Stanford Exploration Project, Report 129, May 6, 2007, pages 75–85

Non-stationary interpolation in the f-x domain

William Curry

ABSTRACT

Interpolation of seismic data has previously been performed using non-stationary prediction-error filters in the t-x domain. This methodology is applied in the f-x domain, and is tested on synthetic data as well as 3D pre-stack marine data in various domains. Benefits include a reduced memory footprint and computational cost, which leads to the ability to interpolate in higher dimensions than with t-x domain interpolation. However, one severe drawback is the assumption of stationarity in time.

INTRODUCTION

Interpolation of seismic data can be performed in many different domains. A simple band-limited interpolation is achieved by padding in the f-k domain. Other methods in the f-k domain involve a sparse inversion of a non-uniform (or uniform) Fourier transform (Duijndam and Schonewille, 1999; Liu and Sacchi, 2004; Xu et al., 2005).

Interpolation beyond aliasing can be performed using prediction filters in the f-x domain (Spitz, 1991) as well as prediction-error filters in the t-x domain (Claerbout, 1999; Crawley, 2000). Interpolating in the t-x domain involves estimating a single filter for the entire dataset while the f-x domain method involves estimating an independent filter for each frequency and separately interpolating each frequency. Interpolation in f-x appears to be much faster than t-x and requires a much smaller memory footprint, while the t-x approach allows for the introduction of a time-variable weighting function and is also less sensitive to noise.

Interpolation in t-x can also be performed using non-stationary filters (Crawley, 2000). These prediction-error filters vary in space and time, and solve a global problem instead of the more traditional approach of breaking the problem up into stationary regions and solving those problems independently. Interpolation in the f-x domain can also be performed using non-stationary filters, but due to the Fourier transform these filters do not vary as a function of time. This can lead to some issues when dealing with non-stationarity in time.

Non-stationary interpolation in the f-x domain provides a much less expensive alternative to the t-x domain. With this method it is now possible to perform higher-dimensional interpolation for large-scale applications such as surface-related multiple prediction. The drawbacks include acausal energy appearing in the result as well as the inability to weight in time.

PEF-BASED INTERPOLATION

A prediction-error filter (PEF) can be estimated by minimizing the following residual,

$$\mathbf{0} \quad \approx \quad \mathbf{r} = \begin{bmatrix} d_2 & d_1 & d_0 \\ d_3 & d_2 & d_1 \\ d_4 & d_3 & d_2 \\ d_5 & d_4 & d_3 \\ d_6 & d_5 & d_4 \end{bmatrix} \begin{bmatrix} 0 & \cdot & \cdot \\ \cdot & 1 & \cdot \\ \cdot & \cdot & 1 \end{bmatrix} \begin{bmatrix} 1 \\ f_1 \\ f_2 \end{bmatrix} + \begin{bmatrix} d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_6 \end{bmatrix}, \tag{1}$$

where f_i are unknown filter values and d_i are known data values. In this case the filter has two free coefficients and there are seven data points. In practice, the PEF is multi-dimensional and contains many more coefficients. Once the PEF has been estimated it can be used in a second least-squares problem

$$\begin{aligned} \mathbf{S}(\mathbf{m}-\mathbf{d}) &= \mathbf{0} \\ \mathbf{F}\mathbf{m} &\approx \mathbf{0}, \end{aligned} \tag{2}$$

where $\mathbf{S}$ is a selector matrix which is 1 where data is present and 0 where it is not, $\mathbf{F}$ represents convolution with the PEF and $\mathbf{m}$ is the desired model. The first line in equation 2 is a hard constraint while the second line is not.

In order for this method to work, the data used as input to equation 1 needs to be similar to the desired output model in equation 2. The slightly different approximations used in t-x and f-x PEFs are discussed next.

