Missing Data: Wanted, Good Convergence in a Few Steps

Jon F. Claerbout

In the 1960s we learned how to apply optimization theory to seismic data. In the 1970s we learned to apply the wave equation. One of the big tasks for the 1980s will be to learn to do both at once. To illustrate the need and the potential we have only to notice that about half of the data presently recorded is never migrated. The data is noisy to begin with, and migration would only add to the chaos.

A useful and interesting problem which combines optimization and wave propagation theory is the problem of missing data. In one dimension the term *missing data* means nothing more nor less than the interpolation and extrapolation of a time series. In one dimension the theory presented here may add little or nothing to the conventional mathematical approach.

In two dimensions we know there is something wrong with "conventional wisdom" because we often see crossing events which can be interpolated better by human beings than by computers. Perhaps we really don't need new mathematics, we just need to figure out how to use what is already there, in such books as that of Wiener and Khinchin. On the other hand perhaps we already know all the math we need. Maybe we just face an impossibly large optimization problem.

Overall Strategy

Start by dividing data space into two parts. One part contains the known data, the other part contains the missing data. We presume the existence of an *invertible transformation* from *data space* to *model space*. (Later on we may worry about whether the transformation is exactly invertible or only approximately so.) Next we must further presume that we can find the missing data by making things as simple as possible in model space. That is, we obligate ourselves to devise some optimization function to express the idea of

idea of simplicity in model space. The function may be just a weighted sum of squares in model space. The function is indirectly a function of the missing data. The task is to test all possibilities of missing data so as to extremalize the optimization function, hence to create the most simplicity in model space. Thus the missing data is chosen to create the most parsimonious model space. Examples of various situations are shown in the table below.

data space	transformation	model space	for whom parsimonious
$y(t_1),y(t_2),y(t_3)\ldots$	polynomial $y(t)=a_0+a_1t+a_2t^2$	a_0,a_1,a_2	mathematician (doesn't use $a_3,a_4,\cdots$)
$y(t)$	Fourier transform	$F(\omega)$ $\quad \lvert\omega\rvert<Nyquist$	most engineers
$y(t)$	Fourier transform	$F(\omega)$ $\quad \omega_{\max}-\omega_{\min}<2\pi$	clever engineers
y_t	polynomial divide	x_t	auto regression modeler
$CDP(h,t)$	focus	$CDP(h,z)$	layer cake geophysicist
$Section(y,t)$	migrate	$Section(y,z)$	geologist
$f(t,x)$	2-D Fourier	$F(\omega,k)$	vertical seismic profile
$f(t,x)$	2-D Fourier	$F(\omega,p=k/\omega)$	vertical seismic profile
$f(t,x)$	slant stack	$F(t,p)$	refraction geophysicist

TABLE 1. Data space, transformations, and model space in various applications.

The Curse of Dimensionality

Many geophysicists have the comfortable feeling of having powerful enough computers to solve all the optimization problems they think they need to solve. Missing data problems in two dimensions present a large class of useful problems which have not yet been solved. Consider the familiar problem of Wiener-Levinson filters. The computation time is proportional to $N_d\ N_f\ +\ N_f^2$ where N_d is the number of data points and N_f is the number of filter points. Without the familiar Toeplitz trick, which forces us to make certain regularity and stationarity assumptions, the cost would be $N_d\ N_f\ +\ N_f^3$. Crudely speaking, for all optimization problems the cost goes as N^3 where N denotes the number of unknowns. Suppose we wish to interpolate a trace between every trace on a section. In this case N is about a million, say a thousand time points on a thousand channels. Now N^3 is truly huge, that is, 10^{18}, a quintillion, compared to the familiar deconvolution problem where $N_d\ N_f\ +\ N_f^2$ is only about 10^5. The situation is not, however, hopeless. After some definitions and a more precise statement of the problem, we'll return to the issue of finding a reasonable computation procedure.

