Data processing

Refraction statics are normally run on all lines in a survey, and brute stacks with refraction statics are checked for line ties. If refraction statics did their job, all lines will tie properly. Velocity analyses should be run after refraction statics corrections. NMO corrections are then applied to the data, followed by surface-consistent statics, NMO refinements, residual statics, and CDP stack. Velocity analyses run after refraction statics will be more consistent throughout an area than previous analyses. This improvement is expected because near-surface anomalies affect apparent structure and stacking velocity.

Data examples

Figure 3a shows a line crossing the South Timbalier trench, south of Louisiana. It is part of a 27-line grid positioned directly over the trench. Seismic data from this area are generally poor because fill material absorbs much of the energy needed for good reflections, and velocities in the fill are low compared to surrounding material, causing time delays in transmitted and reflected raypaths. The first problem can be alleviated by using large air gun arrays, high air pressure, and recording with a drag cable. The second problem must be treated as a long period static.

Figure 3b shows the same line processed with refraction statics. Referring back to the statics profile on Figure 2, one can see that the maximum applied static was -250 ms. Note the improved continuity below the trench. Anomalous "breaks" have been removed, and even the deep data look better. Figure 4a is from a line in the South Pass area, offshore Louisiana. Variable accumulations of mud and gas on the sea bottom cause stacked sections to have areas with little or no interpretable data. Figure 4b shows the same line with refraction statics processing. There is a dramatic improvement on the left side between 1.0 and 2.5 s, and structure conforms to known geology in the area.

Conclusions

Refraction statics properly solved the static problems shown by the two examples in this paper, and helped produce reliable sections showing correct structure. In the first example, refraction statics solved a long period problem with a magnitude much larger than reflection techniques alone could handle. In the second example, statics in an area with little usable reflection energy were solved and data were stacked successfully.

Acknowledgments

We would like to thank Professional Geophysics, Inc. for permission to publish this paper and use their data examples.

FXDECON and Complex Wiener Prediction Filter — POS 2.10

Necati Gulunay, Petty-Ray Geophysical

The complex one-step-ahead prediction filter for use in reducing random noise in stacked seismic data was demonstrated by Canales (1984). The examples shown by Canales demonstrated results equivalent to poststack coherency filtering in improvement, but possibly without the usual ill effects such as loss of frequency content, lateral mixing, smearing across faults, etc. Implementation of the method described by Canales was vague until incorporating information from an article by Treitel (1974) describing the complex Wiener filter. The combination of Canales' idea and Treitel's theory produced the implementation of *f-x* decon described in this paper. FXDECON described herein was tested on synthetic data for algorithm validity and then applied to real seismic data. It proved effective in removing spurious data, ground roll, random noise and even diffractions from stacked data. There are two outputs from the process: a "signal" section with the noise removed, and a "noise" section which shows exactly what was removed. If the noise section and the signal section are recombined, the result is the input section. The main points in the implementation will be tied to Treitel's theory and a limiting case presented. Real data cases will also be presented.

---

f-x domain

A group of traces can be considered to represent a physical phenomenon in the *t-x* domain, where t is the time and x is the horizontal distance (offset) of each trace from the first one. When the Fourier transform of each trace is done, the resultant complex values are said to represent the *f-x* domain. That is, now there is a value at each frequency f and offset x.

Predictability

A tutorial review of linear prediction was given by Makhoul (1975). Here it suffices to say that linear predictability by a one-step-ahead prediction filter means that for a given input of length N:

$$u_1 \; u_2 \; u_3 \; . \; . \; . \; u_N,$$

there must exist a filter of length M

$$f_1 \; f_2 \; f_3 \; . \; . \; . \; f_M,$$

in such a way that any value of the input can be expressed in terms of its past M values

$$u_{k+1} = f_1 u_k + f_2 u_{k-1} + \; . \; . \; . \; + f_M u_{k-M+1}.$$

That is, there must exist a convolutional operator f_t in such a way that

$$u_t * f_t = u_{t+1},$$

where u_{t+1} is a one sample advanced version of u_t and (*) indicates convolution. In other words, there must exist a filter in such a way that,

$$(1, \; -f_1, \; -f_2 \; . \; . \; ., \; -f_M) * (u_1 \; u_2 \; . \; . \; . \; u_N) = (u_1, 0, 0, \; . \; . \; ., 0).$$

