VARIABLE NORM DECONVOLUTION

William Gray

Variable Norm Deconvolution is a generalization of Wiggin's Minimum Entropy Deconvolution. It is an iterative multichannel method which tries to achieve a deconvolution with a small number of events and least residual error. It is shown to be related to minimizing the entropy of the inputs. An efficient implementation in the frequency domain is presented. It is applied to synthetics and a real data set. Results are subjectively equivalent to those achieved with other methods of deconvolution.

Introduction

The recorded seismic trace can be represented by a source wavelet convolved with a reflectivity series plus additive noise. Deconvolution is a process which estimates a reflectivity series from the recorded trace.

Currently variants of predictive or homomorphic deconvolution are used to estimate the reflectivity series when the source wavelet is unknown. Predictive deconvolution assumes a white reflectivity series which is uncorrelated with the noise. It estimates a source wavelet and then finds an inverse which is convolved with the recorded data to yield a reflectivity series. This method assumes the source wavelet is minimum phase, an assumption that in many situations is incorrect.

Homomorphic deconvolution assumes the source wavelet and reflectivity series are separable in the cepstral domain. To be separable the source wavelet must be slowly varying relative to the reflectivity series. This method has problems when the noise level is high or the source has high cepstral components.

Recently Ralph Wiggins [1] developed Minimum Entropy Deconvolution. It differs from previous methods because no assumptions except ergodicity are made about the source wavelet. MED attempts to account for the observed data in terms of a small number of large events and with the smallest residual error. There is an important trade off here that is effected by the density of events and the variance of their amplitudes.

Claerbout [2], realized this and consequently introduced parsimonious deconvolution in SEP 13. Intuitively he felt that MED is excessively biased towards the larger events and a method which "sees" more of the data would result in a better deconvolution. Claerbout uses the concept of parsimony to define a better deconvolution.

The law of parsimony relates to economy of description. When loosely applied to deconvolution it requires that no more events than necessary be allowed to account for the recorded data. One deconvolution is said to be more parsimonious than another when it contains fewer events and has the same residual error. A paper by Ferguson [3] gives an excellent

discussion of parsimony and entropy in factor analysis.

Here a variable norm deconvolution is introduced. It is a generalization of MED. Where MED maximizes the varimax norm of the reflectivity series averaged over the channels,

$$V = \sum_{i=1}^{nch} \frac{\sum_{j=1}^{nt} x_{ij}^4}{\left(\sum_{j=1}^{nt} x_{ij}^2\right)^2} , \tag{1}$$

the variable norm deconvolution maximizes,

$$U(\alpha) = \sum_{i=1}^{nch} \frac{\sum_{j=1}^{nt} |x_{ij}|^{\alpha}}{\left(\sum_{j=1}^{nt} x_{ij}^2\right)^{\alpha/2}} \tag{2}$$

When $\alpha=4$ the variable norm deconvolution reduces to MED.

The parameter α governs the relative weightings of the amplitudes. Large values of α yield deconvolutions which are biased towards the larger events. Economic considerations can also determine the choice of norm. Large values of $\alpha(\alpha \sim 6)$ require four or five iterations to convergence. Small values $(\alpha \sim 2.2)$ can require ten to forty iterations.

Derivation of the Algorithm

An algorithm to maximize $U(\alpha)$ is derived in the time domain. The derivation closely follows that of Wiggin's for the MED algorithm.

Let y_{ij} be the jth sample of the signal recorded on channel i where there are nt samples per channel and nch channels. Let f_k be the kth sample of an inverse filter having nf samples. The reflectivity series, x_{ij}, is the convolution of the recorded signal and inverse filter,

$$x_{ij} = \sum_{k=1}^{nf} y_{i\ j-k} f_k \quad . \tag{3}$$

The variable norm is defined as

$$U(\alpha) = \sum_{i=1}^{nch} \frac{\sum_{j=1}^{nt} (x_{ij})^{\alpha}}{\left(\sum_{j=1}^{nt} x_{ij}^2 \right)^{\alpha/2}} \quad . \tag{4}$$

It is scale invariant or homogeneous so the gains on different channels do not affect the solution.

To find an inverse filter that gives a relative maximum of $U(\alpha)$, take the partial of $U(\alpha)$ with respect to the filter coefficients. When the partial vector is zero, a relative maximum is found.

$$\frac{\partial U(\alpha)}{\partial f_k} = \sum_{i=1}^{nch} \left[\left(\sum_{j=1}^{nt} x_{ij}^2 \right)^{-\alpha/2} \left(\sum_{j=1}^{nt} \alpha |x_{ij}|^{\alpha-1} \operatorname{sgn}(x_{ij}) \frac{\partial x_{ij}}{\partial f_k} \right) \right.$$

$$\left. - \alpha \left(\sum_{j=1}^{nt} |x_{ij}|^{\alpha} \right) \left(\sum_{j=1}^{nt} x_{ij}^2 \right)^{-\alpha/2-1} \left(\sum_{j=1}^{nt} x_{ij} \frac{\partial x_{ij}}{f_k} \right) \right] \tag{5}$$

From (3),

$$\frac{\partial x_{ij}}{\partial f_k} = y_{i\ j-k} \tag{6}$$

Setting (5) to zero and using (6) gives a non linear expression relating an optimal reflectivity series to the recorded signal.

$$\sum_{i=1}^{nch}\left(\frac{\sum_{j=1}^{nt}|x_{ij}|^{\alpha}}{\left(\sum_{j=1}^{nt}x_{ij}^{2}\right)^{\alpha/2+1}}\right)\sum_{j=1}^{nt}x_{ij}\,y_{i\,j-k}=$$

$$\sum_{i=1}^{nch}\left[\left(\sum_{j=1}^{nt}x_{ij}^{2}\right)^{-\alpha/2}\sum_{j=1}^{nt}|x_{ij}|^{\alpha-1}\,(\operatorname{sgn}x_{ij})\,y_{i\,j-k}\right] \qquad (7)$$

To solve (7) iteratively assume x_{ij} in the sum on the left is defined by (3) and all other occurrences are computed from the previous estimate, then (7) is written,

$$\sum_{\ell=1}^{nf}\left[\sum_{i=1}^{nch}c_i\sum_{j=1}^{nt}y_{i\,j-\ell}\,y_{i\,j-k}\right]f_{\ell}=\sum_{i=1}^{nch}\left[d_i\sum_{j=1}^{nt}|x_{ij}|^{\alpha-1}\operatorname{sgn}(x_{ij})\,y_{i\,j-k}\right] \qquad (8)$$

where

$$c_i=\frac{\sum_{j=1}^{nt}|x_{ij}|^{\alpha}}{\left(\sum_{j=1}^{nt}x_{ij}^{2}\right)^{\alpha/2+1}},$$

and

$$d_i=\frac{1}{\left(\sum_{j=1}^{nt}x_{ij}^{2}\right)^{\alpha/2}}.$$

Using matrix notation (8) can be simply expressed as

$$Rf = g \tag{9}$$

where

R is a weighted sum of the autocorrelation matrices of the inputs,

f is an unknown filter,

and

g is a weighted sum of crosscorrelations between the inputs and the current estimates of the reflectivity series to some power where their signs are retained.

When $\alpha=4$ equation (8) reduces to

$$\sum_{\ell=1}^{nf} \left[\sum_{i=1}^{nch} c_i \sum_{j=1}^{nt} y_{i\ j-\ell}\ y_{i\ j-k} \right] f_\ell = \sum_{i=1}^{nch} d_i \left[\sum_{j=1}^{nt} x_{ij}^3\ y_{i\ j-k} \right] \tag{10}$$

where

$$c_i = \frac{\left(\sum_{j=1}^{nt} x_{ij}^4 \right)}{\left(\sum_{j=1}^{nt} x_{ij}^2 \right)^3} ,$$

and

$$d_i = \frac{1}{\left(\sum_{j=1}^{nt} x_{ij}^2 \right)^2} .$$

This is Wiggin's basic equation for Minimum Entropy Deconvolution.

Frequency Domain Implementation

The algorithm for variable norm deconvolution is implemented in the frequency domain because it is faster. Let capital letters represent the fourier transforms of the corresponding time domain variables. A bar over a variable represents its complex conjugate. The following is a verbal step by step description of a flow chart used to program this algorithm.

Initialization is accomplished by setting the inverse to a unit spike and transforming,

$$f(t) \leftarrow \delta(t-t_0)$$
$$F(\omega) \leftarrow FFT(f(t)) \quad . \tag{11}$$

The shape of the final inverse filter is invarient to the initial time shift t_0 . The length of $f(t)$ should be that of the recorded trace.