T-X VERSUS F-X METHOD

Interpolation in the time domain is accomplished by first estimating the PEF on the input data with a spaced version of the PEF, as shown in Figure 1. The amount of spacing corresponds to the factor of interpolation. Once the PEF has been estimated on the input data, it is returned to the original spacing and then the second step of the interpolation takes place, where the interpolated model space is regularized by the PEF. This PEF is estimated on the entire input dataset and is applied on the entire output dataset as two global problems.

Interpolation in the f-x domain is performed by estimating a PEF for a single frequency; for a two-dimensional f-x scenario this would be a one-dimensional PEF along x. A unique PEF is estimated at each frequency of the input data, and this PEF is then used to regularize the output data at a higher frequency equal to the original frequency multiplied by the interpolation factor. In practice, the data used to create the PEF is padded in time before being transformed to frequency so that the frequencies after multiplication by the scale factor correspond to the output frequencies that are desired.

Because the f-x domain approach interpolates each frequency independently, the input data can be broken along the frequency axis and interpolated in parallel or sequentially. This means that the memory footprint of the f-x method is smaller by a factor corresponding to the size of

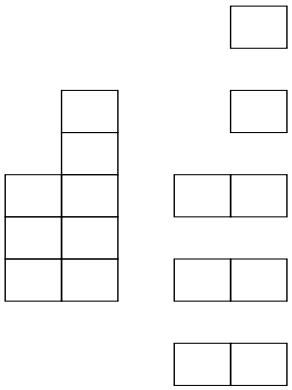

Figure 1: Left: PEF on output data; right: PEF spaced by a factor of two on input data with half the sampling in space. bill1-space [NR]

the time (or frequency) axis. Since this is typically the largest axis, this can mean a reduction in the memory use by two to three orders of magnitude. This savings of memory means that more input data can be held in memory, and simultaneous interpolation of more dimensions is possible with the f-x approach. For example, Figure 2 was quickly created on a laptop in a few minutes, whereas the same interpolation in t-x would strain a larger workstation. The result

