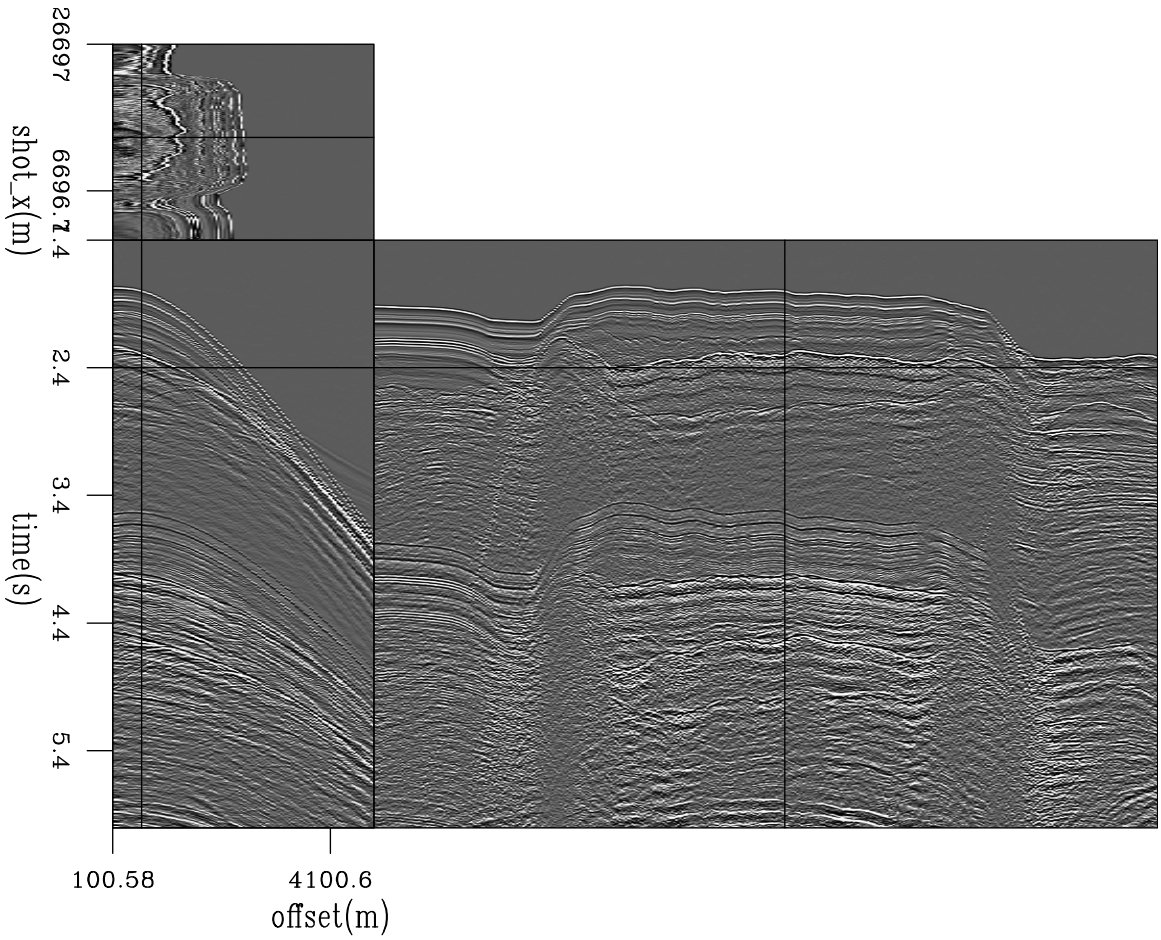

Figure 7: Whole marine pre-stack survey line. [**ER**]