Vectors R and G are used to accumulate the weighted auto and cross correlations of the channels. They are equivalent to R and g of (9). At the start of each iteration they are zeroed,

$$R(\omega) \leftarrow 0$$
$$G(\omega) \leftarrow 0 \quad . \tag{12}$$

Each trace, $y(t)$, is read and transformed.

$$Y(\omega) \leftarrow FFT(y(t)) \tag{13}$$

Before transforming the input trace it may be necessary to double its length by zero padding to prevent aliasing due to performing convolution by multiplication in the frequency domain. This is especially important when the inverse filter has a lot of high frequency energy.

The reflectivity series is computed

$$X(\omega) \leftarrow F(\omega) \cdot Y(\omega) \tag{14}$$

and inverse transformed,

$$x(t) \leftarrow \mathrm{IFFT}(X(\omega)) \quad . \tag{15}$$

The scale factors,

$$c \leftarrow \frac{\sum_{j=1}^{nt} |x_j|^{\alpha}}{\left(\sum_j x_j^2\right)^{\alpha/2+1}} \tag{16}$$

and

$$d \leftarrow \frac{1}{\left(\sum_j x_j^2\right)^{\alpha/2}} \tag{17}$$

are found and the current estimate of the reflectivity series weighted,

$$x'(t) \leftarrow |x(t)|^{\alpha-1} \operatorname{sgn}(x(t)) \tag{18}$$

and its transform computed,

$$X'(\omega) \leftarrow \mathrm{FFT}(x'(t)) \quad . \tag{19}$$

The auto and cross correlations are found, then weighted and added into the accumulation vectors,

$$\begin{aligned} R(\omega) &\leftarrow R(\omega) + c\ \overline{Y}(\omega)\ Y(\omega) \\ G(\omega) &\leftarrow G(\omega) + d\ \overline{Y}(\omega)\ X'(\omega) \end{aligned} \tag{20}$$

Steps described by (13) to (20) are performed on each trace. When all have been processed the inverse filter is found by spectral division,

$$F(\omega) \leftarrow \frac{G(\omega)}{(R(\omega) + \lambda \sum_{\omega'} R(\omega'))} \quad . \tag{21}$$

This step is related to solving the Toeplitz system (9) in the time domain. The parameter λ is analogous to the white noise added to the zero lag auto correlation. It is proportional to the inverse of the squared signal to noise ratio of the recorded data.

In practice it was found that setting the d.c. term of $G(\omega)$ to zero improves the stability of the algorithm. Physically this makes sense because source wavelets are zero mean.

The new inverse filter is scaled so it has unit power,

$$F(\omega) \leftarrow \frac{F(\omega)}{(\sum_{\omega'} \overline{F}(\omega') F(\omega'))} \quad . \tag{22}$$

The norms for each channel are then compared with the norms of the previous iteration. Convergence is assumed if the differences are less than some prespecified tolerence. If another iteration is required steps described by (12) to (22) are repeated.

The final inverse is used to deconvolve the input data by the steps described by (13) through (15). The deconvolved output must be time shifted by t_0 of step (11) to correct for the time delay introduced by the inverse filter.

The algorithm outlined above often is unstable because it tries to spike the channel with highest $U(\alpha)$. This is avoided when the contribution of each channel to the auto and cross correlations is made inversely proportional to its norm. Dividing c and d of equations (16) and (17) by the norm gives

$$c = \frac{1}{\sum_{j=1}^{nt} x_j^2} \tag{23}$$

and

$$d = \frac{1}{\sum_{j=1}^{nt} |x_j|^{\alpha}} \tag{24}$$

Using these scale factors simplifies the algorithm as $R(\omega)$ needs to be computed only on the first iteration. This modification causes the algorithm to seek a solution where the norms on all channels are nearly the same.

Properties of the Algorithm

Assuming an infinite time series and a single channel the algorithm yields an inverse filter which spikes the data. This is shown for $\alpha=4$.

Let the initial filter be a delta function at t_0 so that its Fourier transform is

$$F(\omega)=e^{i\omega(t-t_0)} \ . \tag{25}$$

It follows that the initial reflectivity series is

$$X(\omega) = Y(\omega)e^{i\omega(t-t_0)} \ . \tag{26}$$

Raising a time series to the nth power in the time domain is equivalent to convolving it with itself n times in the frequency domain. Neglecting scale factors the inverse filter is written,

$$F(\omega) = \frac{\overline{Y}(\omega)\ [X(\omega) * X(\omega) * X(\omega)]}{\overline{Y}(\omega)\ Y(\omega)} \tag{27}$$

Substituting (26) for $X(\omega)$ in (27) and realizing that exponentials can be factored out of the convolutions gives

$$F(\omega) = \frac{\overline{Y}(\omega)\ Y^{*3}(\omega)\ e^{i\omega(t-t_0)}}{\overline{Y}(\omega)\ Y(\omega)} \tag{28}$$

where $Y^{*3}_{(\omega)}$ represents the transform of the recorded data convolved with itself three times.

After n iterations the reflectivity series is

$$X(\omega) = Y(\omega)\ F(\omega) = Y^{*3n}(\omega)\ e^{i\omega(t-t_0)} \tag{29}$$

or in the time domain after shifting the output by t_0 ,

$$x(t) = y^{3n}(t) \tag{30}$$

As n goes to infinity the biggest amplitude grows faster than the rest of the trace resulting in a single spike.

Trying to extend the results above to the multichannel case was too difficult. The equivalent expression to (28) for the multichannel case is,

$$F(\omega) = \left(\frac{\sum_{i=1}^{nch} d_i\ \overline{Y}_i(\omega)\ Y_i^{*3}(\omega)}{\sum_{i=1}^{nch} c_i\ \overline{Y}_i(\omega)\ Y_i(\omega)} \right) e^{i\omega(t-t_0)} \tag{31}$$

The effects of variable norm deconvolution are surprisingly similar to spectral smoothing. Figure 1 illustrates its smoothing property. The convolutional model for seismic data is

$$y_t = (x * b)_t + n_t \tag{32}$$

where b_t is the source waveform and n_t is additive noise. Using synthetic data each component, $s(t)$, of the model was weighted by

$$s'(t) = |s(t)|^{\alpha} \operatorname{sgn}(s(t)) \quad . \tag{33}$$

The spectrums of the weighted series for different values of α are displayed below the input series. The parameter α controls the amount of smoothing. Increasing α clearly increases the smoothing.

The variable norm algorithm divides a function which has a smoothed spectrum by the average spectrum to directly estimate the inverse wavelet. This is similar to spectral smoothing as it divides the average spectrum of an ensemble by a smoothed version. From the result a minimum phase source wavelet is derived.