Stacking: The Big Apple

We often seek small signals in the presence of large ones. We may seek primary reflections in the presence of strong multiples and strong peglegs. We may seek primaries in the presence of strong ground roll. We may seek converted shear waves in the presence of strong p-waves.

In all cases, missing data causes a stacking problem. Something akin to spectral sidelobes results from missing far traces, missing zero offset traces, possible missing negative offset traces and missing traces internal to the cable. The conventional approach is to taper the data towards the ends. But this is data falsification and we would do better to try to estimate the missing data.

Since almost all seismic data is stacked, a missing data algorithm which improves stacking would be a truly fruitful discovery.

Definitions

Let d denote a column vector containing the entire data space, both the known data and the missing data. Should the data space be a two-dimensional matrix, we just pack it, by any convenient algorithm, into the very long vector d.

Next we define a "missing data matrix" $\mathbf{M}$ to be a diagonal matrix containing zeros and ones on the diagonal. The placement of the ones and zeros is such that the missing data points are given by $\mathbf{M}d$ and the known data points by $(\mathbf{I}-\mathbf{M})d$. Given a vector full of garbage, say g we can alter the missing data without altering the known data by the construction

$$d' = d + \Delta d = d + \mathbf{M}g$$

We will have many such vectors to test. The transformation from data space to model space will be denoted by $\mathbf{F}$. We use capital letters to denote quantities in model space, say

$$D = \mathbf{F}d$$

$$D' = D + \Delta D = D + \mathbf{F}\,\mathbf{M}g$$

Next we introduce some weighting functions by means of a diagonal matrix $\mathbf{W}$ with a positive weighting function on the diagonal. A simple optimization problem is to minimize the quadratic scalar Q where

$$Q = D^*\mathbf{W}D$$

subject to some constraints. Let α be an unknown scalar and let the garbage vector g be a vector of random numbers. Given any vector d_0 which satisfies the constraints (i.e. the observed data equals (I-M)d_0), we can estimate the missing data in such a way as to improve model space as follows. Let

$$d_1 = d_0 + \alpha \mathbf{M} g$$

$$\mathbf{F} d_1 = \mathbf{F} d_0 + \alpha \mathbf{F} \mathbf{M} g$$

$$D_1 = D_0 + \alpha G$$

As a function of the unknown scalar α, the optimization function is

$$Q = (D_0 + \alpha G)^* \mathbf{W} (D_0 + \alpha G)$$

The usual procedure to find α is to set $\partial Q / \partial \alpha^*$ to zero

$$0 = G^* \mathbf{W} (D_0 + \alpha G)$$

getting

$$0 = G^* \mathbf{W} (D_0 + \alpha G)$$

$$\alpha = - \frac{G^* \mathbf{W} D_0}{G^* \mathbf{W} G}$$

Of course if the g vector were just random numbers, you wouldn't expect a very great improvement in Q, but if you were to repeat the whole procedure enough times, you would expect eventually to get to the minimum value of Q.

Where the Weights Come From

Now you might ask, where do we find the weights W? The answer is given by conventional estimation theory. The weights should be the inverse of the expectation of the power in what you are estimating. The most familiar example is Wiener filter theory where

$$Q = D^* \mathbf{W} D = \sum_\omega \frac{|D(\omega)|^2}{\textit{expected power spectrum of data}}$$

$$Q = D^* \mathbf{W} D = \sum_\omega \frac{|D(\omega)|^2}{\textit{Signal spectrum} + \textit{Noise spectrum}}$$

That is to say, we usually compute the spectrum of the available data, smooth it to simulate expectation, presume it to be the sum of the signal spectrum plus the noise spectrum, then we effectively divide by it when we solve a Toeplitz system.

You don't have to be a Wiener theory wizard to understand why Q as defined above is a good quantity to minimize. You probably know that when sums of squares are minimized, there is a strong endency to avoid concentration of errors in one place. There seems to be a weak tendency for the squares to distribute themselves about as uniformly as the constraints will permit. Thus the missing data is found so as to make the unknown numerator of Q tend towards its presumably known denominator.

