Signal/Noise Decomposition

J.F. Claerbout, W. Harlan, R. Ottolini, J. Toldi, and R. Ullmann

I. Use of Migration for Signal/Noise Decomposition. *Jon F. Claerbout*

When an impulse is diffracted it produces a hyperbola. Many impulses produce many hyperbolas. Let the (y,t)-plane have an impulse at every point. The strength and polarity of each impulse is independently drawn from a Gaussian distribution. You get a lot of hyperbolas. They overlap everywhere. It has never been proven, but the *cognizenti* say that among so many hyperbolas, individuals are no longer distinguishable. The hyperbolas don't look like hyperbolas any more. You can't see their velocity. What you see looks just like more independent Gaussian random variables. This is a quirk of the Gaussian probability distribution. With a "high tail" distribution, you should be able to recognize individual hyperbolas.

We commonly observe hyperbolas in reflection data. Likewise with migrated reflection seismic data we commonly see many semicircles. Indeed, before-stack migration processing seems overwhelmed by semicircles. This leads to the idea that geophysical signals and noises seem to be more spikey than Gaussian. By this idea, signals are spikey in the depth domain, and noises are spikey in the time domain. Certainly missing traces and cable truncations are more spikey along the horizontal space axis in the time domain than would be their representation in the depth domain.

In this section, a specific model is formulated for field data with these attributes, and some clumsy methods are proposed for decomposing it into signal and noise parts. In the following sections others propose their chosen methods of decomposition using this first data model as a starting point. Readers are invited to suggest improvements on all the proposed methods.

A Scatterer Model for Signals and Noises

Define a stretching function

$$Stretch^{\alpha}(x) \quad = \quad sgn(x)\,|x|^{\alpha} \quad = \quad x\,|x|^{\alpha-1} \tag{1}$$

Define a signal model as a stretched Gaussian random function in midpoint-depth space.

$$Z(y,z) \quad = \quad Stretch^{\alpha}[Gaussian(y,z)] \tag{2}$$

Likewise, define a noise model as a stretched Gaussian random function in midpoint-traveltime space.

$$T(y,t) \quad = \quad Stretch^{\beta}[Gaussian(y,t)] \tag{3}$$

Observational data $D(y,t)$ is modeled by

$$D(y,t) \quad = \quad T(y,t) + Diffract[Z(y,z)] \tag{4}$$

The parameters α and β are greater than unity, but otherwise are unknown. The problem is this: Given $D(y,t)$ and some good programs for migration and diffraction, how do you decompose to find T and Z ?

The model embodies many aspects of reality, but in some ways it is artificial. A realistic model for Z would involve some spatial correlation. This is omitted because it is so complicated to characterize the spatial correlation properties of realistic geological models. Naturally we will try to avoid techniques which would involve this artificial aspect of our model.

Tools

Obviously, good migration and diffraction programs are presumed to be available for use. They may be used iteratively, but in practice only about three iterations would be done. Because of evanescent energy, we may assume that diffraction will undo migration, but not the converse.

A basic estimation tool is low-pass filtering. In today's application this is likely to be on the y-axis with a tridiagonal matrix, say

$$\underset{y_0}{LoPass} \quad = \quad \frac{1}{1 - y_0^2\,\delta_{yy}} \tag{5}$$

$$\underset{y_0}{HiPass} \quad = \quad \frac{-\,y_0^2\,\delta_{yy}}{1 - y_0^2\,\delta_{yy}} \tag{6}$$

The parameter y_0 characterizes the cutoff frequency of the filters.