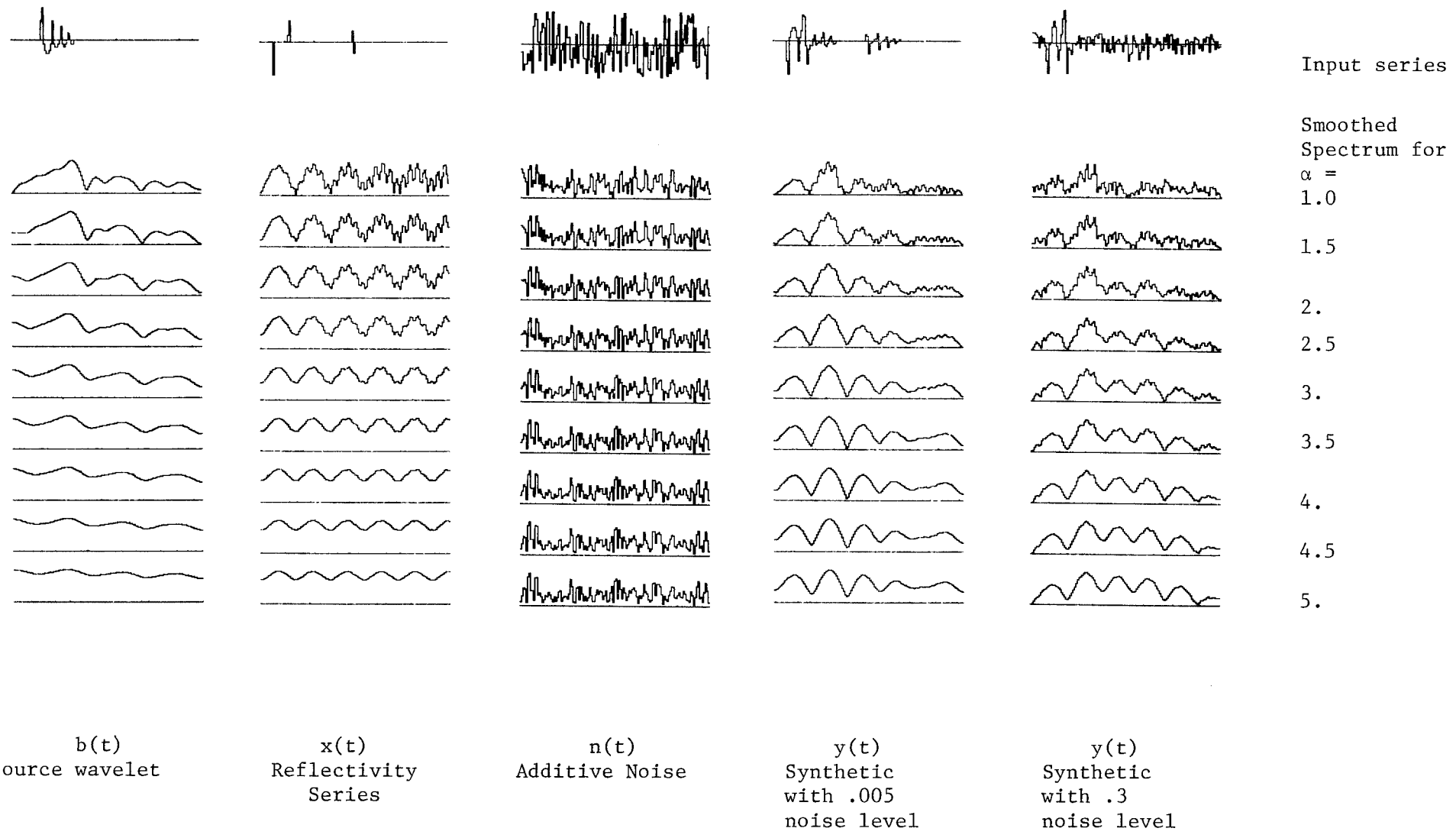


Figure 1 Each column is a component of the convolutional model, $y_t = (x * b)_t + n_t$. At the top are the input series. Below are the spectrums of the input series weighted by $s'_t = |s_t|^{\alpha} \operatorname{sgn}(s_t)$. The parameter α smoothes the spectrum of all components except the noise.

Application to Synthetics

Synthetic data sets are used to show the properties of the variable norm algorithm when the density of events and noise levels are varied.

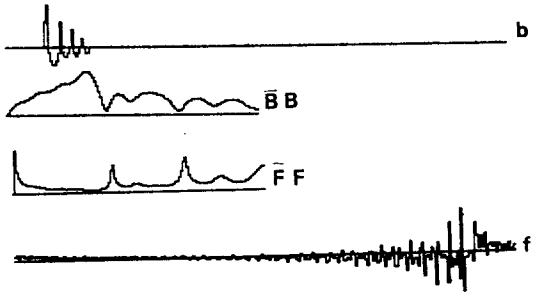


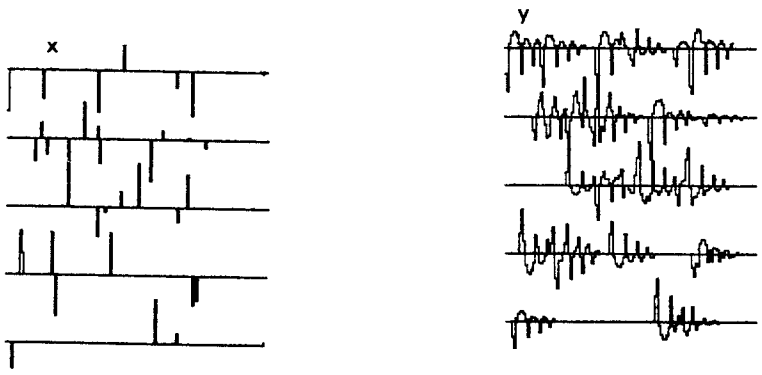
Figure 2 Synthetic wavelet, its spectrum, estimated inverse spectrum, and inverse wavelet.

The same source wavelet is used for all examples. It is displayed in Figure 2 where the inverse transform, $F(\omega)$, was found from

$$F(\omega) = \frac{\overline{B}(\omega)}{\overline{B}(\omega)B(\omega) + (1.\text{x}10^{-8}) \sum_{\omega'} \overline{B(\omega')}B(\omega')} \quad . \tag{34}$$

The inverse wavelet was computed by inverse transforming $F(\omega)$.

The length of the source wavelet is 23 samples, that of the synthetic seismograms is 128 samples. Events were not allowed after sample 105 to eliminate truncation effects. For real data, the end effects can be minimized by using a triangular taper. Convergence of the algorithm was assumed when the norm of the estimate of the reflectivity series of the first trace changed by less than .001 from that of the previous iteration. All examples were initialized by a unit spike in the first sample of the inverse filter. The trace lengths were doubled to eliminate convolutional aliasing.



Reflectivity Series

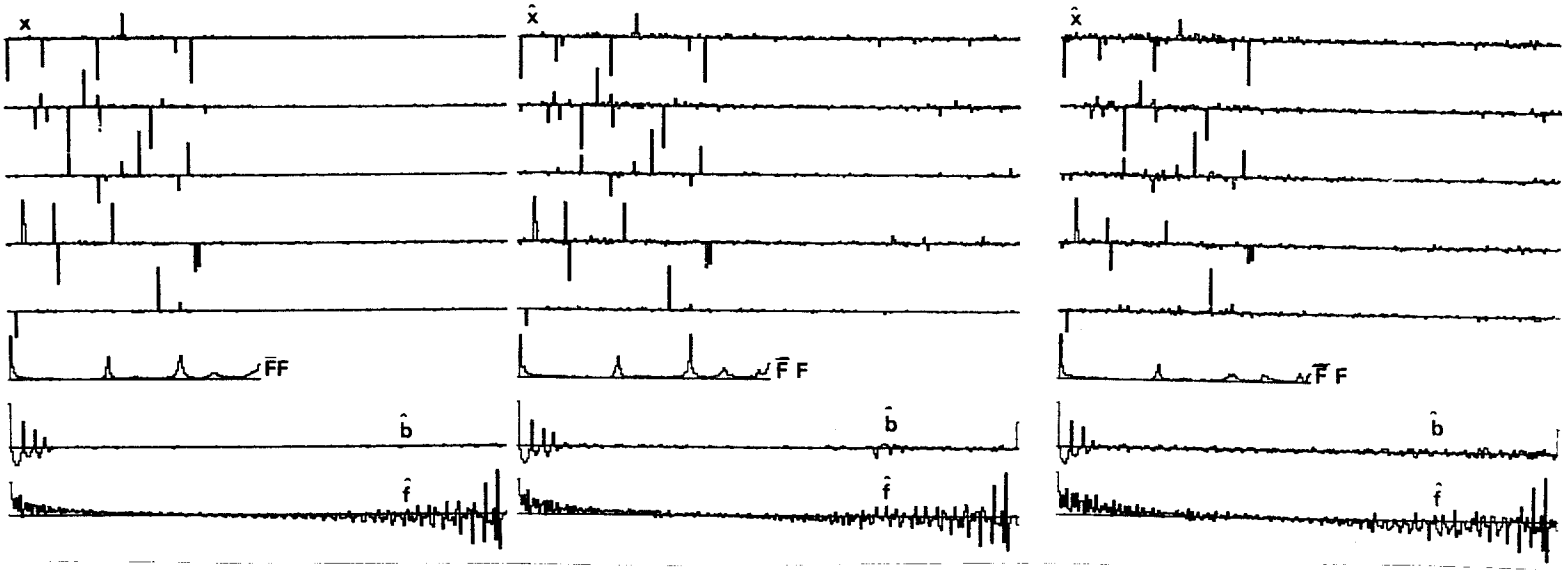
.05 Density

Synthetics

.001 Additive Noise

Figure 3 Synthetics (right) for low density reflectivity series (left).

The first test was practically noise free and had a low density of events. Figure 3 displays the inputs. On the left is the reflectivity series with density (.05) or about one event per twenty samples. Each event has its time and amplitude drawn from a uniform distribution. On the right are the synthetics resulting from convolving the source wavelet (Figure 2) with the reflectivity series and adding noise drawn from a uniform distribution. The noise level is about .001 that of the largest event.