Baring Up Under the Curse

The futility of seeking exact solutions was previously described. Let us now turn to iterative approaches in the optimistic hope that although *exact* answers may escape us, perhaps *good* answers can be quickly found. Before getting carried away with optimism, the true enormity of the two-dimensional problem should be appreciated. The dimensionality of the data space has gone from a thousand to a million. The number of unknowns has gone from 100 in a typical filter problem to say 10^5–10^6 in a missing data problem. Optimistically assuming that a "sensible" number of iterations is proportional to only the square root of the number of unknowns, you conclude that "only" a thousand migrations are required to find missing interlaced traces! Or only 300 migration-diffraction pairs are required to find 50 missing traces off each end of the data.

The uncharacteristically small page count for this author in this report tells you something. It tells you that many wild ideas were tried, few of which worked well enough to warrant reporting. However, I am still optimistic (if slightly paranoid) in the belief that we will find applications in which only one or two or three iterations will be well worthwhile, certainly much better than assuming that the missing data vanishes.

The Purpose of the One-Dimensional Trials

The one-dimensional missing data problem is nothing more than the traditional problem of extrapolation and interpolation. The purpose of the one-dimensional study is not to try to improve on the historical wisdom. The purpose is to work on problems which we already understand very well, but to work on them in such a way that ideas and techniques may be tested quickly and then readily carried over to two dimensions.

We have many questions to answer. The main one of course is, "What methods give good answers in only a few iterations?" This question will lead us to other questions such as, "How important is it to use the "correct" weights?" What will we do when we discover that falsifying the weights gives us subjectively better answers in the first few iterations

although poorer answers at the end? May we find the weighting function by smoothing the solution we are iteratively obtaining?

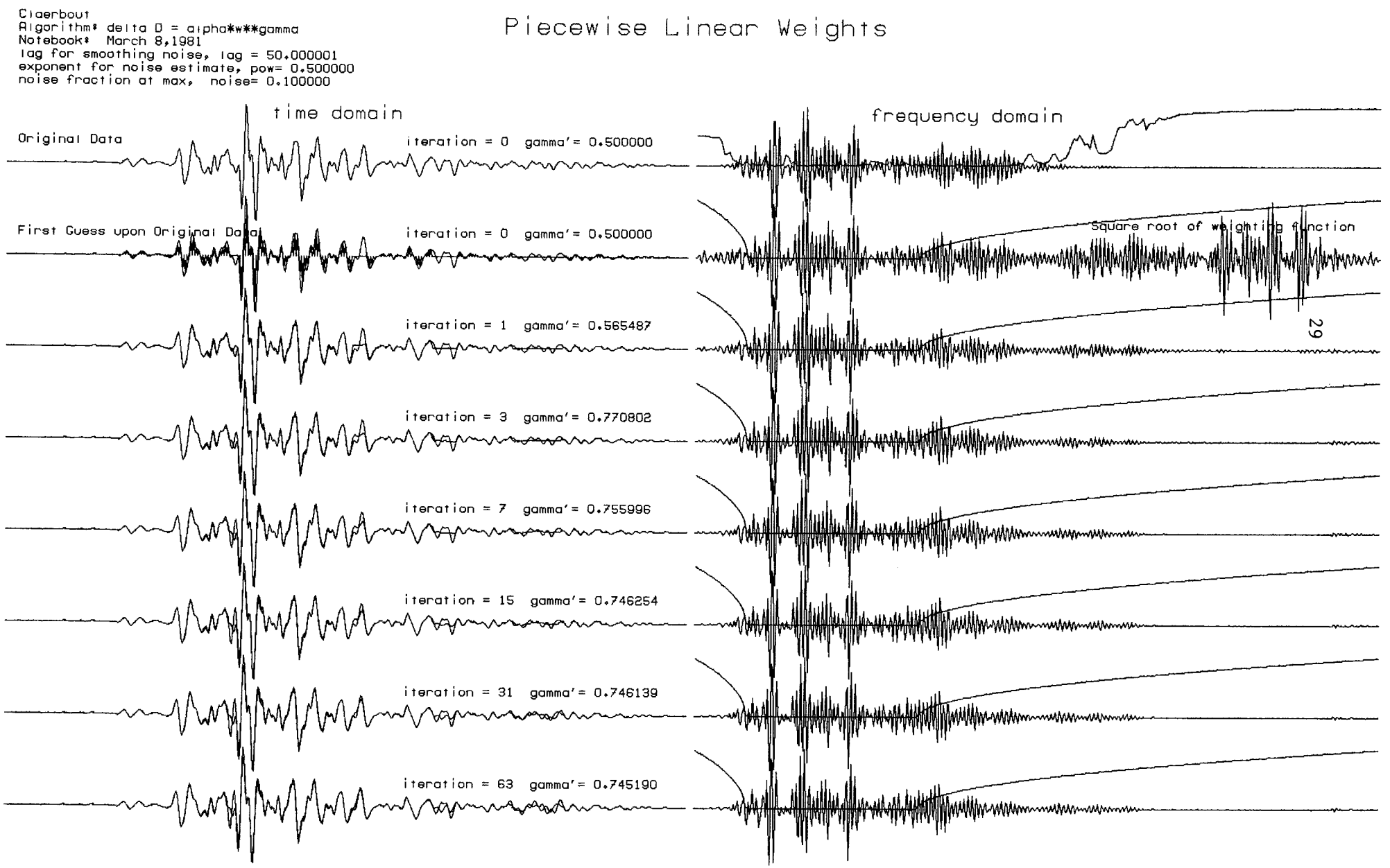
If we are not careful we may fall into the mental trap of demanding answers which converge in the limit of an infinite number of iterations. Convergence is nice, but it is not wise to insist on it. For example, there is a well known class of infinite series with factorials in the numerators. These series, called asymptotic expansions, generally diverge, but the first few terms provide excellent approximations. Indeed some mathematical physicists have been known to jest, "We are lucky if it diverges because then it will provide better practical approximations". Another example, closer to home, is Muir's continued fraction expansion of wave extrapolation operators. This expansion does not converge to the full scalar wave equation. In fact it diverges in the evanescent zone. Yet we are all very pleased with it.