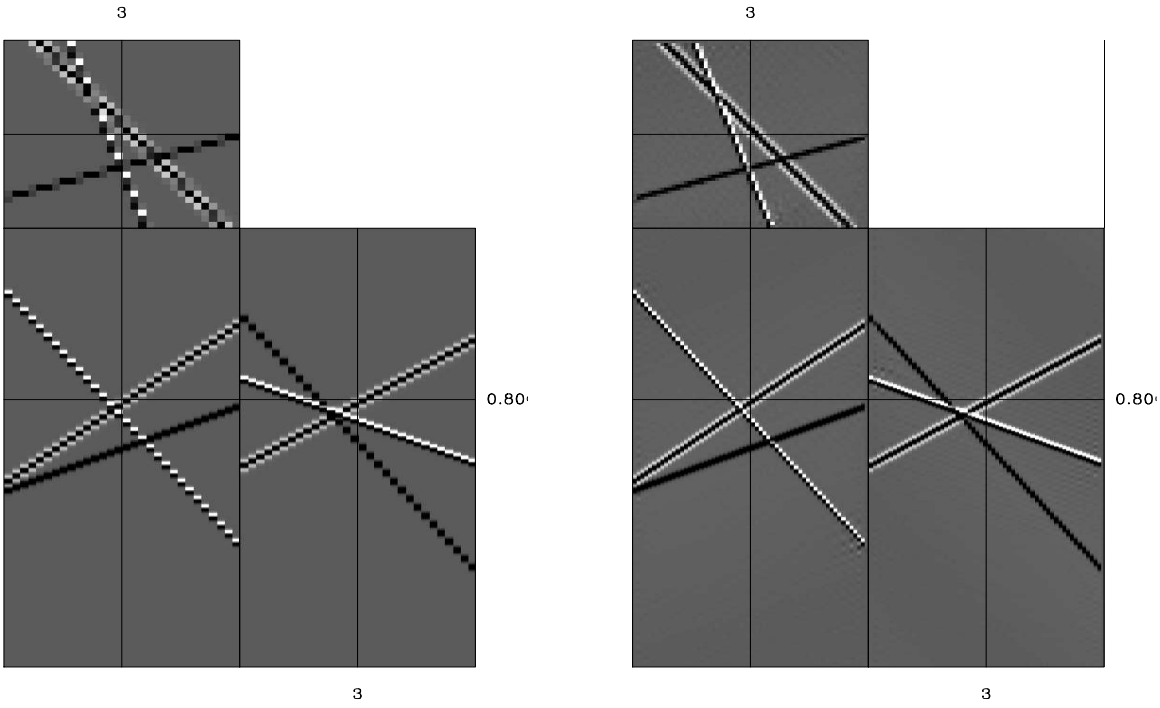

Figure 2: Left: one cube from 4D input data; right: cube interpolated by a factor of eight. The PEF used was 3D with 4x4x4 coefficients. bill1-fxpef [ER]

shown in Figure 2 is interpolated by a factor of two on three of the four axes. This type of approach not only provides a better prediction by increasing the amount of input information but also speeds up the interpolation process by simultaneously interpolating more dimensions, which would require a cascade of lower-dimension interpolations with a t-x approach.

One reason why this f-x interpolation is much faster than t-x interpolation is that the t-x

approach captures spectral information from the data which is not used during the interpolation. F-x PEFs are a much more compact and efficient way to capture the dip information in the training data. Also, the parameterization on the PEF in t-x is less intuitive than in f-x. In f-x, the size of the PEF roughly corresponds to the number of dips that it would be able to predict. In the t-x case, the length of the PEF in time also comes in to play as it relates to the maximum dip that could be predicted by the filter. Also, the spacing of the PEF assumes that the data are very oversampled in time, so if the data are not high-cut filtered before the PEF estimation the filter could be time-aliased. Since in f-x PEFs the lower frequencies are explicitly used to interpolate the higher frequencies this problem does not happen, although it also relies upon oversampling in time.

NON-STATIONARY F-X INTERPOLATION

Since seismic data has dips that vary in space and time, the stationary methodology shown in the previous section cannot be blindly applied to entire data sets. Instead the data are either broken up into regions that are assumed to be stationary or the PEF that is estimated on the data is non-stationary. Non-stationary PEFs estimated in the t-x domain have been used to interpolate data with dips varying in both space and time. Non-stationary PEFs that are estimated in f-x can also be used to interpolate data, but since the dimensionality of the filter is less (with no time/frequency component) and the data are Fourier transformed prior to interpolation, the dip spectrum is assumed to be stationary in time.

In addition to the issue with time non-stationarity, a time-variable weighting function which could be implemented in t-x could not be implemented in f-x as the frequencies are solved independently. Figure 3 shows the qdome synthetic that is both non-stationary in space and time being interpolated by a factor of 16, 4 in each spatial dimension. The issue of non-stationarity in time in Figure 3 does not appear to be a very large problem. While the flat layers at the top and bottom of the interpolated result do contain some erroneous dips, their amplitude is quite weak compared to the dominant dips in that location.

REAL DATA EXAMPLES

Figure 4 shows various interpolations of a common-offset section from a sail line from a 3D marine survey. Figure 4(a) shows the original common-offset section, 4(b) is a 2D interpolation where the only input to the interpolation is what is shown in Figure 4(a). Figure 4(c) shows the result of a 3D interpolation where a single receiver cable as well as the common-offset section from the sail line in (a) were used. Figure 4(d) is the result of a 4D interpolation, where all four receiver cables were used as well as what was used in 4(c). The most obvious issue with all of the interpolations is the introduction of noise before the water bottom reflection. This is a result of the assumption of stationarity in time, putting all dips at all times. This noise is much more prevalent in the 2D interpolation than in the 3D and 4D interpolations. The quality of the 3D interpolation on the whole is much higher than the 2D interpolation, whereas the 4D interpolation is almost identical to the 3D case. This is not overly surprising

