Interpolation with smoothly nonstationary prediction-error filters

Sean Crawley, Robert Clapp, and Jon Claerbout

SUMMARY

Building on the notions of time-variable filtering and the helix coordinate system, we develop software for filters that are smoothly variable in multiple dimensions. Multiscale prediction-error filters (PEFs) can estimate dips from recorded data and use the dip information to fill in unrecorded shot or receiver gathers. The data are typically divided into patches with approximately constant dips, with the requirement that the patches contain enough data samples to provide a sufficient number of fitting equations to determine all the coefficients of the filter. Instead, we estimate a set of smoothly varying filters in much smaller patches, as small as one data sample. They are more memory-intensive to estimate, but the smoothly varying filters do give more accurate interpolation results than discrete patches, particularly on complicated data.

To control the smoothness of the filters. We design filters like directional derivatives that Clapp et al. (1998) call "steering filters". They destroy dips in easily adjusted directions. We use them in residual space to encourage dips in the specified directions. We develop the notion of "radial-steering filters", i.e., steering filters oriented in the radial direction (lines of constant x/t in (t, x) space). Break a common-midpoint gather into pie shaped regions bounded by various values of x/t. Such a pie-shaped region tends to have constant dip spectrum throughout the region so it is a natural region for smoothing estimates of dip spectra or of gathering statistics (via 2-D PEFs). In this paper we use smoothly variable PEFs to interpolate missing traces, though they may have many other uses.

Finally, since noisy data can produce poor interpolation results, we deal with the separation of signal and noise along with missing data.

INTRODUCTION

Claerbout (1998) describes a helical coordinate to cast multi-dimensional filtering as one dimensional, enabling the use of some well-developed signal processing theory in applications including missing data interpolation (Fomel et al., 1997) and low-cut filtering (Claerbout, 1998a). To account for nonstationarity in the data, missing data interpolation with PEFs is typically done in patches or gates where dips are assumed to be approximately stationary (Abma and Claerbout, 1995; Spitz, 1991). Each patch constitutes an independent problem, though they may overlap. The smaller the patch, the more stationary the data is likely to be within the patch; however there is a lower limit on the patch size, because a patch must contain enough data to provide fitting equations for all the filter coefficients. Claerbout (1997) describes a method for estimating smoothly time-varying PEFs without patching. We use the helix to extend the idea of smooth time-variable PEF estimation to smooth time- and space-variable PEF estimation. The new PEFs can perform better at interpolating missing data than PEFs estimated in independent patches.

Clapp et al. (1998) show how to use space-variable inverse steering filters to smooth in adjustable directions, and they show how to solve empty-bin problems filling in missing data along the directions of the steering. We use space-variable steering filters to control the direction of smoothness between PEFs. We orient the steering filters radially in a CMP gather to encourage PEFs to have the same dip information along lines of constant x/t, where data tends to have constant dip spectra. In this paper we review the theory for estimating smoothly varying PEFs, and show examples of their application to missing data interpolation. We describe an improvement to filter estimation for CMP gathers using "radial-steering filters." Finally, we add the notion of signal and noise separation for interpolating noisy data.

TIME- AND SPACE-VARYING PEFS

The time dip of seismic data changes rapidly along many axes, so a single PEF can only represent a small amount of data. Often we divide the data into patches, where we assume the data have constant dips. Because seismic data have curvature and may not be well represented by piecewise-constant dips, it is appealing to extend the idea of time-variable filtering to include spatial dimensions as well, and have smoothly varying PEFs to represent curved events.

We decrease the patch size, to as small as a single data sample, changing the problem from overdetermined to very underdetermined. We can estimate all these filter coefficients by the usual formulation, supplemented with some damping equations, say

$$\begin{aligned} \mathbf{0} &\approx \mathbf{YKa} + \mathbf{r}_0 \\ \mathbf{0} &\approx \epsilon\, \mathbf{Ra} \end{aligned} \tag{1}$$

where $\mathbf{R}$ is a roughening operator, $\mathbf{Y}$ is convolution with the data, and $\mathbf{K}$ is a known filter coefficient mask.