The Test Data

For a realistic example, a field data set was gapped in such a way as to create large regions of indeterminacy in both data space and model space. The data set is the same one considered in SEP-25. It is a far trace on a marine profile. Alternate points were zeroed creating a high frequency indeterminacy in model space. Also some larger regions in the time domain of size about a half, one, two, and four wavelengths have been zeroed. In the larger gaps there is clearly some time domain indeterminacy. The top line of figure 1 shows the original trace on the left, and its spectrum on the right. The second line shows the same data as the first, but it is superposed on the gapped data. Subsequent lines show subsequent iterations which will be described in due course. The right column, which shows the frequency domain, displays a superposition of the real part of the Fourier transform, its imaginary part, a weighting function (often based on the spectrum), and (optionally) the inverse of the weighting function.

Piecewise Linear Weights

The lesson we will learn from the piecewise linear weight example of figure 1 it that is is not terribly important to use weighting functions which faithfully fit the Wiener estimation theory. On the right is superposed on the real and the imaginary parts of the Fourier transform of the data. The weight function is zero in the main part of the seismic band from 1/10 to 1/3 Nyquist. From the edges of this band it tapers linearly to the ends of the frequency axis. This weighting function is nothing but a crude statement that we like to see energy in the seismic band. Notice that this choice of weights is a wide departure from those proposed by the Wiener theory. If we had to take these weights to be Wiener


