AAPG Datapages/Search and Discovery Article #90174 CSPG©2014 CSPG/CSEG/CWLS GeoConvention 2012, (Vision) May 14-18, 2012, Calgary, AB, Canada

White Noise Suppression in the Time Domain – Part I

Patrick Butler, GEDCO, Calgary, Alberta, Canada

pbutler@GEDCO.com

Summary

Two modules for removing white noise from seismic data will be presented. They employ principal component decomposition in the time domain (Jones, 1985). Because all frequencies move together in a spatial sense, this domain offers a considerable advantage in processing the low amplitudes at the top end of the wavelet's spectrum. It will be shown, with the aid of cross correlations, how one can recover most of the signal respecting curvature and statics in space down to about 30 dB with only a few basis functions (which are the elements of the principal component decomposition). The original amplitudes are preserved and the noise is removed without mixing. Finally, examples will illustrate that this process really does work.

In short if you respect that seismic data "wobbles" in space and changes amplitude, then the seismic signal may be extracted relatively easily.

Introduction

This topic has received a lot of attention recently in the guise of trace interpolation (Lui et al., 2004). However that work was done principally in the frequency domain. That has proven to be more difficult than necessary.

The frequency domain has three major disadvantages. First the signal energy falls off at higher frequencies so dip determination there becomes problematic. Of course the lower frequencies can be used to guide the higher but that introduces further problems. The second problem is that these higher frequencies tend to alias. The third and more systemic problem is that the basis functions are linear. Seismic data, even when restricted to small windows, is curved and more important it "wobbles" in both amplitude and statics. This scatters energy into basis functions that have little to do with the primary signal. We assume that the signal is smooth and continuous (premigration) but is this ever really true?

It was decided to rework the problem in the time domain. This domain is handy because all the frequencies move together in space unless we have dispersion. The latter we can deal with by operating in relatively small (say 21 traces by 300 ms) overlapping windows.

Theory and/or Method

It is important to realize that white noise is random and hence unpredictable. Our job therefore is to attack and remove the signal which is recognized because within small windows the traces "look" the same. They move up or down dip, change amplitude and jitter up and down (because of statics) but still "look" the same. Conflicting dips may be removed with successive passes.

Figure 1 below shows how to build a basis function (which is a trace that "looks" like the data). The principal dip is identified by stacking along all possible dips. The stack traces are raised to the 4^{th} power and summed to identify the dip that gives the sharpest stack. Because the data is only approximately linear this stack is low frequency. So we now cross correlate each trace with the stack, static it and restack to yield a high frequency basis function. Note that if some high frequencies are lost at this point, they will never be recovered. Typically the cross correlations are limited to about ±25 ms but more has been used without seeing leg skipping effects. Though it is not entirely apparent on this plot, the basis with cross correlations has an amplitude spectrum that matches the data spectrum even out to 80 Hz and 30 dB down. More evidence of this is available in Part II.

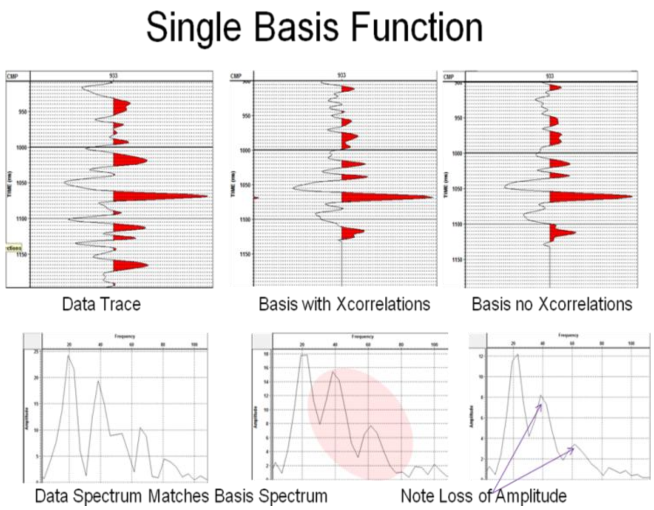

Figure 1: Single Basis Function

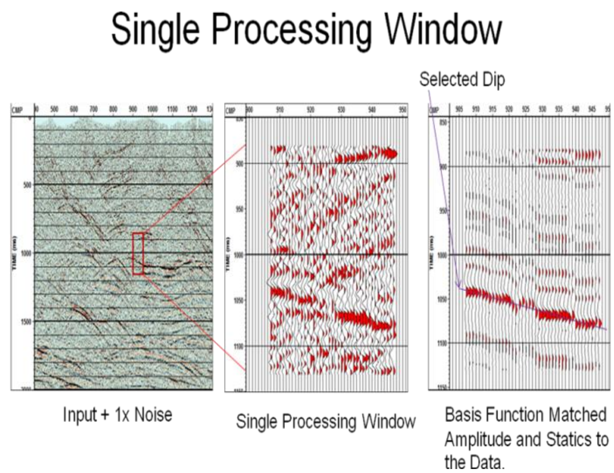

Figure 2: Basis function for a single processing window

If you were to take a constant wavelet and add white noise, then the result would be a sharper wavelet than the original. (The noise may be richer than the wavelet in high frequencies.) For data in which the noise is the same amplitude as the signal we do not see this effect, but when the noise is say seven times the signal then the effect is visible. This implies that even though all frequencies move together, as the noise increases, there is a point where the high frequencies are being lost. see Part II.