When the roughening operator $\mathbf{R}$ is a differential operator, the number of iterations can be large. We can speed the calculation immensely and make the equations somewhat neater by "preconditioning". When we define a new variable $\mathbf{p}$ by $\mathbf{a} = \mathbf{Sp}$ and insert it into (1) we get

$$\mathbf{0} \approx \mathbf{YKSp} + \mathbf{r}_0 \tag{2}$$

$$\mathbf{0} \approx \epsilon\, \mathbf{RSp} \tag{3}$$

Now, because the smoothing and roughening operators are somewhat arbitrary, we may as well replace $\mathbf{RS}$ by $\mathbf{I}$ and get

$$\mathbf{0} \approx \mathbf{YKSp} + \mathbf{r}_0 \tag{4}$$

$$\mathbf{0} \approx \epsilon\, \mathbf{Ip} \tag{5}$$

We solve for $\mathbf{p}$ using conjugate gradients. To see $\mathbf{a}$, we just use $\mathbf{a} = \mathbf{Sp}$. To simplify things, we could drop the damping (5) and keep only (2); then to control the null space, we need only to start from a zero solution and limit the number of iterations.

For $\mathbf{S}$ we can use polynomial division by a Laplacian or by filters with a preferred direction. If the data are CMP gathers, it is attractive to use radial filters, which are explained further down.

INTERPOLATING MISSING TRACES

We estimate missing data in two steps of linear least squares (Claerbout, 1992). The first step is estimation of PEFs. After the PEFs have been estimated they are used to fill in the empty trace bins. This is the second step of least squares. We want the recorded and estimated data to have the same dips. Since the dip information is now carried in the PEFs, this is once again specifying that the convolution of the filter and data should give the minimum output, except that now the filters are known and the data is unknown. We constrain the data by specifying that the originally recorded data cannot change. To separate the known and unknown data we have a known data selector $\mathbf{K}$ and an unknown data selector $\mathbf{U}$, with $\mathbf{U} + \mathbf{K} = \mathbf{I}$. These multiply by 1 or 0 depending on whether the data was originally recorded or not. With $\mathbf{A}$ signaling convolution with the PEF and $\mathbf{y}$ the vector of data, the regression is $0 \approx \mathbf{A}(\mathbf{U}+\mathbf{K})\mathbf{y}$, or $\mathbf{AUy} \approx -\mathbf{AKy}$.

While a PEF at every sample works well for destroying the data, it is not the best choice for reconstructing it; interpolation with PEFs estimated at every data point gives poor results (in addition to requiring extravagant memory usage). One answer is just that zero is not the correct value of ϵ in (5); but we can greatly improve the results and decrease the memory usage without adding equations, by using very small patches, such as $2 \times 2 \times 2$; small enough that the assumption of stationarity within a patch holds. This is similar to putting an extra roughener in the damping equation, in that it is essentially an infinite penalty on variations of **p** between small groups of samples, and it has the important economizing effect of reducing the memory allocation. In the method where the patches are independent (Crawley, 1998), the number of filter coefficients puts a lower bound on patch size; the problem has to stay well overdetermined to produce a useful PEF. Using smoothly nonstationary filters effectively reduces the minimum patch size, so that the filter estimation problem can be underdetermined, and still produce useful PEFs.

Example

A cube of input data was generated by taking a window from several shot gathers, reversing it, and adding it to itself to produce data with both seismic wavefield character and a complex set of conflicting dips.

The top half Figure 1 shows the data after every other trace was zeroed, and then re-estimated with smoothly varying PEFs in very small patches, and independent PEFs in patches just large enough to determine the filter coefficients. The top left is the interpolation result using independent PEFs, the top right is the result using smoothly varying PEFs. The lower half of the figure shows the differences between the original data and the interpolation results. The difference for the independent PEFs result (bottom left) is much greater than the difference for the smoothly varying PEFs result. The independent PEFs result is not bad, but it does not have the accuracy of the smoothly varying result.

Radial Smoothing

In the previous example, the smoothing applied to filters was (approximately) isotropic. Clapp et al. (1998) show how to control the direction of smoothing. In certain cases, it may make sense to specify some preferred direction of filter smoothing. For instance, CMP gathers tend to have approximately constant dip spectra at constant values of x/t, which correspond to radial lines. So it makes sense to arrange patches and smooth filter coefficients in the radial direction on a CMP gather, to accelerate convergence and get good results with as small a number of coefficients as possible. This arrangement has the pleasing result that the largest patches fall at far offsets, where data tends to be the most linear because events are at their asymptotes. Figure 2 shows randomly scattered points smoothed with radial-steering filters. The next data example uses this technique.