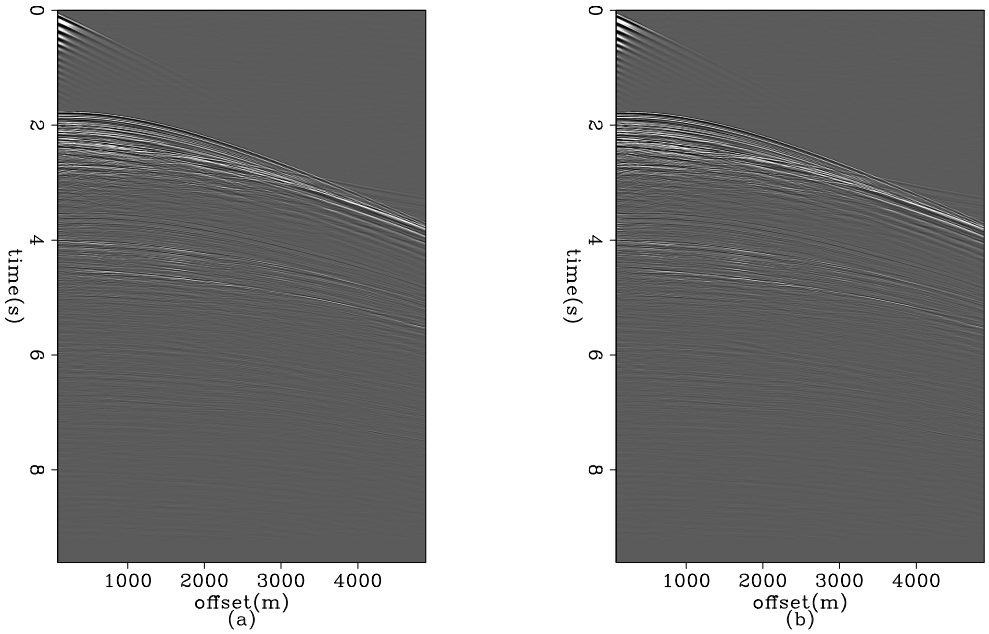

Figure 8: Two shot gathers. [**ER**]

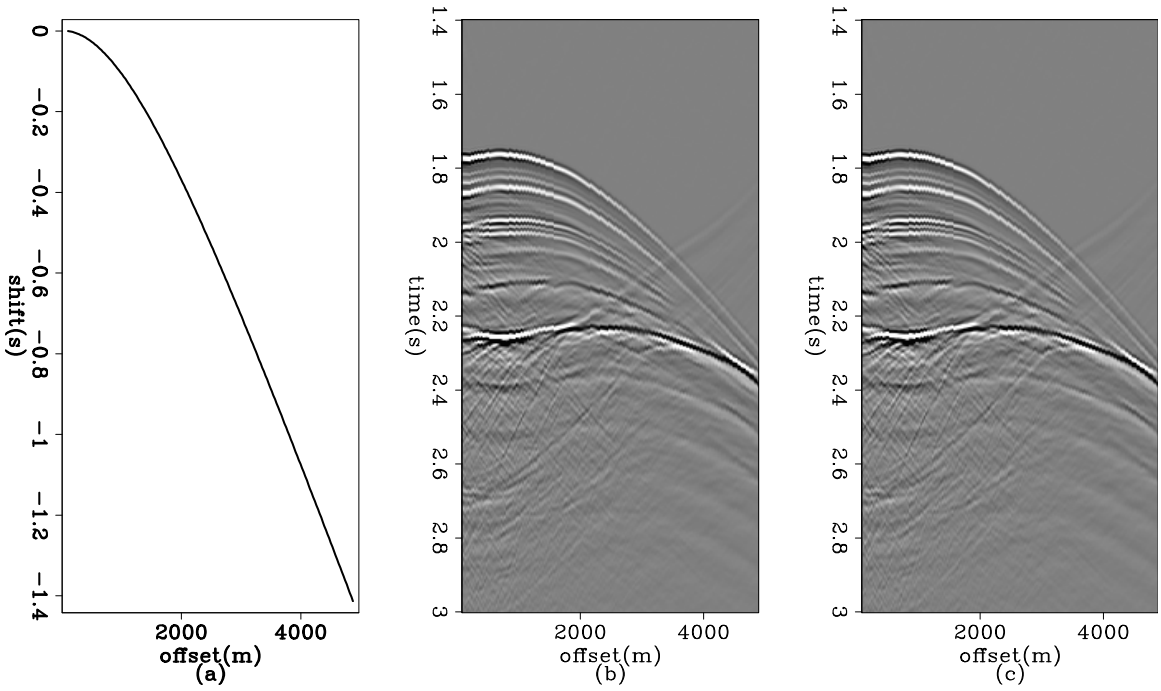

Figure 9: (a)The time shift function to flatten the gather; (b) and (c) two shifted shot gathers. [**ER**]