Here u_1 is the first sample of the input and represents the irreducible part of input. The time series $f_1 \; f_2 \; . \; . \; . \; f_M$ is called the prediction operator and $(1, \; -f_1, \; -f_2, \; . \; . \; . \; -f_M)$ is called the prediction error operator (Robinson and Treitel, 1980). In terms of z-transforms, predictability by a one-step-ahead prediction filter means that there must exist a non zero filter $F(z)$ such that

$$[1 - zF(z)] \; U(z) = u_1,$$

where $U(z)$ is the z transform of the input and u_1 is the first term of the input. That is, the prediction error filter $1 - zF(z)$ reduces the input to its first sample in a similar fashion as spike decon, and the prediction filter $F(z)$ produces a one sample advanced version of input $U(z)$:

$$F(z) \; U(z) = z^{-1} \; [U(z) - u_1].$$

Complex sinusoids are predictable by a one-step-ahead complex prediction filter

The fact that the complex time series

$$1, \cos(k\Delta x) + j\sin(k\Delta x), \cos(2k\Delta x) + j\sin(2k\Delta x), \ldots, \cos(nk\Delta x) + j\sin(nk\Delta x),$$

or

$$1, e^{jk\Delta x}, e^{j2k\Delta x}, \ldots, e^{jnk\Delta x},$$

is predictable by the prediction filter

$$\cos(k\Delta x) + j\sin(k\Delta x) = e^{jk\Delta x},$$

can be proven using the polynomial equality

$$1 + x + x^2 + \ldots + x^n = \frac{1 - x^{n+1}}{1 - x},$$

with

$$x = z\, e^{jk\Delta x}.$$

Sloping spike train

A spike series of constant amplitude and slope is given by

$$t = I + sx$$

in the t-x domain, where I is the intercept and s is the slope. The f-x domain response of this event is

$$e^{j2\pi fI} \cdot e^{j2\pi fsx}.$$

For a given frequency f, this response is a complex sinosoid Ae^{jkx}, where $A = e^{j21\pi fI}$ and $x = 0, \Delta x, 2\Delta x, \ldots$, and $k = 2\pi fs$. Therefore, it is predictable by $e^{j2\pi fs \cdot \Delta x}$.

Spike trains with conflicting slopes

Shown here are spike trains with conflicting dips need infinite operators, and therefore are not practically predictable by one-step-ahead prediction filter. This is intuitively obvious since two spikes are sloping in separate directions. Mathematically, let x-transforms of 2 different sloping events be given by

$$A_1(z) = 1 + ze^{jk_1\Delta x} + z^2 e^{jk_1 2\Delta x} + \ldots,$$

$$A_2(z) = 1 + ze^{jk_2\Delta x} + z^2 e^{jk_2 2\Delta x} + \ldots.$$

We know from the above discussion that

$$A_1(z) = \frac{1}{1 - ze^{jk_1\Delta x}},$$

and

$$A_2(z) = \frac{1}{1 - ze^{jk_2\Delta x}},$$

i.e., $1 - ze^{jk_1\Delta x}$ is the inverse for $A_1(z)$ and $1 - ze^{jk_2\Delta x}$ is the inverse for $A_2(z)$. The question is, What is the inverse of $A_1(z) + A_2(z)$? Since,

$$A_1(z) + A_2(z) = \frac{1}{1 - ze^{jk_1\Delta x}} + \frac{1}{1 - ze^{jk_2\Delta x}}$$

$$= 2\,\frac{1 - az}{1 - 2az + bz^2},$$

where

$$a = \frac{e^{jk_1\Delta x} + e^{jk_2\Delta x}}{2} \text{ and } b = e^{j(k_1 + k_2)\Delta x},$$

the inverse to $A_1(z) + A_2(z)$ would be

$$\frac{1}{A_1(z) + A_2(z)} = \frac{1}{2}\frac{1 - 2az + bz^2}{1 - az} = \frac{1}{2}\frac{(1 - az)^2 + bz^2 - a^2z^2}{1 - az}$$

$$= \frac{1}{2}\left[(1 - az) + (b - a^2)\frac{z^2}{1 - az}\right].$$

Unless $b = a^2$, this inverse contains infinitely many terms. $b = a^2$ means

$$e^{j(k_1 + k_2)\Delta x} = \left(\frac{e^{jk_1\Delta x} + e^{jk_2\Delta x}}{2}\right)^2.$$

This relation can only be achieved if $k_1 = k_2$. In other words, unless slopes are the same, there is no finite inverse. This means that f-x decon will not work for conflicting dips.