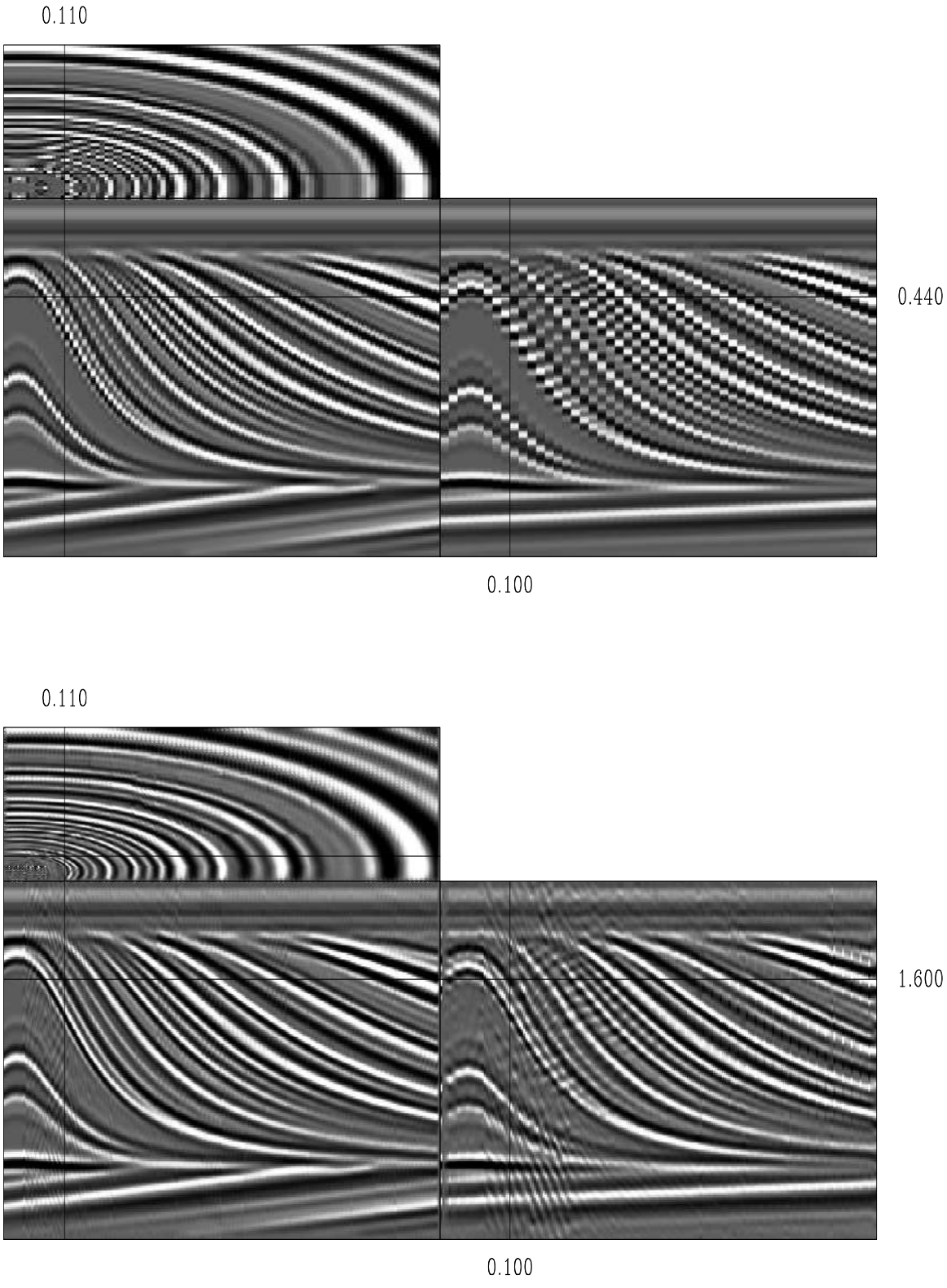

Figure 3: Top: qdome input data; bottom: qdome interpolated by a factor of 4 in each direction. PEF size is 5x5. bill1-nspef [ER]

since there are only 4 points in the 4th dimension (cross-line offset) which do not contribute much to the result.