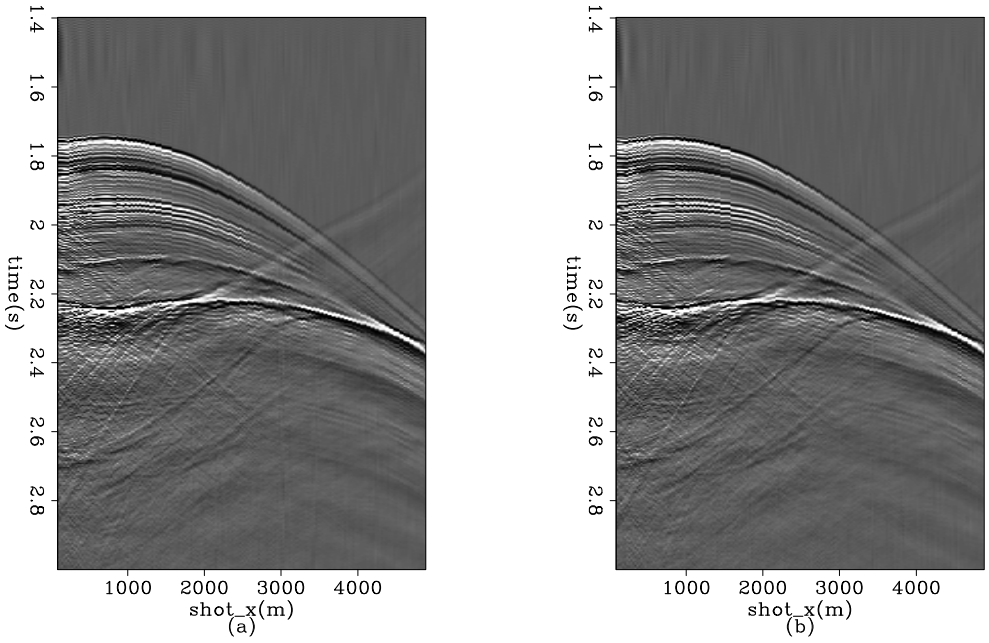

Figure 10: Two shot gathers after applying Burg PEF preconditioning. Different PEF filters are estimated and applied separately. [**ER**]

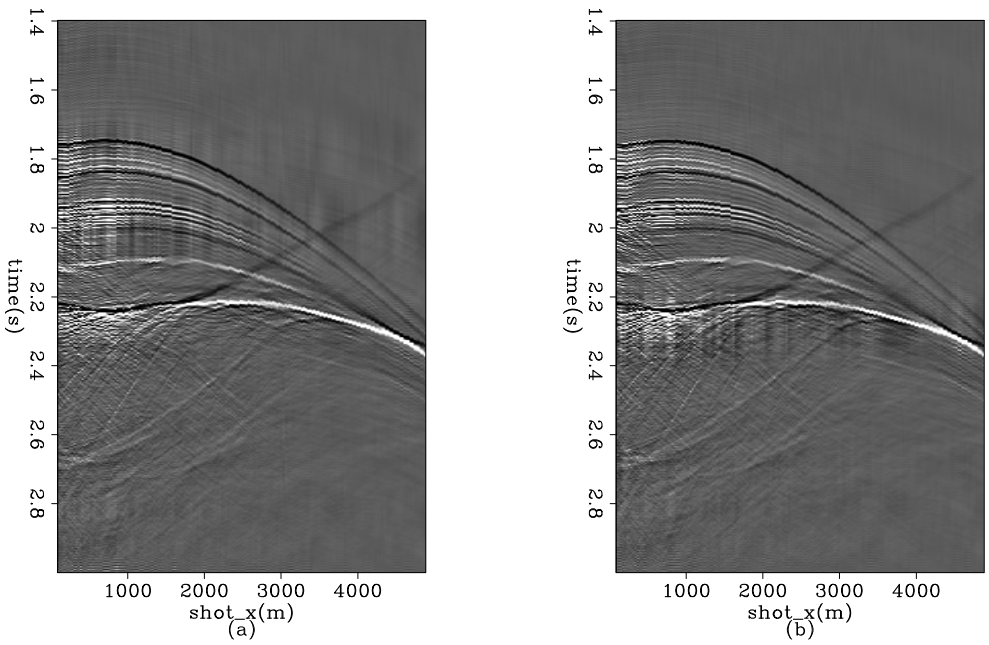

Figure 11: Two shot gathers after logarithm Fourier domain bidirectional deconvolution. The deconvolution is applied separatedly for each gather. [**ER**]