Claerbout
Algorithm: delta D = alpha***gamma
Notebook: March 8,1981
lag for smoothing noise, lag = 50.000001
exponent for noise estimate, pow= 0.500000
noise fraction at max, noise= 0.100000
Piecewise Linear Weights
time domain
frequency domain
Original Data
iteration = 0 gamma'= 0.500000
First Guess upon Original Data
iteration = 0 gamma'= 0.500000
Square root of weighting function
iteration = 1 gamma'= 0.565487
iteration = 3 gamma'= 0.770802
iteration = 7 gamma'= 0.755996
iteration = 15 gamma'= 0.746254
iteration = 31 gamma'= 0.746139
iteration = 63 gamma'= 0.745190
29

weights it would mean that we expected infinite energy within the crudely determined seismic band, and proportionately zero outside the band. Yet despite such obviously poor assumptions we see that the interpolated data is not at all bad after a dozen iterations. It certainly is a lot more realistic than assuming zeros.

After a large number of iterations the solution appears to converge to an answer which has a nice seismic look. It seems to be reasonable in the small gaps, but as we expect, it is clearly incorrect in the large gaps.

The problem solved here is apparently somewhat indeterminate since anything could happen where the weights vanish. In practice I found that some descent procedures indeed gave very large garbage in the long gaps. But this is not of any real interest, since we clearly have no need to use zero weights. With positive, non-zero weights there was never divergence.

Example of Wiener's Weights

Having seen what may be considered the worst, or most crude, choice of weights, let us now see something near the best, namely Wiener weights. The general conclusion is that the final solution is much better, and convergence comes somewhat quicker. The equation below relates to the computer program parameters

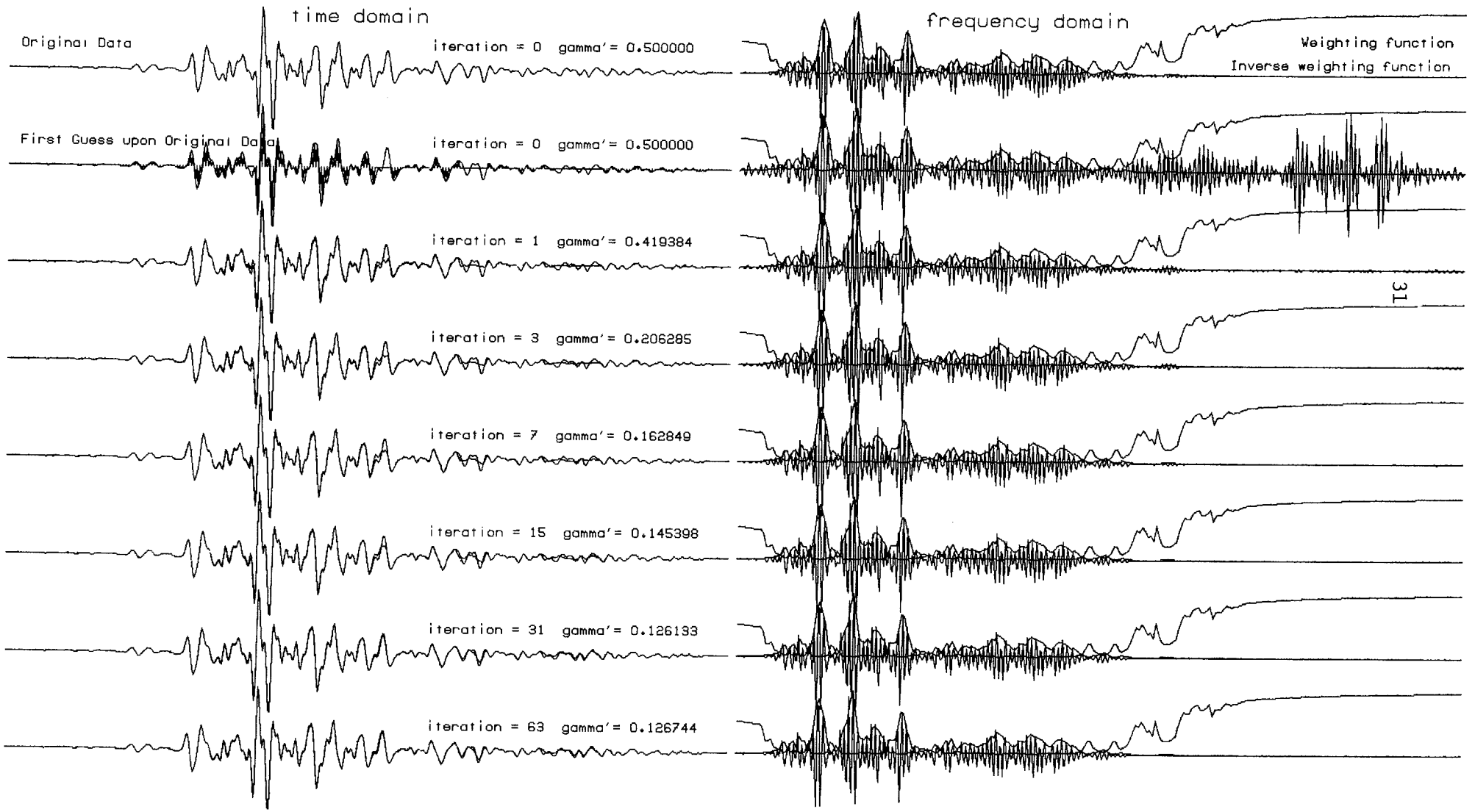
$$\frac{1}{\mathbf{W}} = \textit{Estimated Data Spectrum } (pow, lags, noise)$$