Automatic gain is defined as any dynamic range compression technique. Often it is smoothing the envelope, followed by raising to a fractional power. In today's application it may be useful to define a parameter y_1 as the amount of smoothing along the y-axis.

$$\underset{y_1}{Gain^{\alpha}} \; D(y,t) \quad = \quad \frac{D(y,t)}{<|D|>_{y_1}^{\alpha}} \tag{7}$$

Inverse automatic gain is defined only after automatic gain has already been done and the gain divisor has been stored away for later use.

$$G^{-\alpha} \, [\; G^{\alpha}[D(y,t)] \;] \quad = \quad D(y,t) \tag{8}$$

A possible hint is provided by some data processing philosophies which say that you should always transform to Gaussian before averaging. It is like the idea that you should never average numbers which are suspected to be out of scale with one another, because then the average will be biased by the averaging technique. Low pass filtering is averaging. This philosophy seems to say that you should always AGC before lowpass filtering, then un-AGC afterwards. Ordinarily, a high-pass filter is just one minus a low-pass filter. This philosophy seems to say that a proper high-pass (PHP) operator should also be done after AGC

$$PHP \quad = \quad 1 - \frac{1}{G} \, LowPass \; G \quad = \quad \frac{1}{G} \, (1 - Lowpass) \; G \quad = \quad \frac{1}{G} \, HiPass \; G \tag{9}$$

The Crudest Decomposition

A crude guess is that the flat top of a hyperbola or the flat bottom of a semicircle can be discriminated against by means of low frequency k_y rejection in the appropriate space. Specifically

$$\hat{T}(y,t) \quad = \quad \underset{y}{HiPass} \; D(y,t) \tag{10a}$$

$$\hat{Z}(y,z) \quad = \quad \underset{y}{HiPass} \; D(y,z) \tag{10b}$$

A problem with (10) is that it makes no attempt to satisfy the constraint $D = T + Z$. Some of D is copied into both T and Z and other components of D may not go into either.

Next Guess

The next guess attempts to prevent energy in D from ending both in T and Z. Obviously, T, Z, and D can be expressed in either (y,t)-space or in (y,z)-space. In an effort to avoid clutter so that key ideas stand out, we will no longer explicitly indicate migration or diffraction. That will be deducible from context. For the next guess, you iterate with

$$\hat{T} \quad = \quad PHP(D - \hat{Z}) \tag{11a}$$

$$\hat{Z} \quad = \quad PHP(D - \hat{T}) \tag{11b}$$

This solves the problem of energy of D ending out in both $\hat{T}$ and $\hat{Z}$. The problem remains that low k_y-frequencies don't end out either in $\hat{T}$ or $\hat{Z}$. Thus (11b) seems to be able to capture the texture on a horizontal bed, but not the bed itself. Sufficiently dipping beds should be captured nicely by (11b). We can get the data constraint (4) to be fulfilled if after iteration of (11a,b) we append one final step

$$\hat{Z} \quad = \quad D - \hat{T} \tag{11c}$$

II. An Iterative Method of Removing Noise *Ron Ullmann*

Below is a possible solution to the problem of how to process seismic data to separate the signal and from the noise. This process uses a great deal of computation, so some short cuts must be made. The assumptions for this method are that the signal is an extended Gaussian function in the depth domain, the noise is an extended Gaussian function in the time domain, and the data is the sum of the noise and the diffracted signal component. The extended Gaussian function means the seismic data consists mainly of noise spikes and hyperbolas which are diffracted from the signal spikes. The purpose of any processing is to separate the noise spikes from the hyperbolas.

The first step is to start migrating down the time axis, but stopping part way down the axis. The migration will start to collapse the hyperbolas into point sources and cause the noise spikes to broaden and form semi-circles. The migration is stopped when the noise spikes have developed into short flat events on the section. By running a proper high-pass filter along the y-axis, the flat events from the noise spikes and the top of the hyperbolas will be attenuated. Vertical events, such as the arms of the hyperbolas, will not be affected. Next, pass this section through a diffraction program to return it to its original state. The noise that was causing the semi-circles is gone and the signal still exists in the arms of the hyperbolas. This assumes that the energy in the arms of a hyperbola will not diffract into a shape other than a hyperbola. This results in a severe attenuation of the noise and a less severe attenuation of the signal. Subtract this section from the original section, which has the effect of removing part of the signal without removing the noise.