Next, I compare the interpolation of receiver cables using a 3D and a 4D interpolation for a single shot. Figure 5 shows a 3D shot with the front panel as a recorded receiver cable, interpolated by a factor of 2. The number of cables is also interpolated from 4 to 8. There is an obvious improvement associated with moving from a 3D to 4D interpolation. In Figure 6the same shot is interpolated in 3D and 4D, but the front panel of the cubes now show an interpolated receiver cable. This result looks poor in both cases, although noticeably better for the 4D interpolation. The issue of non-stationarity in time is crippling, although the overall trend of the water-bottom arrival is captured in both cases. In combination with the previous Figure shows how the length of the axis being added to the interpolation has a profound effect on the result. The interpolated cables are of much worse quality than the interpolated inline receivers, which is due to both the larger distance between the cables and the small number of receiver cables in the acquisition.

CONCLUSIONS AND FUTURE WORK

Non-stationary f-x domain interpolation appears to be a promising route to generate the vast quantities of data needed for surface-related multiple elimination. The process is embarrassingly parallel and leaves the data in a similar state as would be needed by subsequent processing algorithms. The process is also much faster than a t-x approach, and needs orders of magnitude less memory to run. However, the issue of non-stationarity in time is large, and impossible to ignore. This problem is more pronounced on the poorest-sampled axes, such as cross-line offset. Applying this non-stationary f-x methodology in time windows should hopefully address this problem.

One important thing to note is that while higher-dimensionality of the interpolation could improve the end result, this was only the case when the axis that was added was well-sampled, such as the in line source or inline offset axis. The addition of the 4 points on the cross-line offset axis did little to the end result.

In order to combine this approach with an extrapolator to quickly generate input data for a 3D surface-related multiple prediction, the next steps will be to apply this method in time-windows to address the time-non-stationarity issue as well as attempt to apply this method in the log-stretch frequency domain so that an efficient AMO operator could be applied.

ACKNOWLEDGMENTS

I would like to thank VeritasDGC for providing the dataset and Dr. Jan Pajchel of Norsk Hydro for his help in securing permission to use the dataset.