Also this process depends on reducing the noise through stack. If some of the noise is strong and not white it will overwhelm the stack and force the program to use the first pass to remove it. In practice, a different algorithm (based on median filtering) removes this strong noise before we run the algorithm described here.

Now that a high frequency basis function has been found, it can be matched in amplitude and statics to each trace in turn – as shown in Figure 2. It is then subtracted from the original data. Note that if the original event truncates abruptly (say after migration) then the amplitude match will zero the basis contribution just as abruptly. There is no smearing of the data and the output will match the input signal amplitude.

These windows are tapered and overlapped to cover the whole section. After one dip has been removed this process can be repeated until the visible signal is gone. In practice three passes are usually sufficient. When the signal is gone this residual is subtracted from the original to get the enhanced section.

Examples

To test the algorithm a stack containing some noise is combined with white noise band limited to the signal frequency range. The RMS amplitude of the noise is equal to that of the original data – as shown in Figure 3.

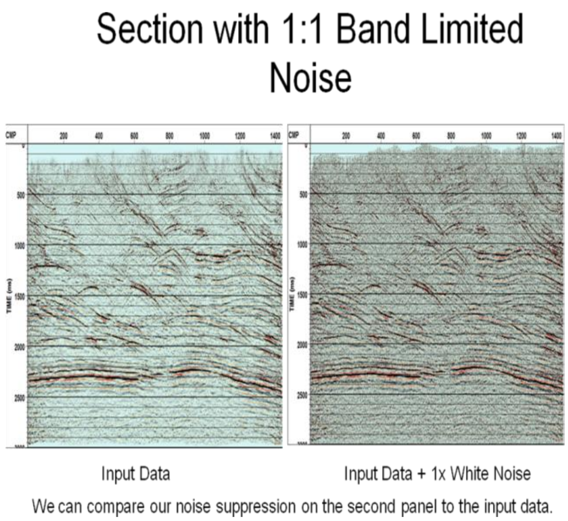

Figure 3: Input test data

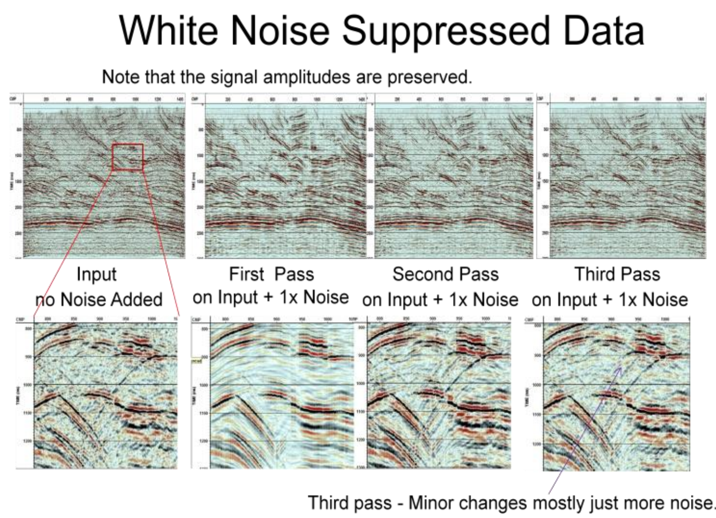

Figure 4: Result of algorithm on input data

In Figure 4 the results of processing the input data + 1x white noise are shown. The left most panel is the original input data. A window of 21 traces was used. (4.6x noise reduction). Note that in using 3 basis functions a simultaneous solution should be used, but in practice successive subtraction is good.

The first pass picks up only one dip. The second pass has two dips and creates character changes on the first dip. The third pass gets character changes on the secondary dips and any remaining third dips. In most parts of the section, however, it is just adding more noise.

In Figure 5 the events are examined in detail. The event truncations are not smeared by even one trace. Event amplitude and character are preserved. The output is not identical to the original input but this is mostly because the original input contains some noise.