$\alpha = 2.2$	$\alpha = 4.$	$\alpha = 6.$
12 iterations	5 iterations	4 iterations

Figure 4 Results for low density reflectivity series (.05), low noise (.001). Each column displays for a different α the estimated reflectivity series, inverse spectrum, forward wavelet, and inverse wavelet.

The results of the first test are shown in Figure 4 for $\alpha = 2.2$, 4, and 6. The estimates of the source wavelet, inverse spectrum, and inverse filter are displayed below the deconvolutions. The source wavelet was estimated using equation (34) where the roles of the source and inverse are reversed.

For $\alpha = 6$ the inverse filter almost spikes the fifth trace and clutters the others up. This is generally caused by choosing too high a norm. For $\alpha = 4$, the algorithm is still trying to spike the fifth trace but the other traces constrain it so fairly good results are obtained. The best results were achieved with $\alpha = 2.2$. Close comparison with Figures 2 and 3 shows a nearly perfect deconvolution and estimate of the source.

These results imply that setting α small yields the best deconvolution. When $\alpha = 2$ nothing happens as $U(\alpha) = 1$. When $\alpha < 2$

the algorithm tries to equalize the amplitudes on all channels. The results look like the noise displayed in Figure 1. It seems that α just barely greater than 2 would yield the best deconvolution.

This is wrong as the next example (Figure 5) shows. Letting $\alpha = 4$, and 6 were also tried. When $\alpha = 6$ the algorithm spiked one trace. For $\alpha = 4$ the deconvolution was a little noisier than $\alpha = 2.5$ but converged in eight iterations.

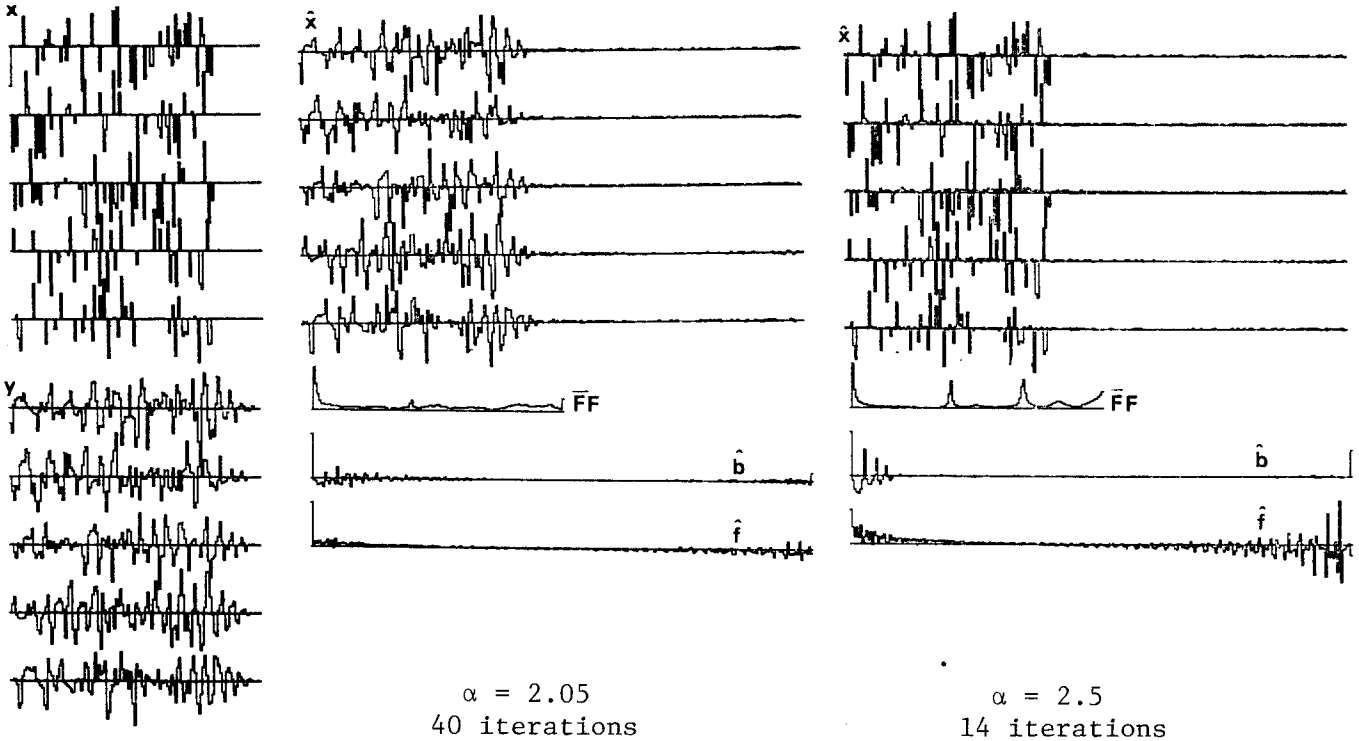


Figure 5 High density (.3) reflectivity series and synthetics on the left. On the right are results for $\alpha = 2.05$ and 2.5. $\alpha = 2.5$ yields a near perfect deconvolution. $\alpha = 2.05$ hardly changes the inputs.

Figure 1 shows that varying α has little effects on the spectrum of noise. Several tests on synthetics with varying levels of additive noise were performed.

Figure 6 shows results from synthetics with an ambient noise level of .1 and reflectivity series density .05. The deconvolution for $\alpha = 2.5$ is better than $\alpha = 4$ for all but the fifth channel. As in Figure 4,

when $\alpha = 4$ the fifth channel is given too high a weight. This is because its norm is about twice as high as any other channel.

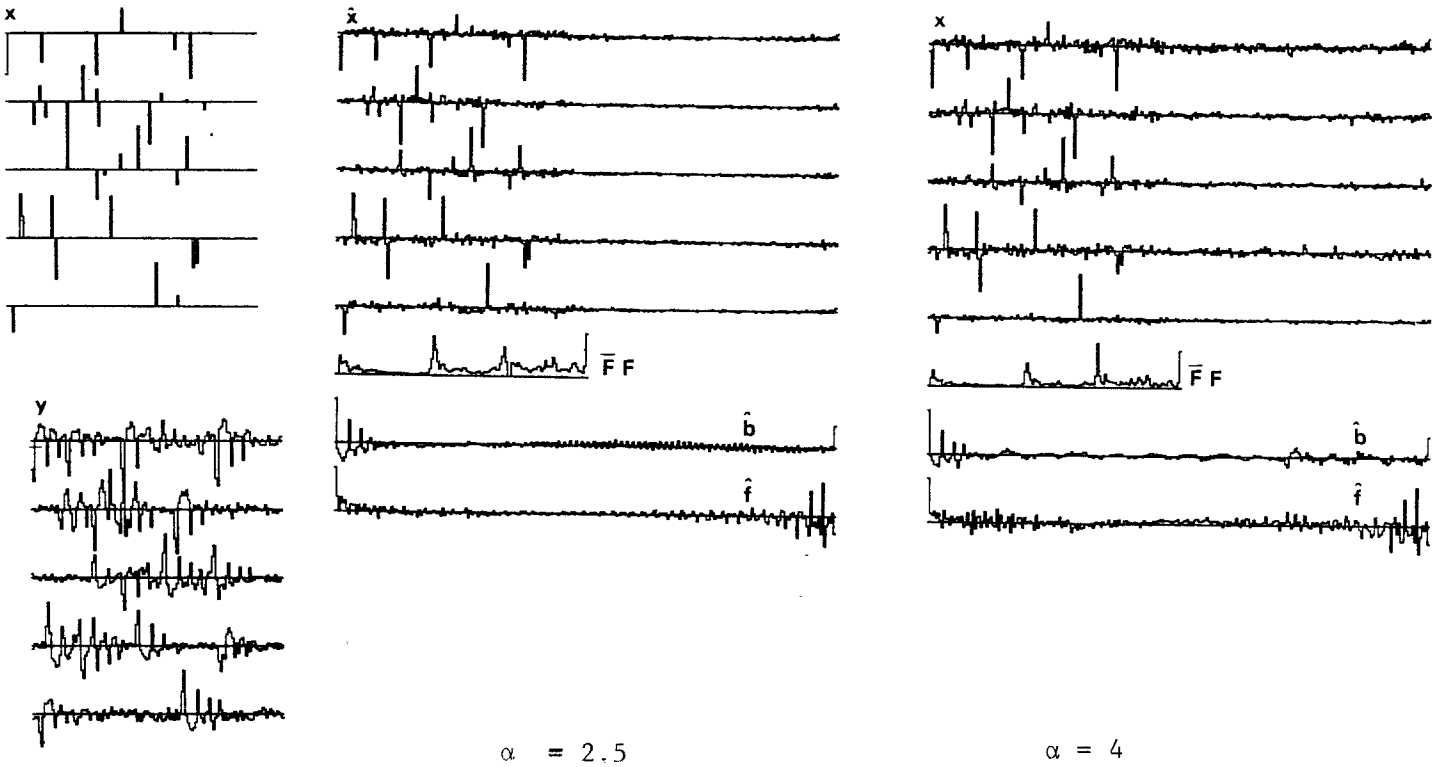


Figure 6 Results for moderate noise level. RMS signal to noise ratio is 3.8, ambient level is .1 .