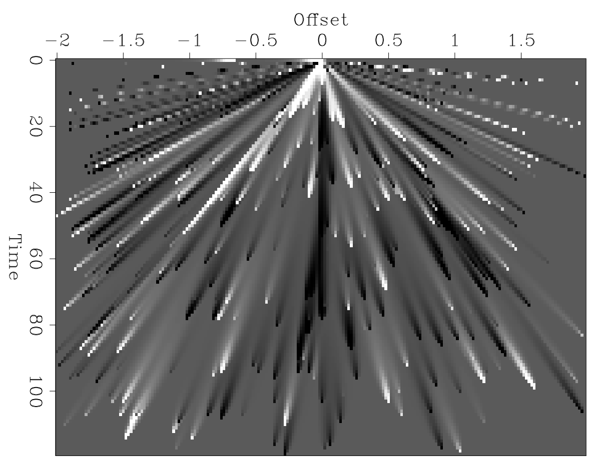

Figure 2: Radial smoothing. Panel shows result of smoothing random scattering of dots with the adjoint radial steering operator. The forward operator points out from the origin.

Noisy Data

It is appealing to think of interpolating land data, because land data can be so expensive to acquire. Land data is more difficult to interpolate than marine data, however, because it tends to be much more noisy. We can guess that the PEFs will find the predictable part of the data, and that that will be the signal. However, even assuming we do estimate the correct PEF (one that captures the dips of the signal rather than the noise), energy from the noise will be carried along those dips to nearby interpolated traces. As an alternative, we can attempt to separate the signal from the noise while interpolating the missing traces. Taking the theory from Claerbout (1997), data space can be decomposed into known plus missing parts, and also into signal plus noise. Given predictors for the signal and the noise, we can attempt to separate signal and noise, and prevent noise energy from creating artifacts in the interpolated traces. For this to work, we need a signal predictor and a noise predictor. For the signal predictor we use the PEFs estimated from the data. In the example to follow we will throw out alternating midpoint gathers and attempt to interpolate them back, so we define noise as whatever is incoherent across midpoints and choose the noise predictor to be an average in a small window along the midpoint axis.

Example

The top left panel of Figure 3 shows a CMP gather from a land seismic survey. There is a visible “noise cone” defined by some low velocity. Inside the cone there is very little coherency along the midpoint axis (not shown). As an experiment with noisy data interpolation, we reduced the data to every second CMP, then performed NMO and stack to produce the top right panel. Then we further reduced the data to every fourth CMP and interpolated with and without noise estimation, and again did NMO and stack to try to reproduce the stack in the top right panel. The bottom left panel, the result without noise estimation, has significant problems with reflector continuity in the stack, while the bottom right, the result with noise estimation, is a much more coherent stack.

CONCLUSIONS

We describe a method for interpolation missing data with smoothly varying PEFs. Forcing the PEFs to vary smoothly enables to us to use very small patches, even a single data sample, and produces good interpolation results on complicated data. We also describe a method for separating signal from noise and interpolating just the signal.

REFERENCES

Abma, R., and Claerbout, J., 1995, Lateral prediction for noise attenuation by t-x and f-x technique s: Geophysics, **60**, no. 6, 1887–1896.

Claerbout, J. F., 1992, Earth Soundings Analysis: Processing Versus Inversion: Blackwell Scientific Publications.

Claerbout, J. F., 1997, Geophysical exploration mapping: Environmental soundings image enhancement: Stanford Exploration Project.

Claerbout, J., 1998a, Factorization of cross spectra: SEP–**97**, 337–342.

Claerbout, J. F., 1998b, Multi-dimensional recursive filtering via the helix: Geophysics, **63**, no. 5, 1532–1541.

Clapp, R. G., 1998, Regularizing velocity estimation using geologic dip information: 68th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1851–1854.

Crawley, S., 1998, Shot interpolation for radon multiple suppression: 68th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1238–1241.

Fomel, S., Clapp, R., and Claerbout, J., 1997, Missing data interpolation by recursive filter preconditioning: SEP–**95**, 15–25.