Complex Wiener filter

To summarize Treitel's (1974) paper since it applies to our topic, first it must be remembered that input, output, and filter all consist of complex numbers. Let us define the following:

Input: $x_1\, x_2 \ldots x_{N_x}$
Filter: $f_1\, f_2 \ldots f_{N_f}$
Desired output: $d_1\, d_2 \ldots d_{N_d}$ $\quad N_d = N_x + N_f - 1.$

Then the complex output $0_1\, 0_2 \ldots 0_{N_d}$ is obtained by a complex matrix multiplication

$$0 = \underset{\sim}{\mathbf{X}} \cdot \mathbf{f}$$

$(N_d \times 1)\ (N_d \times N_f)\ (N_f \times 1)$, where $\underset{\sim}{\mathbf{X}}$ is called the convolutional matrix which is made up of the input array $x_1\, x_2 \ldots x_{N_x}$ as described by Treitel (1974). For example if $N_f = 3$ and $N_x = 5$, then we would have

$$\underset{\sim}{\mathbf{X}} = \begin{bmatrix} x_1 & 0 & 0 \\ x_2 & x_1 & 0 \\ x_3 & x_2 & x_1 \\ x_4 & x_3 & x_2 \\ x_5 & x_4 & x_3 \\ 0 & x_5 & x_4 \\ 0 & 0 & x_5 \end{bmatrix}.$$

Let $\underset{\sim}{\mathbf{X}}^{TC}$ represent the transpose conjugate of $\underset{\sim}{\mathbf{X}}$. Then the Hermitian autocorrelation matrix $\underset{\sim}{\mathbf{R}}$ is defined as

$$\underset{\sim}{\mathbf{R}} = \underset{\sim}{\mathbf{X}}^{TC} \cdot \underset{\sim}{\mathbf{X}}$$

$$(N_f x N_f) \qquad (N_f x N_d) \qquad (N_d x N_f)$$

The complex crosscorrelation array $\mathbf{g}$, between the desired output $\mathbf{d}$ and the input $\mathbf{x}$ is given by matrix multiplication:

$$\mathbf{g} = \underset{\sim}{\mathbf{X}}^{TC} \cdot \mathbf{d}$$

$$(N_f x1) \qquad (N_f x N_d) \qquad (N_d x1)$$

If $\mathbf{x_g}$ represents the real part of $\mathbf{g}$ and $\mathbf{y_g}$ represents the imaginary part of it, then real part $\mathbf{x}_f$ and imaginary part $\mathbf{y}_f$ of complex filter $\mathbf{f}$ are the solution of $2N_f$ real equations:

$$\underset{\sim}{\mathbf{M}} \cdot \begin{bmatrix} x_f \\ y_f \end{bmatrix} = \begin{bmatrix} x_g \\ y_g \end{bmatrix}$$

$$(2N_f x 2N_f) \qquad (2N_f x1) \qquad (2N_f x1)$$

Where $2N_f x 2N_f$ real matrix $\underset{\sim}{\mathbf{M}}$ is given in terms of real part P and imaginary part Q of Hermitian matrix $\underset{\sim}{\mathbf{R}}$ as

$$\underset{\sim}{\mathbf{M}} = \begin{bmatrix} P & -Q \\ Q & P \end{bmatrix}.$$

A technique such as Gauss-Jordan method could be used to solve these real equations for $2N_f$ unknowns x_f and y_f.

Complex Wiener filter as deconvolution operator in *f-x* domain

The *f-x* domain is described above and the complex Wiener filter for a given complex time series is summarized. For a given frequency f, the *f-x* response in x direction becomes input to the complex Wiener filter. The main question is how to choose the desired output. We know that for a one-step-ahead prediction operator it must be a one sample advanced version of the input. That is, if the input is

$$x_1\, x_2\, x_3 \;.\;.\;.\; x_{N_x},$$

then the desired output must be

$$x_2\, x_3\, x_4 \;.\;.\;.\; x_{N_x}, \quad 0, \;.\;.\;.\; 0,$$

$$1\; 2\; 3 \;.\;.\;.\; N_x - 1 \quad N_x \;.\;.\;.\; N_d.$$

One of the main criteria in any processing is that the process does not harm the data if the data do not have what the process is looking for. For example, if the data have no statics, a statics program should find no statics. In our case the main criteria should be that if the data has no noise FXDECON should find no noise. The above desired output is found to create noise for pure signal *unless* output energy is piecewise normalized to the energy level of the input. The permanent cure to this problem is to keep an extra trace x_{N_x+1}, and use

$$x_2\, x_3 \;.\;.\;.\; x_{N_x} \quad x_{N_x} + 1 \quad 0, \;.\;.\;.\; 0,$$

$$1 \quad 2 \qquad N_x - 1 \quad N_x \;.\;.\;.\; N_d,$$

as desired output.