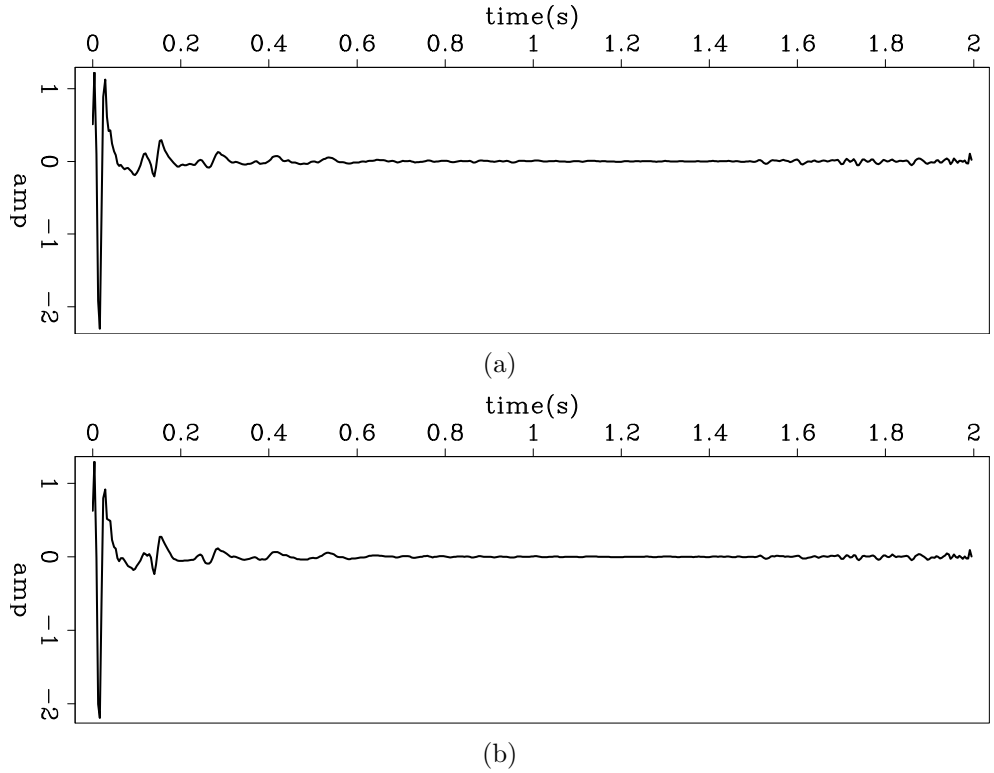

Figure 12: The estimated wavelets by logarithm Fourier-domain bidirectional deconvolution. (a) Shot gather 14000; (b) shot gather 14028.

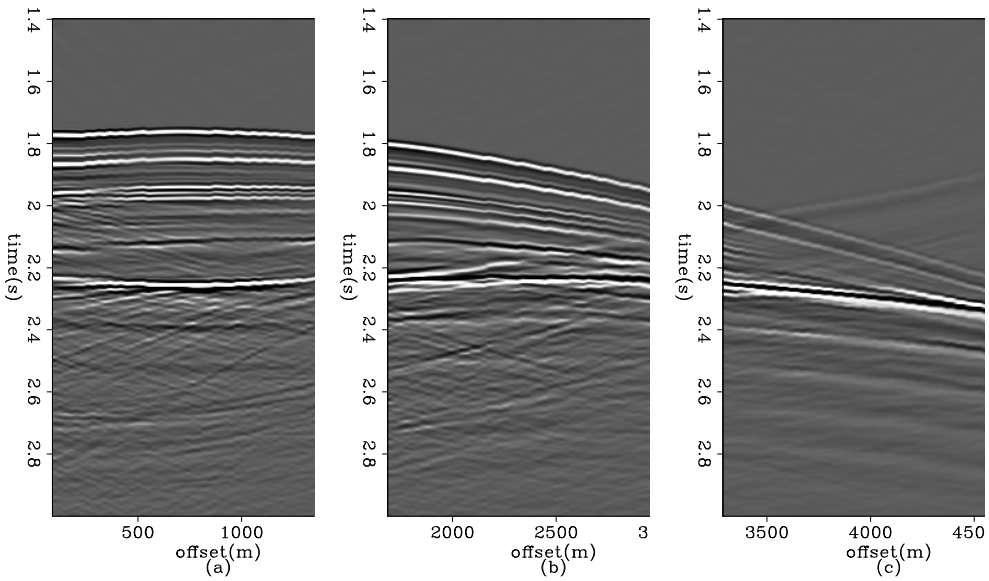

Figure 13: Three offset ranges of shot gather 14000. (a) near offset; (b) middle offset; (c) far offset. [**ER**]

Figure 15 shows the deconvolution results of the three offset panels. We can easily see a time shift of the salt body top from the near-offset panel to the middle-offset panel. This caused by the 90° phase shift we discussed before. Because we estimated the deconvolution filters separately for the three offset panels, the wavelet has a 90° phase shift and introduces a time shift into the output deconvolution spike.