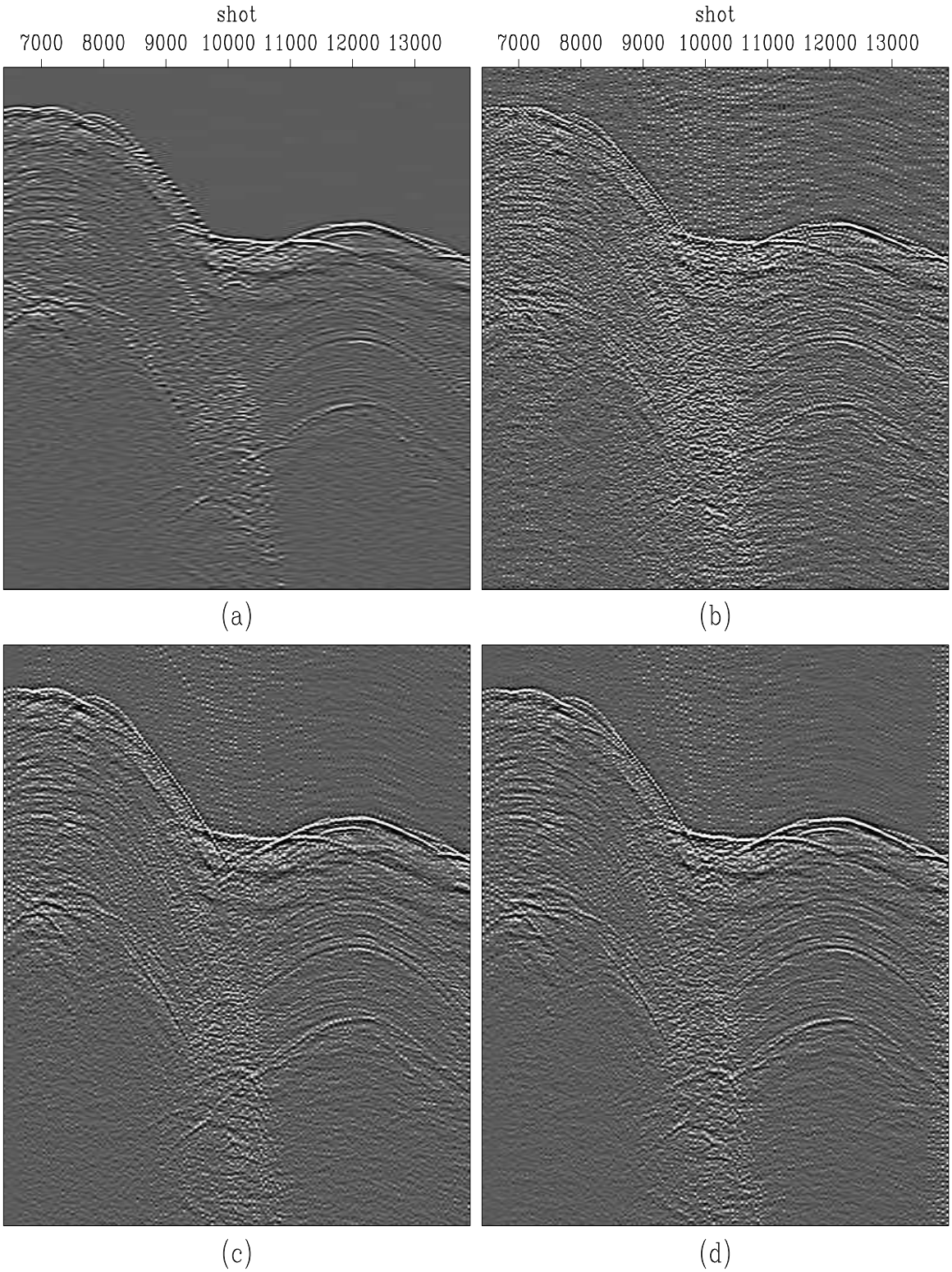

Figure 4: Interpolation of a common-offset section. (a) Input data. (b) 2D interpolation in shot,frequency. (c) 3D interpolation in shot, offset$_x$, frequency. (d) 4D interpolation in shot, offset$_x$, offset$_y$, frequency. bill1-coffsetcomp [CR]