When the ambient noise level reached about .25 the results started degenerating. Figure 7 shows deconvolutions for the same data as in Figure 6 but with noise levels of .3 and .7 . The inverse spectrums have some similarity to the spectrum of the source wavelet. It seems that at high noise levels the inverse filter that maximizes the norm is the time reversed source wavelet - a matched filter.

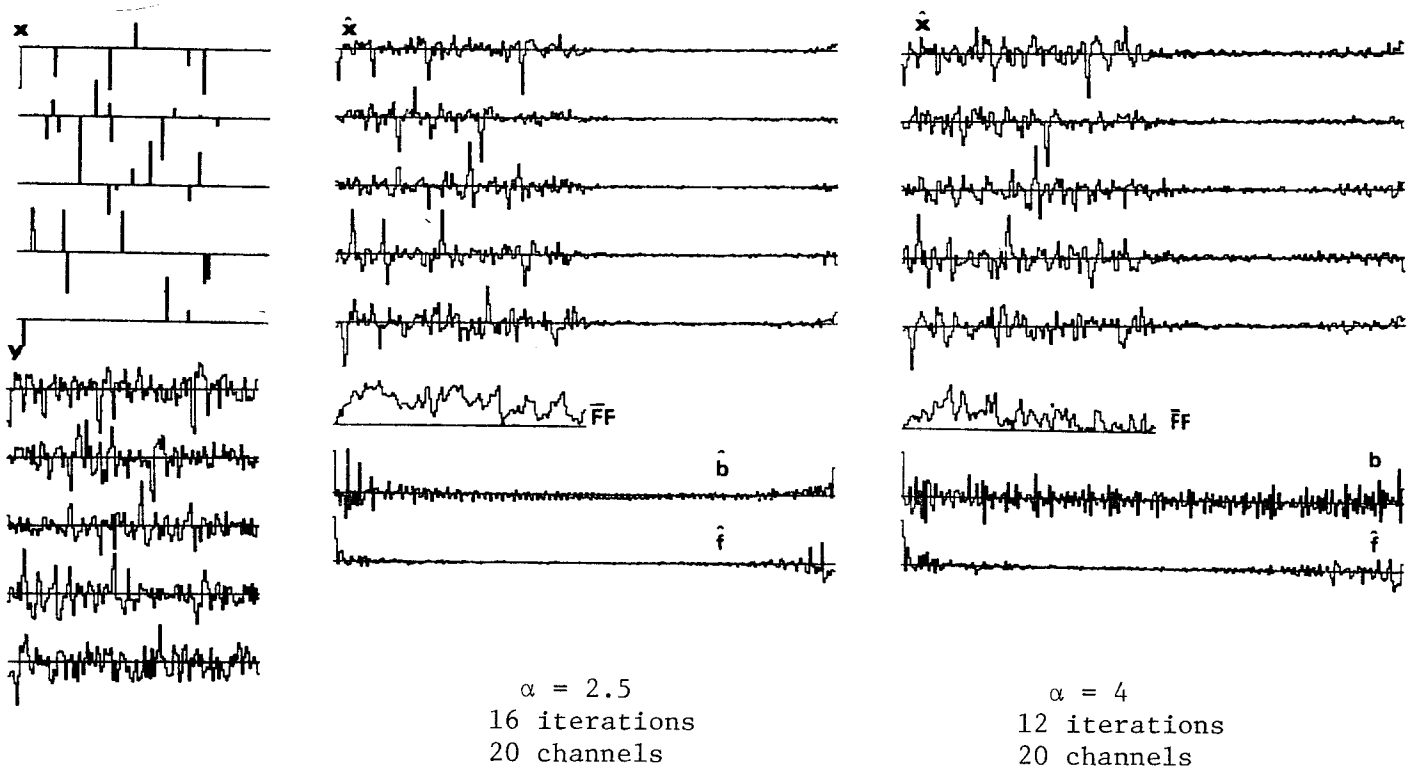


Figure 7 Results for high noise levels. For the $\alpha = 2.5$ deconvolution the ambient noise level was .3 and RMS signal to noise ratio 1.1 . For the $\alpha = 4$ case the ambient noise is .75 and signal to noise ratio .42 . Inputs for the $\alpha = 2.5$ case are displayed on the left.

Application to Field Recorded Data

Marine seismic data recorded in 1975 for the USGS in the Gulf of Alaska was available for testing. The source was twenty-two airguns of 1200 cu. in. size. Six seconds of data were recorded at a sampling rate of 4 ms on each of forty-eight channels. The near trace offset was 200M and the far trace 2550M.

The data was chosen for deconvolution because bubble oscillations of the reflected pressure pulse have noticeable amplitude for 250 ms after the initial pulse. These low frequency oscillations obscure primaries and are difficult to remove with stacking.

Two seconds of data on each of the near sixteen traces were used as inputs to the algorithm. A triangular taper was applied to reduce truncation effects and the d.c. component of each trace removed. Three deconvolutions for α = 2.5, 4, and 6 are presented in Figures 11, 12, and 13. For comparison the original data and deconvolutions achieved with spectral smoothing and parsimony are presented in Figures 10, 14, and 15. Only two seconds of the near twenty-four traces are displayed. The same time varying gain correction was applied to the original data and deconvolutions before plotting.

The inverse spectrums estimated source wavelets and inverse filters are displayed in Figure 9. The effect of increasing α is clearly related to the power present in the higher frequencies of the spectrums. The estimated source wavelets are ringy because of a hole in the inverse spectrum at 60 Hz.

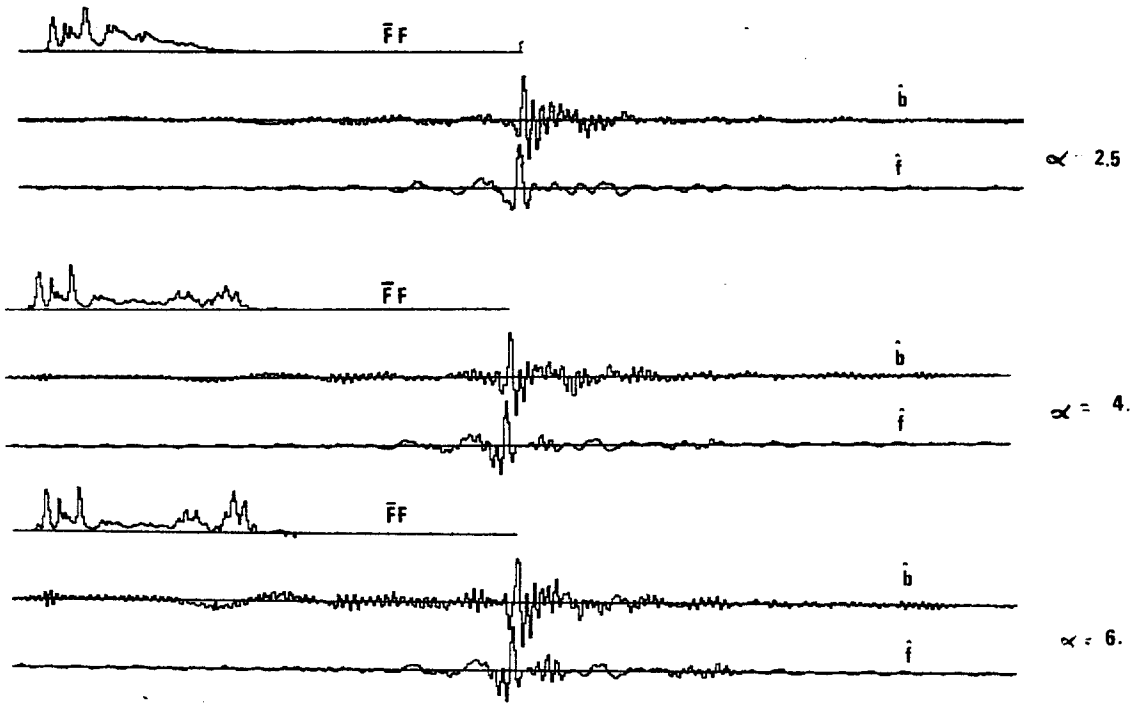


Figure 9 Spectrums of the inverse filters, estimated source wavelets, and inverse filters for variable norm deconvolutions on field data.