The four panels of figure 16 show the different estimated wavelets from different offset ranges. Figure 16(a) is the wavelet from the whole shot gather, which is the same as 12(a). Figures 16(b), 16(c) and 16(d) are the wavelets estimated by near, middle and far offset, respectively. We see that the wavelet from the whole gather is similar to the wavelet of the near offset, and the wavelet of the middle and far offsets are quite similar to each other but very different from the one estimated from near offset. The wavelet has an almost 90° phase shift from the near-offset panel to the middle-offset panel, which is consistent with our observation of the raw data; as we discussed before, the major event has a 90° degree phase shift from the near-offset panel to the middle-offset panel.

We also tried multi-shot gathers with one filter. Figures 17 and 18 show deconvolution results for 39 and 451 shot gathers, respectively, and figures 19(a), 19(b) and 19(c) show a comparison among estimated wavelets of single shot gathers and multi-shot gathers. The wavelets are very similar, and from panels (a) to (c) we can see that using more shot gathers reduces jitter. These results tell us that the shot waveforms do not change significantly from shot to shot. This is consistent with our observations in the previous analysis of two shot gathers.

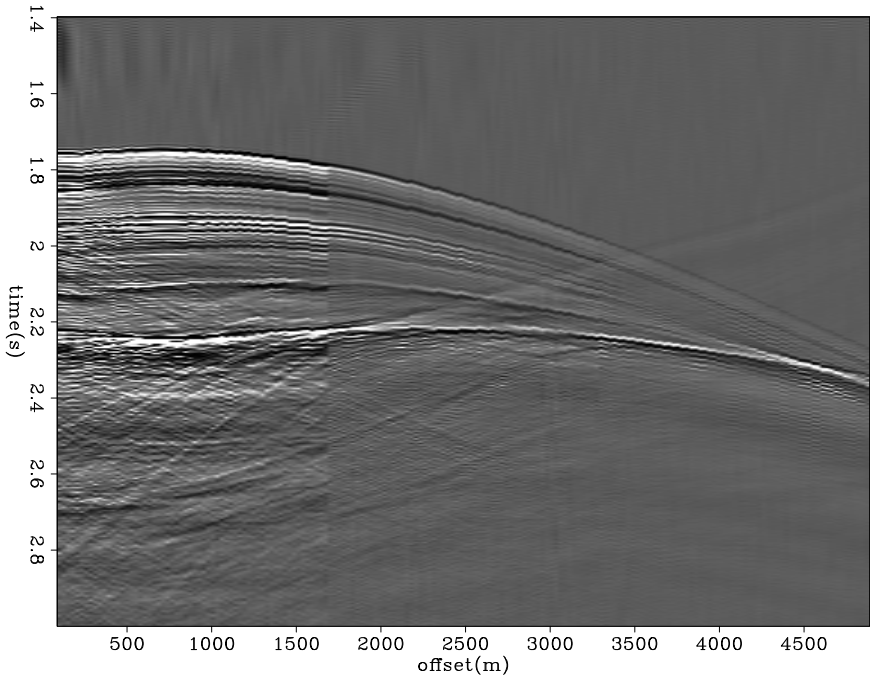

Figure 14: Three offset-range panels (combined together) after Burg PEF preconditioning. Different PEF filters are estimated and applied separately for each offset range. [**ER**]