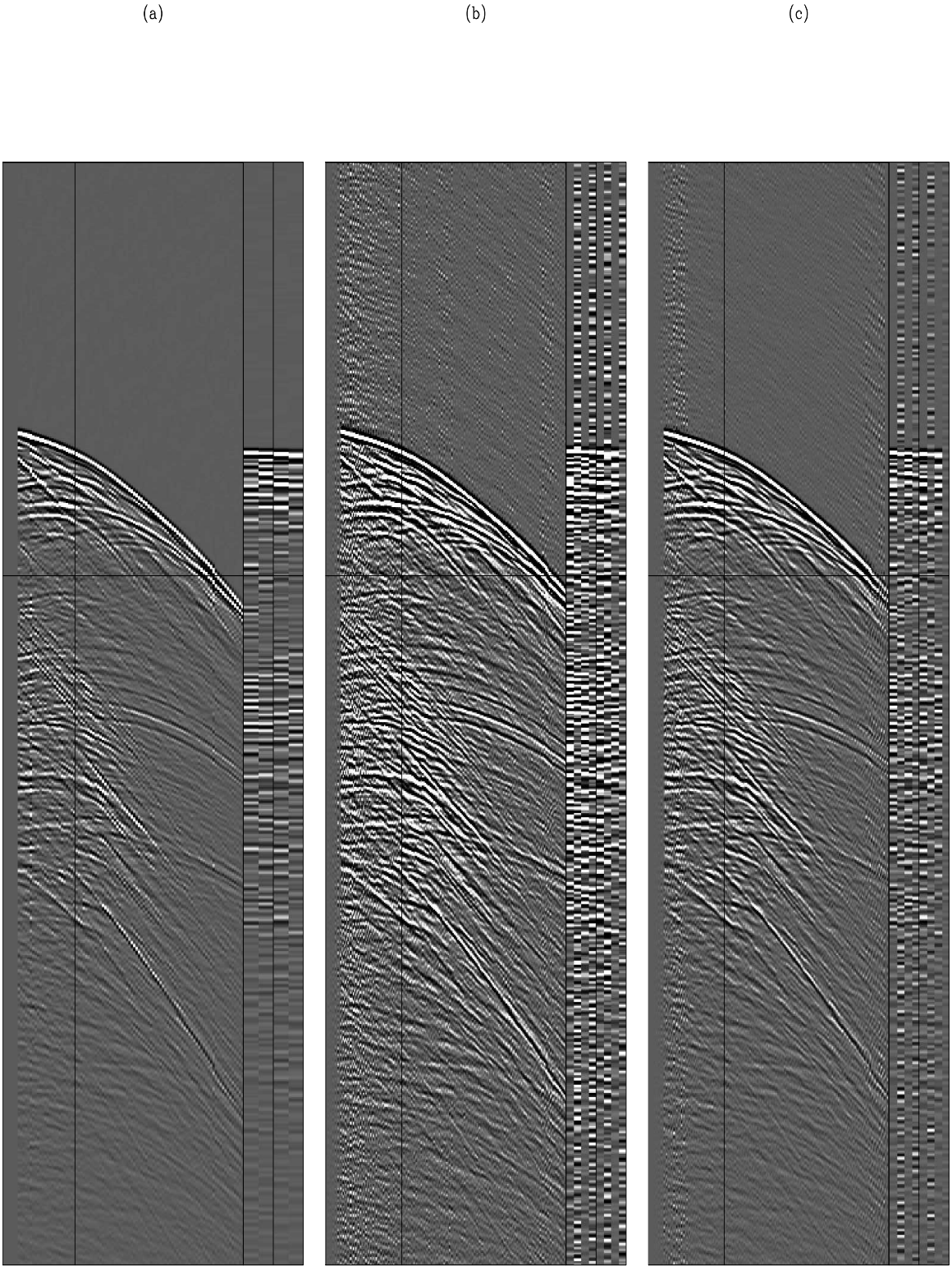

Figure 5: Interpolation of a single shot. The front panel is a recorded cable. (a) Input data with 4 streamers. (b) 3D interpolation of receivers along a cable. (c) 4D interpolation in shot, offset_x, offset_y, frequency of same shot as in (a). bill1-shotcomp1 [CR]