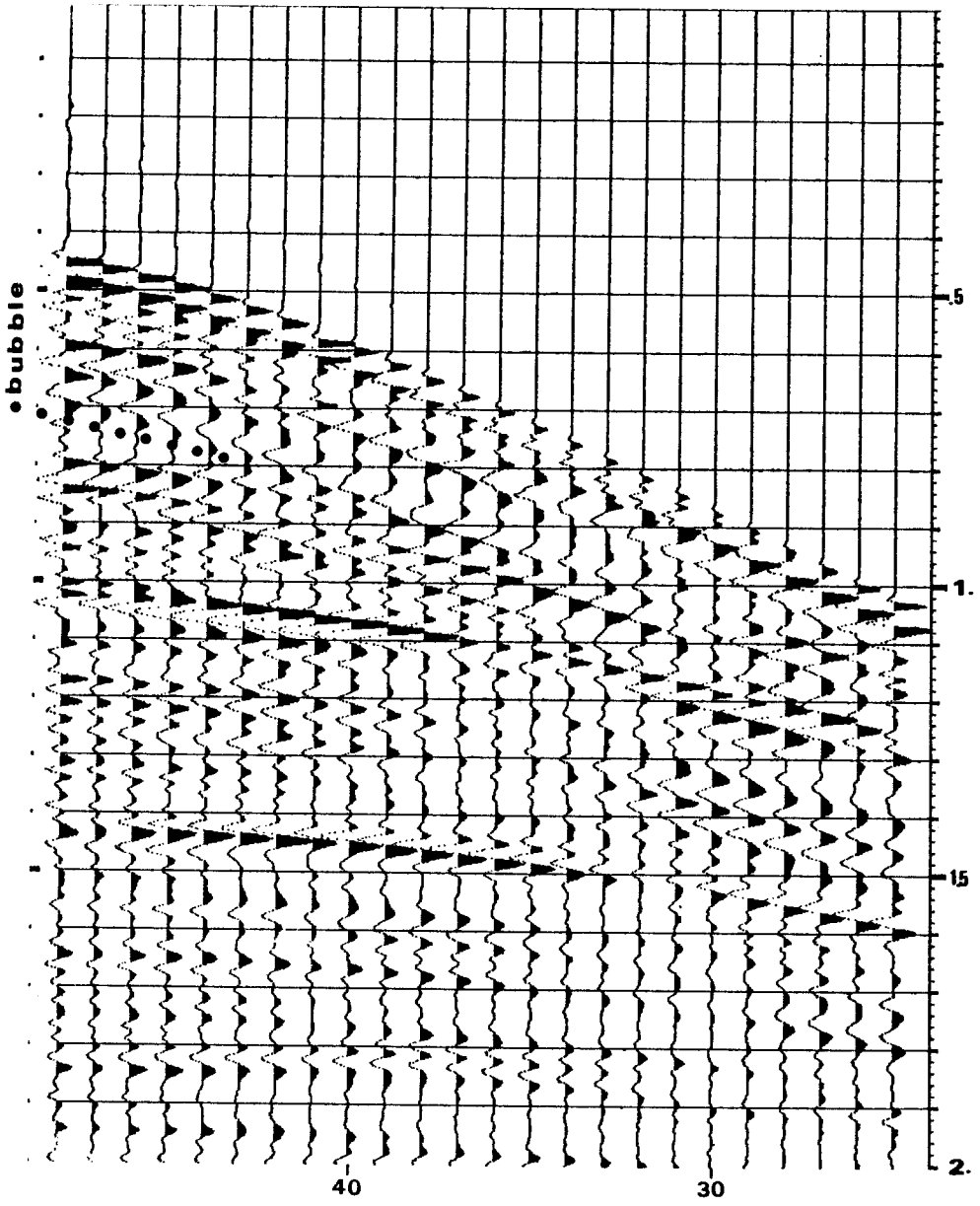


Figure 10

Original data. Bubble pulse is denoted by the dots between .7 and .8 sec. on the left.

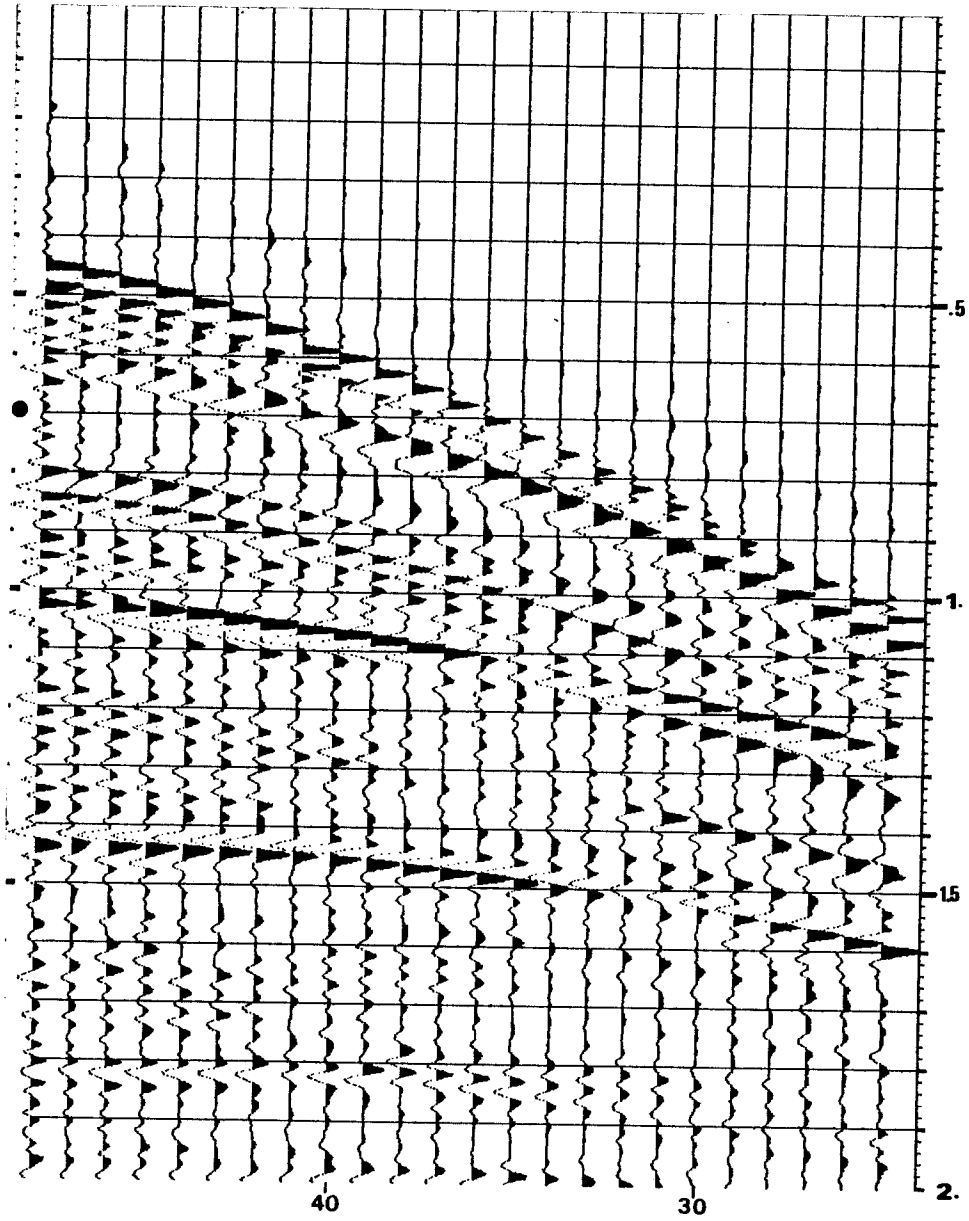


Figure 11

Variable norm deconvolution for
$\alpha = 4$
after 6 iterations.