Repeat the above process on the data iteratively, but migrate further down the time axis before applying the high-pass filter. After this process is repeated several times, the final section will consists only of noise. Apply automatic gain to the noise section so that noise level is near the level of the original data and subtract it from the original data. The final result is a section in which the noise is attenuated.

The important property of this process is how it affects the reflections from flat, horizontal beds. The migration program will not affect the flat reflections. When the high-pass filter is applied, the reflections will be removed along with the noise. When this is subtracted from the original data, the noise and the flat beds will remain unchanged. After this process is done, the only thing left when the noise section is subtracted form the original data will be the signal spikes that caused the hyperbolas. To correct for this, a final proper high-pass filter should be applied to the noise section before it is subtracted from the original data. The filter will eliminate the flat reflections before they are subtracted from the original data.

III. Migrating for Signal/Noise decomposition. *John Toldi*

The observational data $D(y,t)$ consist of two parts: the noise, modelled as a series of large, randomly distributed (in y and t) spikes and the signal, modelled as a series of large amplitude hyperbolas (the diffracted $y - z$ spikes). The spikiness present in the models comes about through the application of a stretch function to gaussian random functions. The stretch function reduces the amplitude of values less than one, while amplifying values greater than one. The net effect is to produce a distribution which is richer in small values than a gaussian distribution, and hence spikier. The goal of the process I am describing is to separate the signal from the noise.

The first phase is to crudely eliminate the bulk of the signal, through the application of a high-pass y-axis filter. To properly precondition the data, we need to first apply Automatic Gain Control (AGC) along the y direction. At this point we would want to use a short smoothing operator in determining the envelope, since otherwise the large noise spikes would still be way out of scale with too many of their neighbors. Now high-pass filter the data along y to eliminate the flat tops of the hyperbolas, and hence much of the signal energy. Even though we have filtered out some of the low frequencies of the noise spikes, the large inverse AGC for those points will somewhat boost them back up, relative to the hyperbolas. The tails of the hyperbolas will still be present, since they appear to be rich in high frequencies in y. With inverse AGC, this phase becomes:

$$E(y,t) = \frac{1}{G}[Highpass_y(D(y,t)]\ G \tag{1}$$

The second phase begins by migrating the filtered data. The noise spikes will produce semi-circles, while the tails of the hyperbolas will focus back into depth-domain spikes. Only the evanescent energy will no be properly re-focussed. Now AGC along the y-axis, followed by a low-pass filter. This sequence will filter much of what remains of the signal spikes, while leaving the bottoms of the semicircles produced by the noise spikes. Of course, some of the noise spike energy residing in the semi-circle tails will be lost, but much of the energy is concentrated near the bottom. Now inverse AGC, then diffract to get back to the time domain:

$$F(y,z) = \frac{1}{G}[Lowpass_y(migrated\ [E(y,t)])]\ G \tag{2}$$

$$H(y,t) = Diffract(F(y,z)) \tag{3}$$

Finally, we identify:

$$\hat{T}(y,t) = H(y,t) = noise \tag{4}$$

$$\hat{Z}(y,t) = D(y,t) - H(y,t) \tag{5}$$

and

$$\hat{Z}(y,z) = migrate\ [\hat{Z}(y,t)] = depth-signal \tag{6}$$

IV. Separation by filtering amplitudes. *William Harlan*

Let us use a filter which extracts locally strong, "spikey" amplitudes and rejects regions of low, uniform amplitudes. Such a filter should reject hyperbolas or semicircles because of their unfocussed low amplitudes. If hyperbolas or semicircles dominate a particular section, then focussed, spikey energy will dominate after diffraction or migration. By beginning the decomposition in the proper domain, we can always see that the spikey information stands out.

Let us not define this filter in terms of frequencies present. We cannot expect spikey peaks to contain frequencies with *magnitudes* very different from those of hyperbolas or semicircles. Phases do differ substantially, but in a difficult way to grasp in the frequency domain. A dip filter, which relies on phases, could discriminate some hyperbolic energy from focussed energy, but a very poor percentage of it. But we already have a perfect hyperbola filter--migration. We only need recognize a single high amplitude point in depth to know the energy and location of a hyperbola in time. Filter amplitudes then, not frequencies. The essence of our filter will be that focused energy, spikey peaks, will be relatively high in amplitude, whereas unfocussed energy, hyperbolas and semicircles, will not.