Spitz, S., 1991, Seismic trace interpolation in the f-x domain: Geophysics, **56**, no. 6, 785–794.

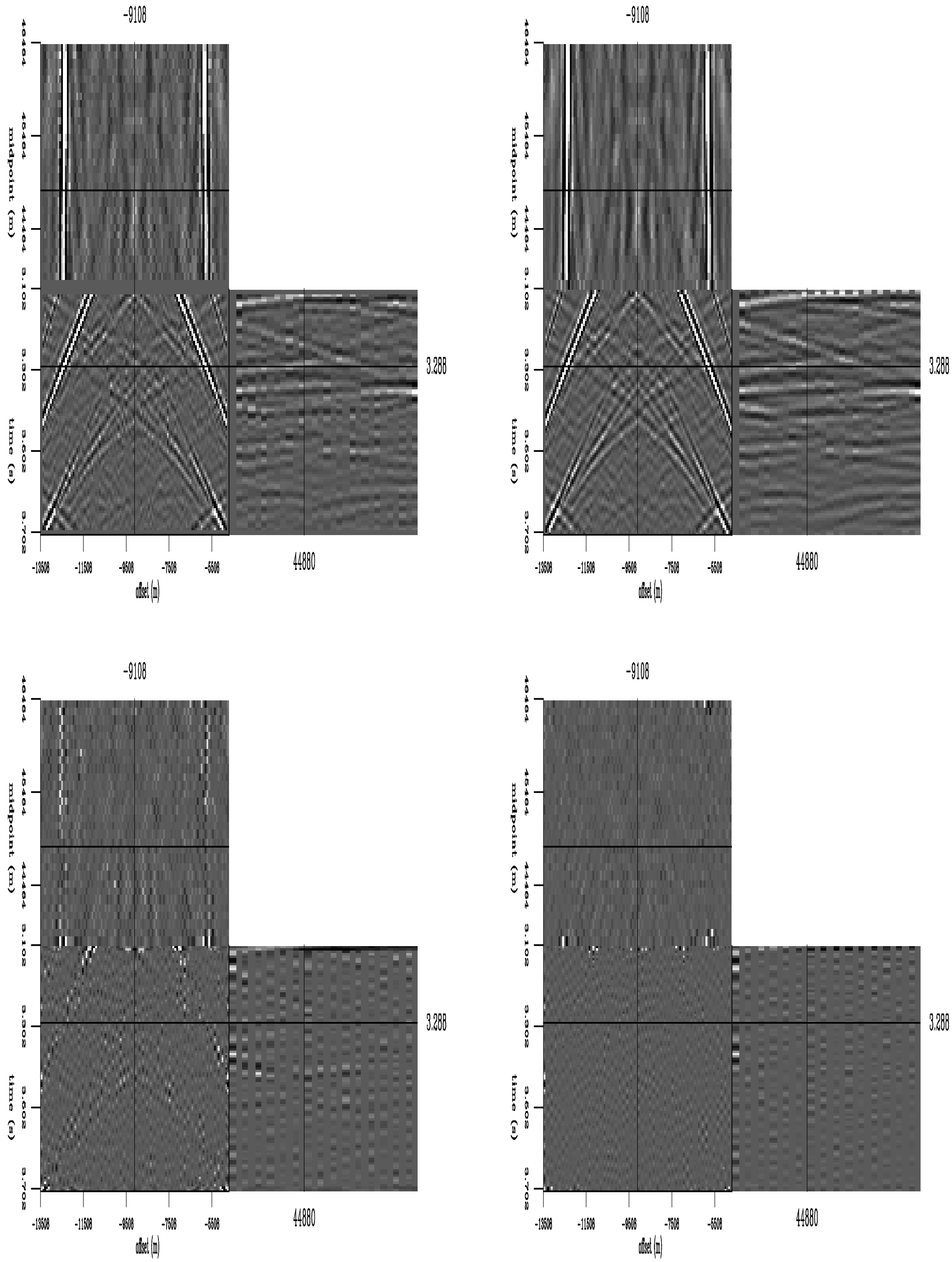

Figure 1: Interpolation results. Top left panel shows data interpolated with independent PEFs, bottom left panel shows difference between top left and original data. Top right panel shows data interpolated with smoothly varying PEFs, bottom right shows difference between top right and original data.