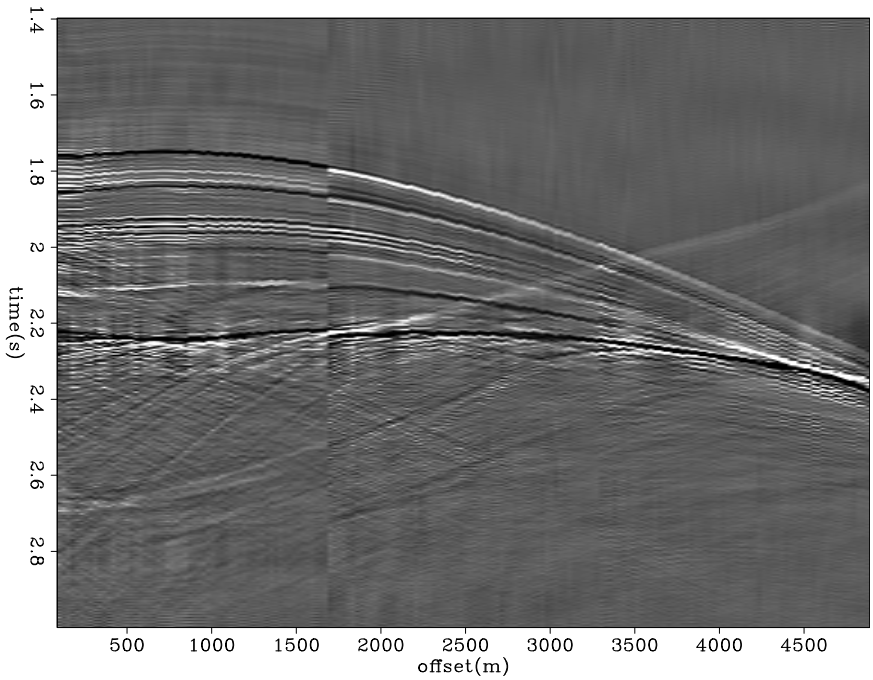

Figure 15: Three offset-range panels (combined together) after logarithm Fourier-domain bidirectional deconvolution. The deconvolution is applied separately for each range. [**ER**]

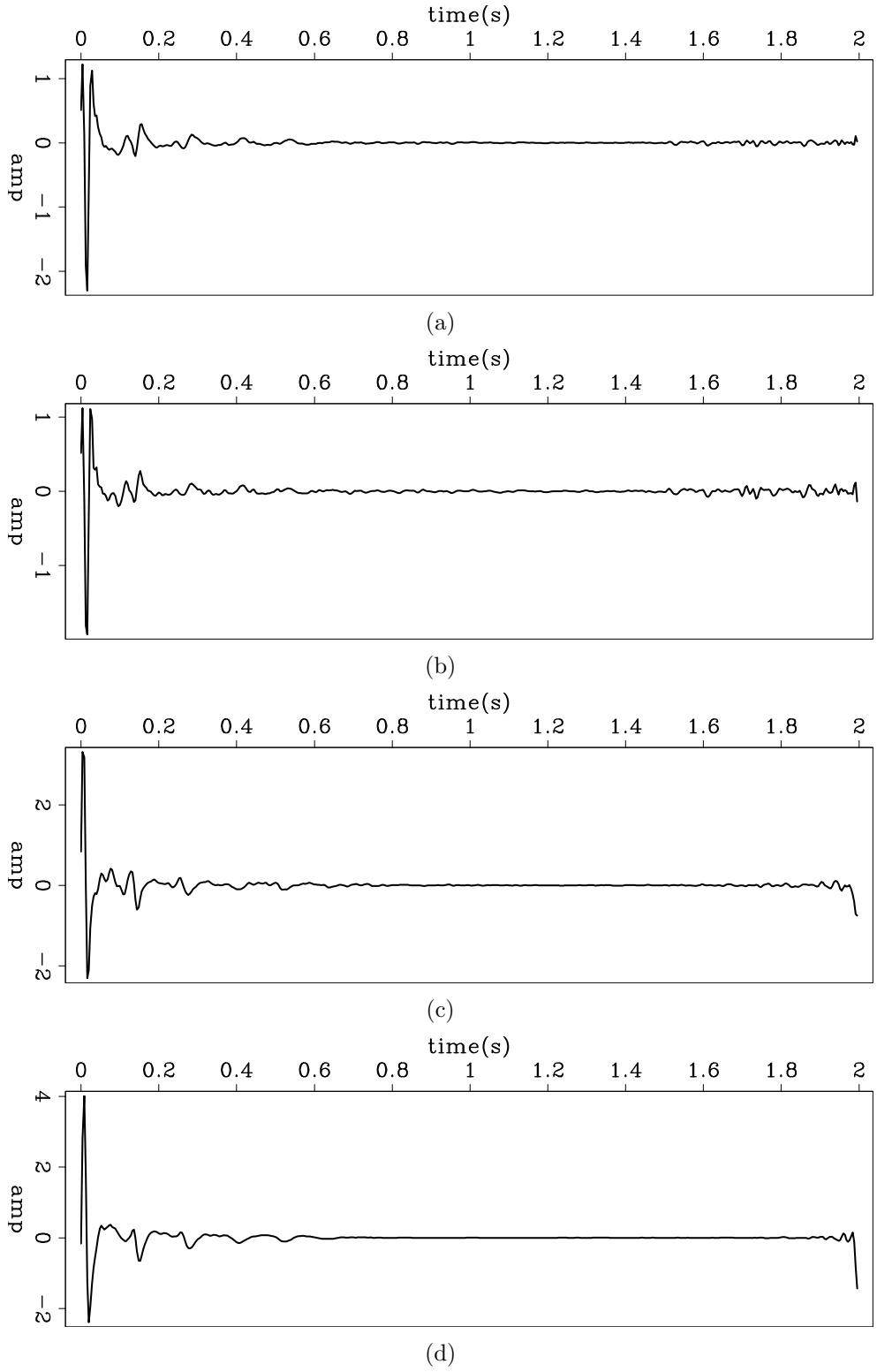

Figure 16: The wavelets estimated from (a) the whole offset range (same as 12(a)); (b) near offset; (c) middle offset; (d) far offset. [**ER**]