Now define the "amplitude filter." This filter will operate directly on individual samples of the data, but as a function of the gained analytic envelope we store previously. Samples with high envelope values, down to a certain cutoff, should be retained at full strength; lesser samples should be attenuated as a smooth function of the envelope values. Such a filter is

$$C'_i = \frac{C_i}{1 + \left[\frac{A_0}{Envlp(C_i)} \right]^6}$$

C_i is the sample, and A_0 is the cutoff amplitude of the envelope. By operating as a function of the gained analytic envelope, the filter will not attempt to deepen troughs between high peaks, a change which would greatly alter the frequency content. The preserved high amplitude portions of the data should resemble the same regions of the original gained waveforms as much as possible.

We are almost ready to describe an iterative decomposition of the data. First, we require migration and diffraction subroutines which preserve energy: the sum of the squares of amplitudes should remain the same after transformation (except for boundary losses). Then, if one domain contains greater amplitudes than the other, we know that it contains more focussed energy. Begin the separation of Z(y,z) and T(y,z) in the section with the highest amplitudes, for the highest peaks are certain to contain focussed energy; the other domain may be dominated by strong hyperbolas or semicircles. Assume for demonstration (but not in general) that the migrated section has the highest amplitudes.

Now for the loop. Begin with the migrated section of the original ungained data. Apply the amplitude filter, with a relatively high cutoff, so that we keep only the highest-amplitude, best-focussed peaks. Diffract this section and subtract from the original data. Now we should expect the noisy peaks to stand out because the strongest hyperbolas are gone. Extract the spikey peaks corresponding to noise from this section by the same procedure used on the depth section. Migrate the extracted peaks and subtract from the original migrated data. Amplitude filter this section, but with a lower amplitude cutoff than before because the strongest semicircles are gone and cannot interfere. Continue the cycle at least once more and end in the depth domain, after removing the migrated noise.

Though this algorithm worked very well with synthetic sections, it failed with real data because of the density of geologic events. When high amplitude geologic events were extracted, most of the energy of the overlying semicircles were extracted too, thereafter impossible to identify as noise. We need an additional linear transformation to make the geological events more parsimonious, to reduce the strong lateral predictability of these events. In this way we can reduce the overlapping of signal and noise. A slant stack serves the purpose very well as we saw in the preceding article.

V. DIPXOR Noise Removal. *Rick Ottolini*

The DIPXOR process removes noise spikes from an unmigrated seismic section. The name of this process derives from that it is a combination of dip filtering and exclusive or-ing of seismic sections. This process requires no migration or diffraction during the course of noise removal.

This process makes use of the fact that reflections and noise have different characteristics in dip space. On an unmigrated seismic section, reflection events appear as line segments, hyperbolas, or some combination thereof. Noise is defined as spatially and temporally uncorrelated spikes. Any local portion of a reflection has a relatively narrow dip range. A noise spike consists of essentially all dips.

The DIPXOR process begins by decomposing the unmigrated seismic section into several seismic sections, each representing a different narrow range of dips. The dip slices cover all dips between -90 and +90 degrees. The dip decomposition is linear, so the original section is returned by summing all of the dip sections.

The second part of the process is to *selectively* sum together the dip sections to obtain the unmigrated section free of noise spikes. The summation algorithm is that if a sample exceeds a certain strength on more that one dip section, it is not summed into the result. In digital logic this called exclusive or-ing. Since noise spikes would appear on most of the dip sections, then they would be eliminated by this selective summation. Of course, the selective summation would have to take in account the spill over of reflector energy into neighboring dip sections and the possibility of intersecting reflectors of different dips.