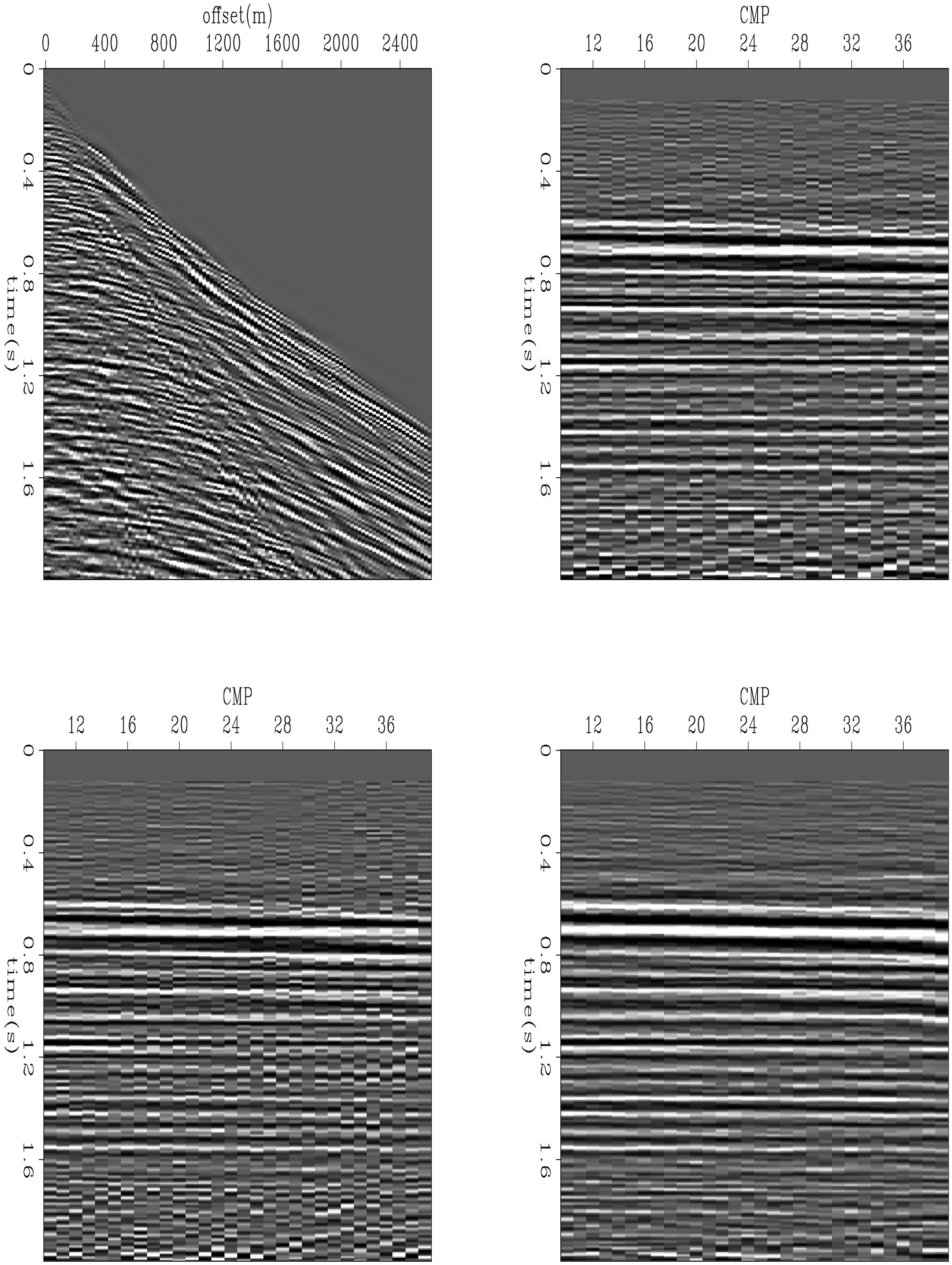

Figure 3: Interpolation of noisy data. Top left shows an example CMP gather from the original data, top right shows part of a CMP stack of the original data. Bottom left shows a stack of the same data after zeroing half the CMP gathers and reinterpolating them without attempting to separate the noise from the signal, bottom right shows a stack after zeroing half the gathers and reinterpolating them while also attempting to separate the signal from the noise.