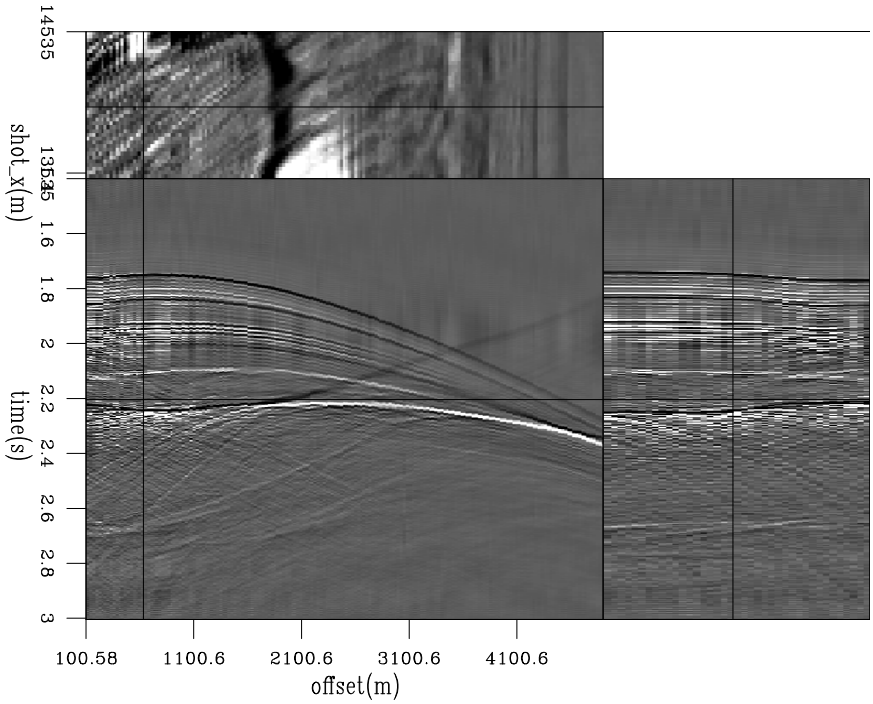

Figure 17: Logarithm Fourier-domain bidirectional deconvolution result on 39 shot gathers. [**ER**]

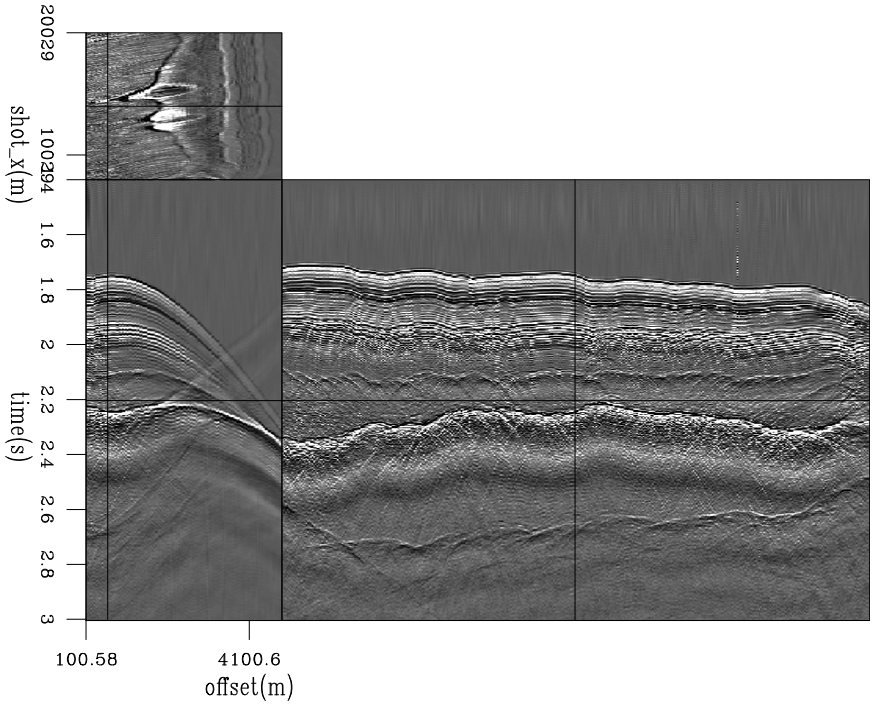

Figure 18: Logarithm Fourier-domain bidirectional deconvolution result on 451 shot gathers. [**CR**]