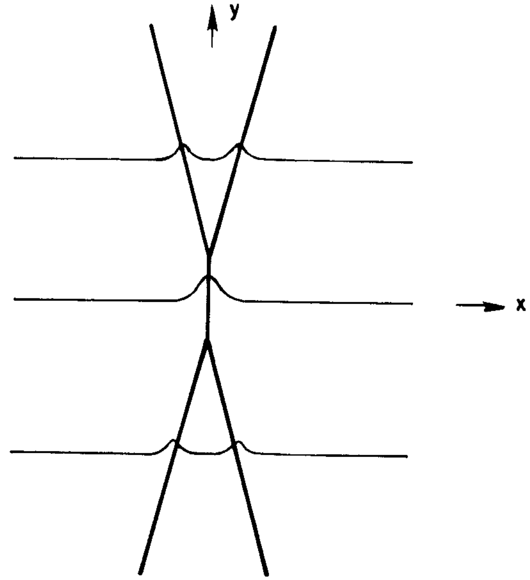

FIG. 4.7a. *Sketch of two-soliton solution of Kddomtsev-Petviashvili equation from* (3.3.91). *In this symmetric case,* $k_1 = k_2$, $p_1 = -p_2 > \sqrt{3}k_1$. *This pattern moves in the x-direction with speed* $(k_1^2 + p_1^2)$.

sheet of water lies, and restrict our attention to the symmetric modes (so that (4.1.6) holds). Now we are considering waves on a freely suspended sheet of water, such as that studied by Taylor (1959). The main advantage of this configuration is that viscous effects are much less important.

(v) If $\hat{T} > \frac{1}{3}$ (dominant surface tension), Zakharov and Manakov (1979) have solved (4.1.23) exactly by a generalization of IST. They require boundary conditions restrictive enough that lumps are excluded a priori. The asymptotic ($t \to \infty$) behavior of the solution with these boundary conditions was given by Manakov, Santini and Takhtadzhyan (1980).

4.1.b. Internal waves. The internal oscillations due to gravity of a stably stratified fluid are known as internal waves. Both the oceans and the atmosphere usually are stratified, and they support rich spectra of these waves (e.g., Phillips (1977)). In fact, the waves at the air-water interface that we have just discussed may be thought of as an extreme case of internal waves (caused by an extremely large density gradient at the interface).

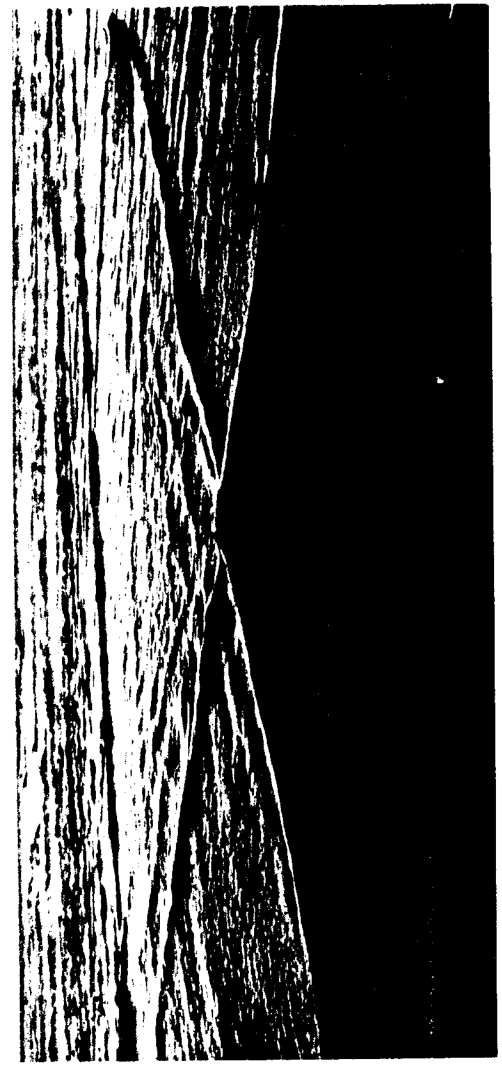

FIG. 4.7b. *Oblique interaction of two shallow water waves. (Photograph courtesy of T. Toedtemeier.)*