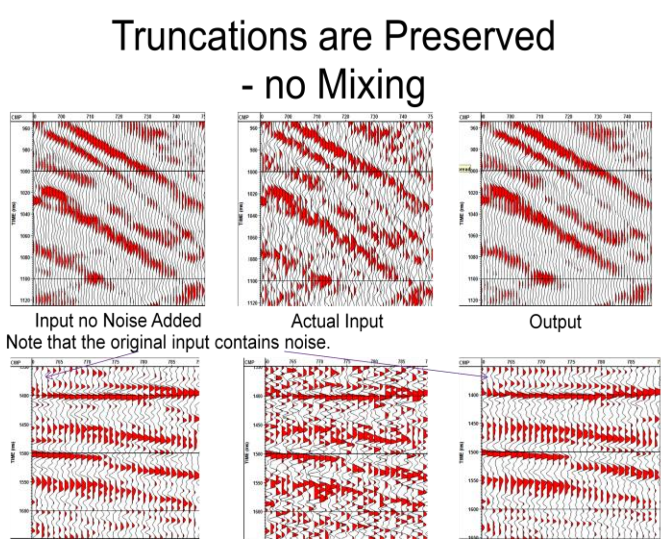

Figure 5: zoom of Input data and Output data

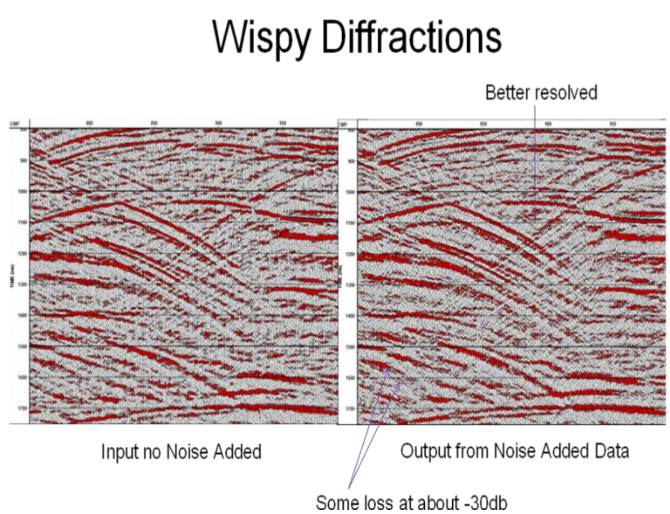

Figure 6: effect of algorithm on diffracted energy

In Figure 6 we examine the effect of the algorithm on diffracted energy – particularly when that energy is very low. There is some loss on the wispy diffractions about 30 dB down. However some has been gained too. This limit is dependent on the square root of the number of traces used to create a basis function. The data may or may not support longer windows.

The amplitude changes as events cross are shown in Figure 7. These must be preserved if the data is to migrate properly.

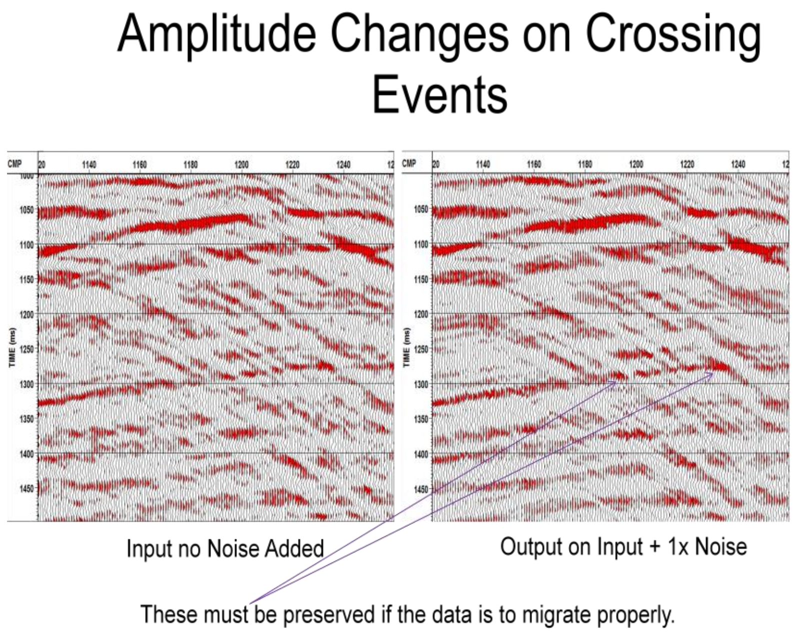

Figure 7: zoom data in Figure 6

Conclusions

The signal has been modeled with a small number (up to three) principal components in the time domain. The frequency content, amplitudes and fine structure of our original signal has been preserved in the presence of a moderate amount of noise down to a limit of about 30 dB. In short if you respect that seismic data "wobbles" in space and time and changes in amplitude, then the seismic signal may be extracted relatively easily.

Acknowledgements

Thanks to GEDCO for allowing me to present this algorithm.

References

Jones, I,F., 1985, Thesis, Applications of the Karhuhunen-Loeve transform in reflection seismology: University of British Columbia.

Lui, B., Sacchii, M.D., 2004, Minimum weighted norm interpolation of seismic records: Geophysics, **69**,1560-1568.