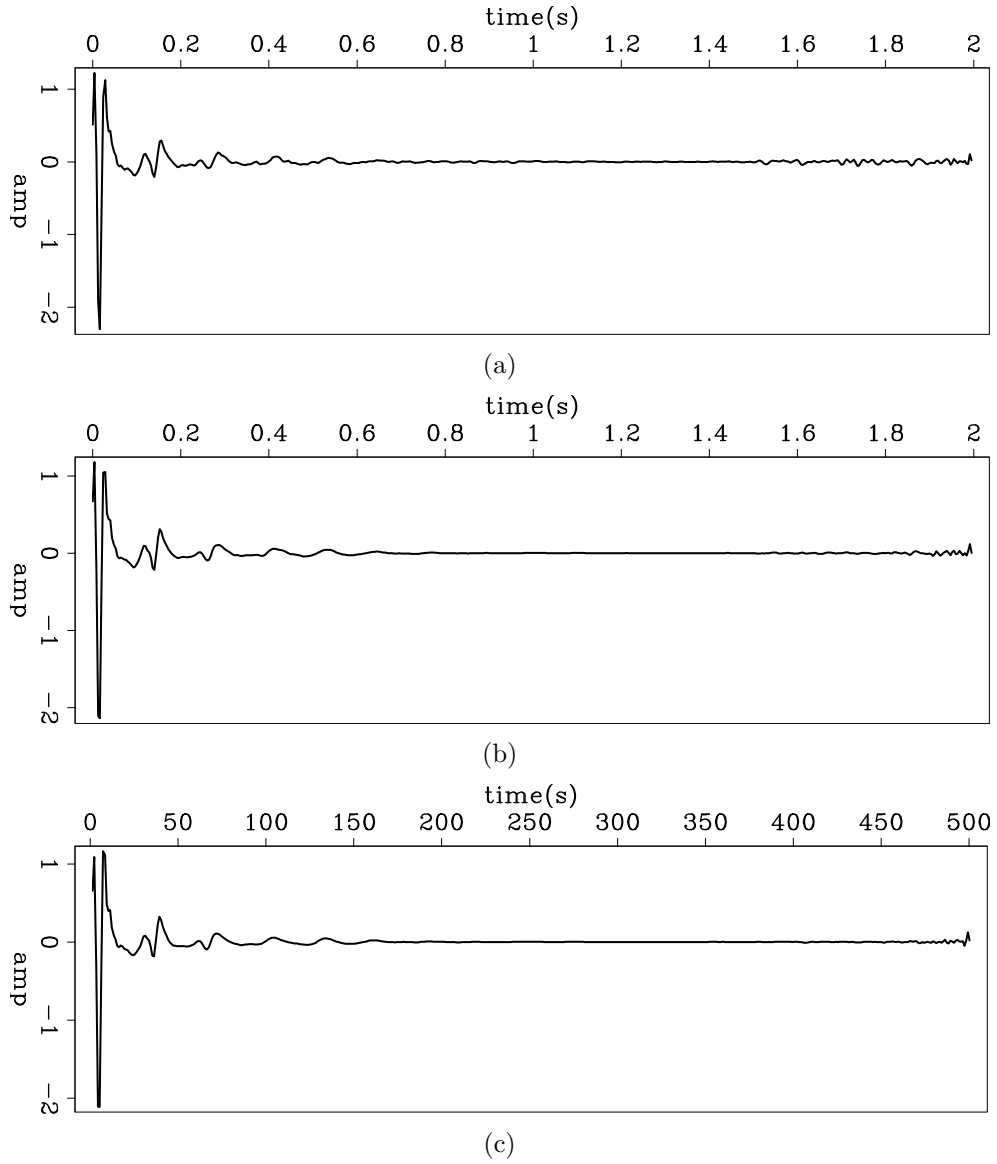

Figure 19: The wavelets estimated from (a) one shot gather (same as 12(a)); (b) 39 shot gathers (from shot location of 13500 meters to 14500 meters); (c) 451 shot gathers (from shot location of 8000 meters to 20000 meters). [**ER**]

CONCLUSIONS

We test the logarithm Fourier-domain method using several data sets including both synthetic data and field data. The results comfirm that this Fourier-domain method has advantages over previous time-domain methods, for both the speed of convergence and the quality of the result.

ACKNOWLEDGMENTS

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

REFERENCES

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

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

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

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

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

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