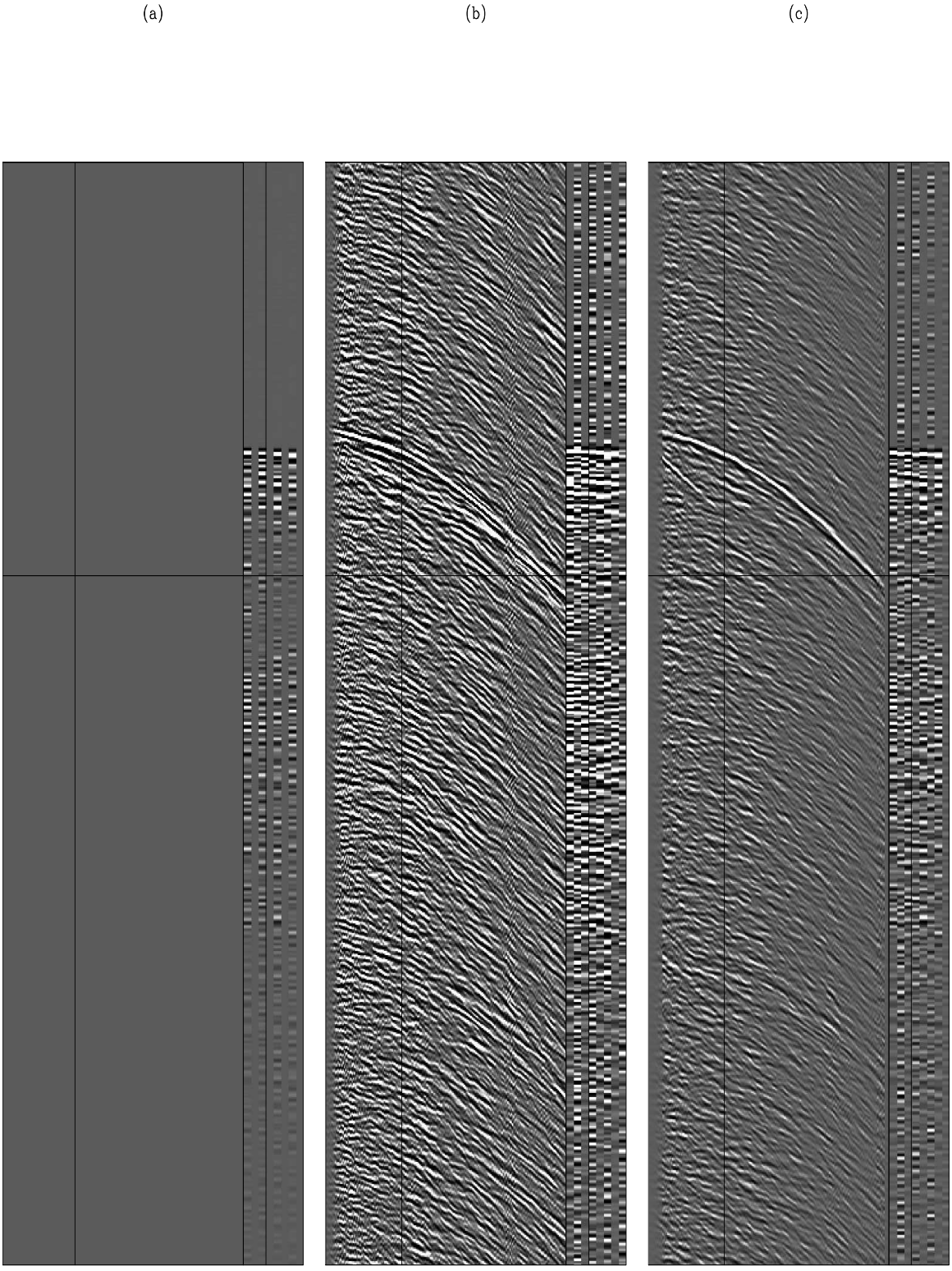

Figure 6: Interpolation of a single shot. The front panel is an interpolated cable (a) Input data. (b) 3D interpolation of a streamer using a single shot. (c) 4D interpolation in shot, offset$_x$, and offset$_y$, using multiple shots. bill1-shotcomp2 [CR]

REFERENCES

Claerbout, J., 1999, Geophysical estimation by example: Environmental soundings image enhancement: Stanford Exploration Project, http://sepwww.stanford.edu/sep/prof/.

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

Duijndam, A. J. W. and M. A. Schonewille, 1999, Nonuniform fast Fourier transform: Geophysics, **64**, no. 2, 539–551.

Liu, B. and M. D. Sacchi, 2004, Minimum weighted norm interpolation of seismic records: Geophysics, **69**, no. 6, 1560–1568.

Spitz, S., 1991, Seismic trace interpolation in the F-X domain: Geophysics, **56**, no. 06, 785–794.

Xu, S., Y. Zhang, D. Pham, and G. Lambare, 2005, Antileakage Fourier transform for seismic data regularization: Geophysics, **70**, no. 04, V87–V95.