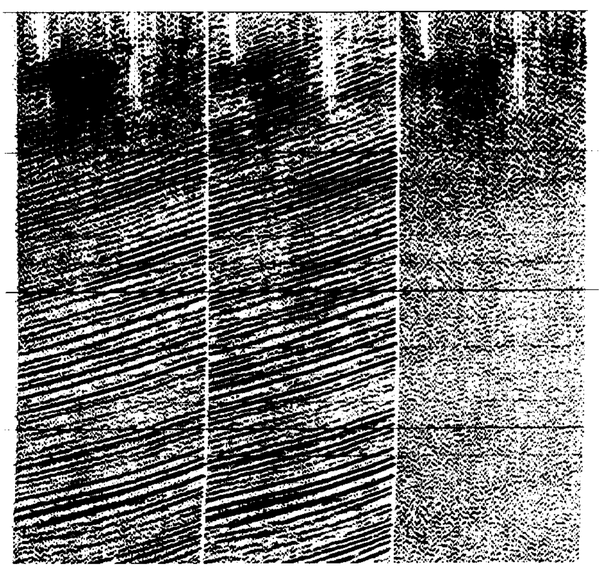

FIG. 1. Input, signal output, and noise output to FXDECON.

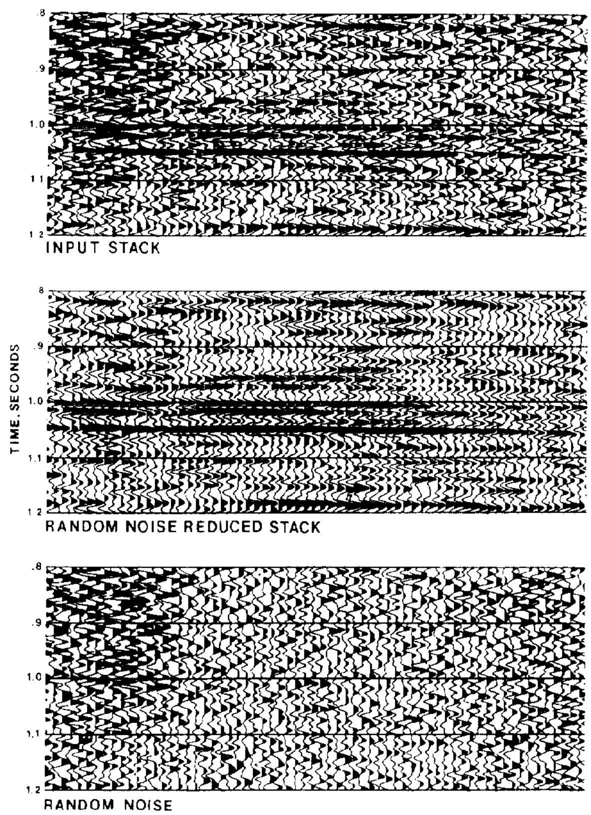

FIG. 2. Input and signal, noise outputs of FXDECON.

Data examples

Two examples of FXDECON are presented here. Figure 1 shows an input and two outputs from FXDECON. The main output is the signal output and the auxiliary output is the noise output. That is, signal plus noise sums to input for every time sample. Shallow and deep noise content show that the method works well in removing random noise. Comparison of signal to input shows that event continuity is enhanced without damaging the waveforms. Figure 2 shows a close up view from a different line. Again, significant noise is found and removed by FXDECON. Coherency and detail obtained after noise removal could be stratigraphically important.

Acknowledgments

I wish to thank my colleague, Dana Stanley, for carefully designing and implementing the production portions of the program and our center manager, Jerry Jennings, for bringing the process to my attention.

References

Canales, L. L., 1984, Random noise reduction: Presented at the 54th Annual SEG Meeting, Atlanta.

Makhoul, J., 1975, Linear prediction: A tutorial review: Proc. IEEE, **63**, 561.

Robinson, E. A., and Treitel, S., 1980, Geophysical signal analysis: Prentice Hall, Inc., 27, 270–272.

Treitel, S., 1974, The complex Wiener filter: Geophysics, **39**, 169–173.