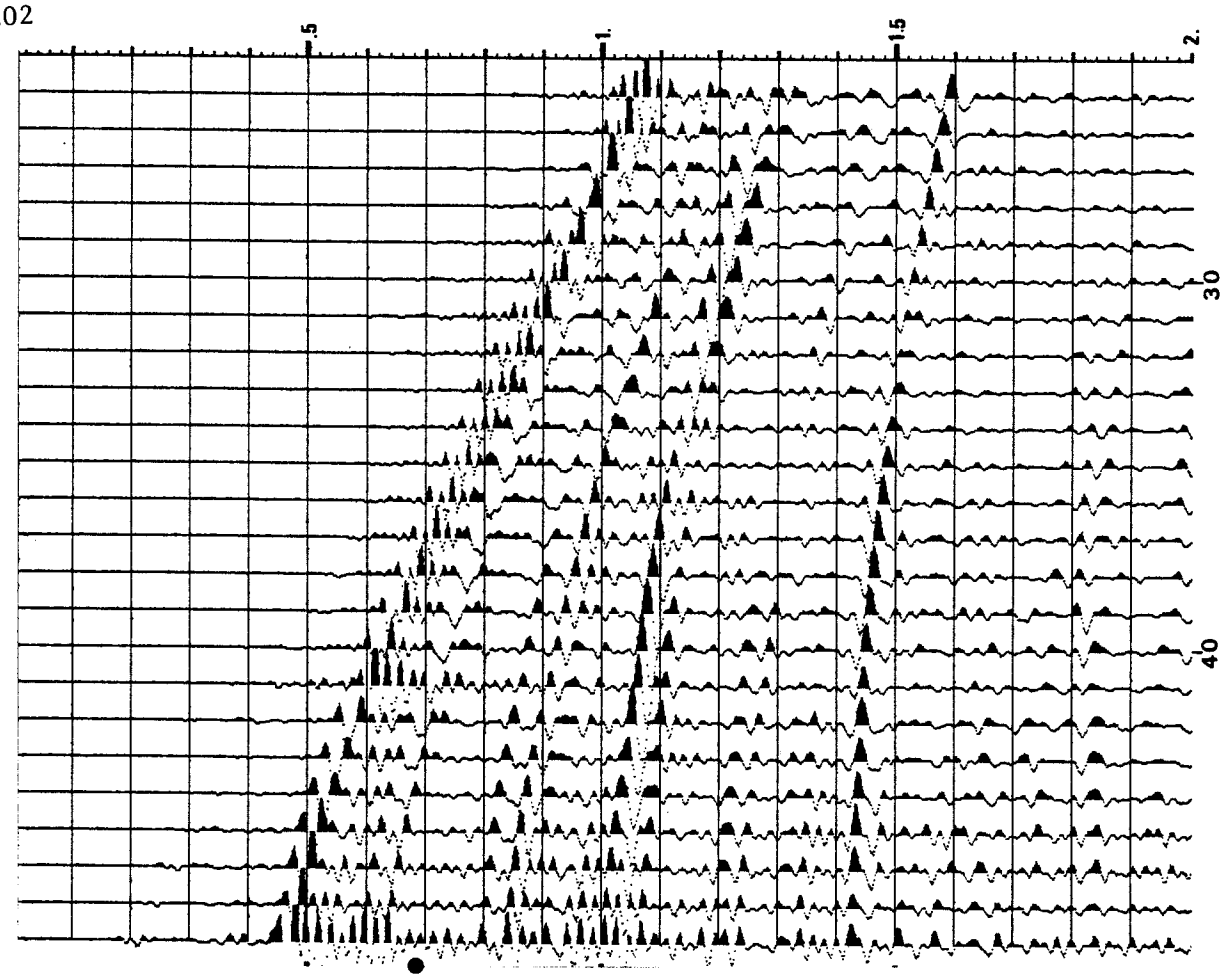


Figure 13
Variable norm deconvolution for
$\alpha = 6$
after 5 iterations

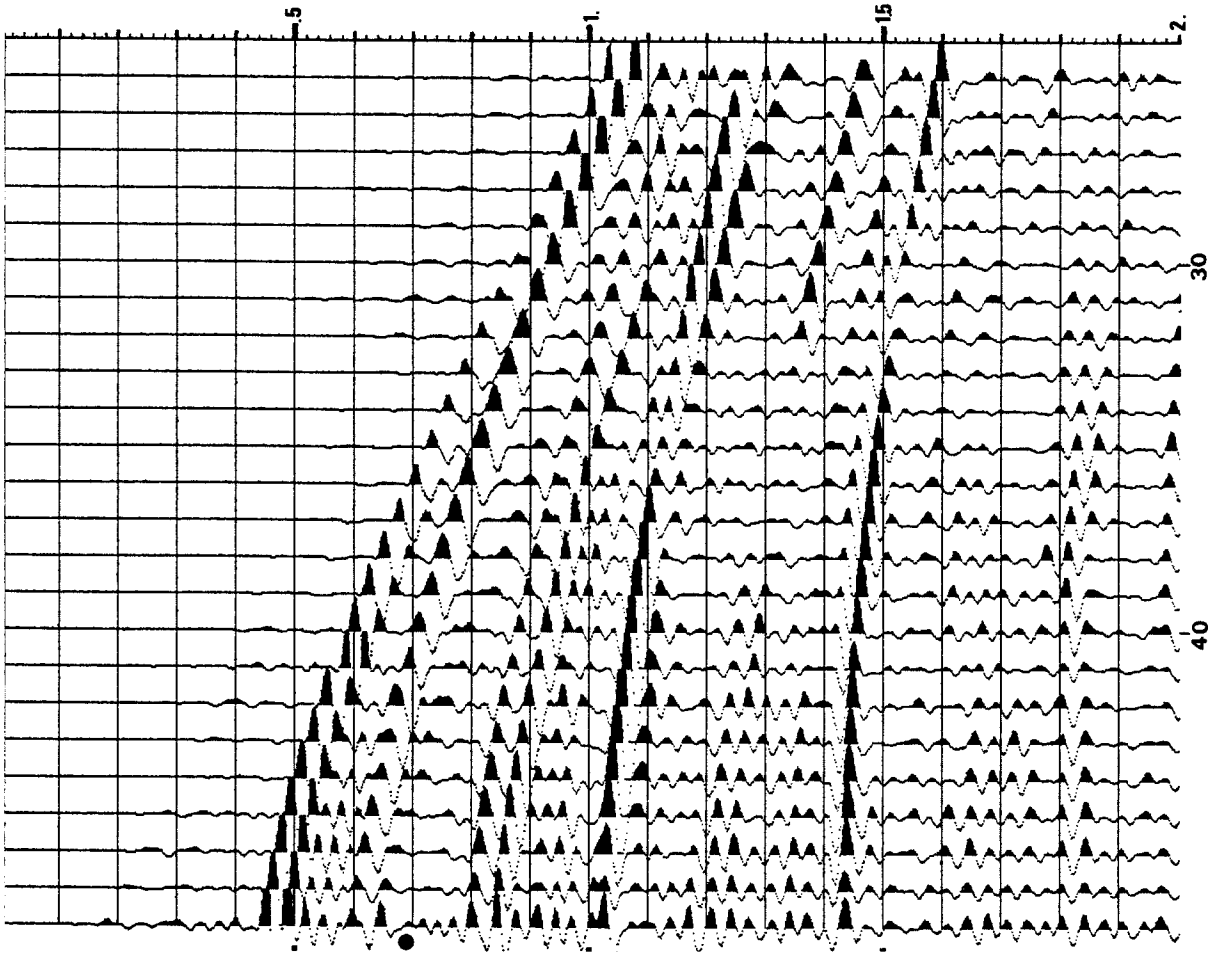


Figure 12
Variable norm deconvolution for
$\alpha = 2.5$
after 9 iterations

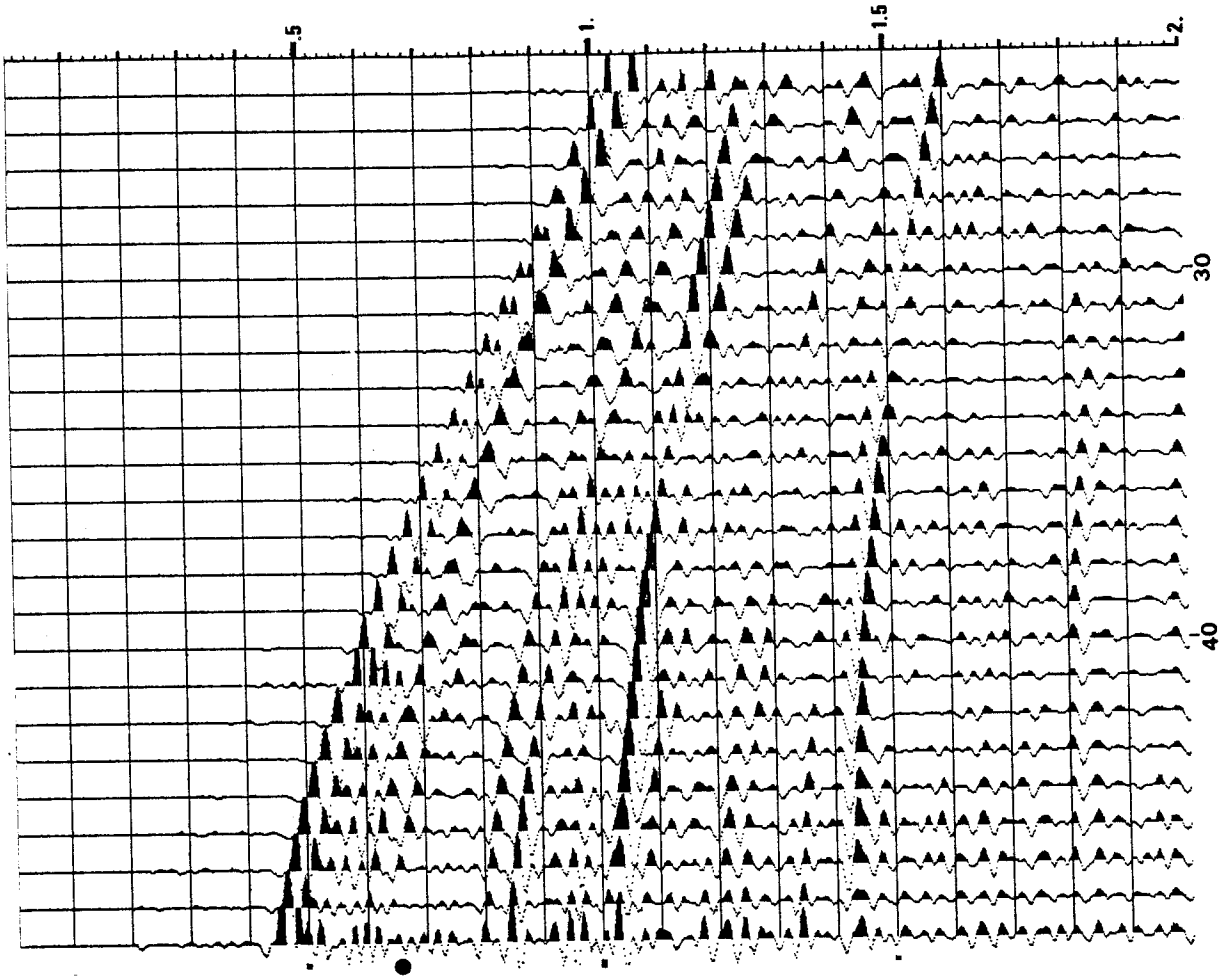


Figure 15
Parsimonious Deconvolution

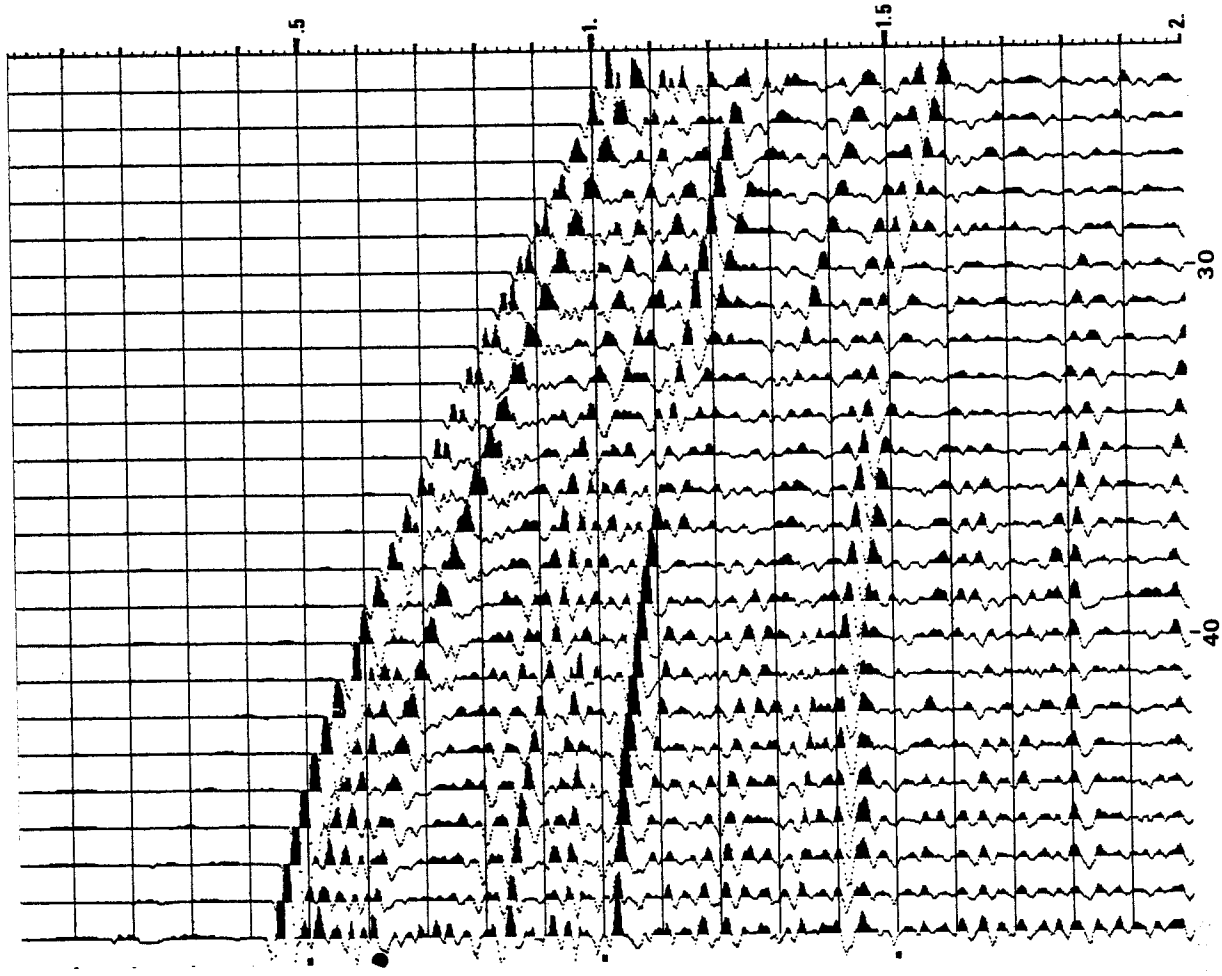


Figure 14
Spectral Smoothing Deconvolution

Objective comparison of the deconvolutions is difficult. All have eliminated the prominent bubble oscillation labeled on the original data. The deconvolution for $\alpha = 6$ looks noisy but when viewed from the side at a low angle it shows the layering near the sea floor in greater detail than any of the others. Deeper in the section it seems to have degraded some reflections. All the variable norm deconvolutions have a slightly non-causal source estimate. This is evidenced by a small amount of energy before the sea floor reflection.

Conclusion

The variable norm algorithm was developed because MED seemed excessively biased towards the larger amplitudes of the recorded data. The parameter α seems to govern this bias. This has not been shown theoretically but is indicated by the deconvolutions performed on both synthetic and field data.

References

[1] Wiggins, R. A., "Minimum Entropy Deconvolution," Proceedings of the International Symposium on Computer Aided Seismic Analysis and Descrimination, (1977), June 9 and 10, Falmouth, Mass., IEEE Computer Society, pp. 7-14.

[2] Claerbout, J. F., "Parsimonious Deconvolution," Stanford Exploration Project, Report number 13, (1977), Stanford University, Stanford, CA., pp. 1-9.

[3] Ferguson, G. A., "The Concept of Parsimony in Factor Analysis," *Psychometrika*, (1954), Vol. 19, No. 4, pp. 281-289.

[4] Bracewell, R., "The Fourier Transform and Its Applications," McGraw-Hill, (1965), New York, N.Y., pp. 143-173.