$$= \frac{|D(\omega)|^2}{\max_{\omega} |D(\omega)|^2} + Noise \left[\frac{<|D(\omega)|^2>_{lags}}{\max_{\omega} |D(\omega)|^2} \right]^{pow}$$

You will see that this choice of weights is so good that the example is almost a cheat. First of all I had the original data before the so-called "missing" data points were abandoned. This original data was used to compute the signal spectrum $|D(\omega)|^2$. The "white" model of noise was thought to be a poor one. So for the noise model the signal spectrum was brought into the autocorrelation domain and tapered under a linear taper of length $lags$ =50 time points. Next the spectrum was raised to the power pow =1/2. This is thought to be a nice compromise between white noise, pow=0, and noise with the same spectrum as the signal pow=1. Finally, the amount of the noise was taken to be 10% at the signal maximum. Because of the square root you can see that at high frequencies the noise actually dominates the signal. Figure 2 shows excellent results after a large number of iterations.

Claerbout
Algorithm: delta D = alpha***gamma
Notebook: March 8,1981
lag for smoothing noise, lag = 50.000001
exponent for noise estimate, pow= 0.500000
noise fraction at max, noise= 0.100000

Wiener Type Weights



The Search Direction

The Wiener weights, however estimated and smoothed, are our favorite choice for the ultimate solution. But we are really not so interested in the best ultimate solution as we are in getting a good solution in the first few steps. Clearly, we should focus on the search direction. We have already commented that the search direction ΔD could be chosen randomly, that is, if G is a random vector in model space then

$$\Delta D \;=\; \alpha\, \mathbf{F}\mathbf{M}\mathbf{F}^{-1} G$$

is a possible search direction. The determination of the scaling factor α is what really depends on the Wiener weights. It is very popular, however, to avoid the use of a random direction and use the gradient direction, that is,

$$E \;=\; D^* \mathbf{W} D$$

$$G \;=\; \frac{\partial E}{\partial D^*} \;=\; \mathbf{W} D$$

$$\Delta D \;=\; \alpha\, \mathbf{F}\mathbf{M}\mathbf{F}^{-1}\mathbf{W} D$$

This seems like a good idea, to have the search direction also depend on the Wiener weights, but from the point of view of getting the best possible answer in the first iteration, it really isn't such a good idea. Consider a situation where there are no effective constraints, say, in some region of interest, $\mathbf{F}\mathbf{M}\mathbf{F}^{-1}$ behaves almost like an identity matrix. With little or no constraint the value of E should turn out to be near zero with a value of $D' = 0$. And we should be able to find this result in one iteration. That is to say, we are better off to choose an α (which would turn out near - 1) for the trial solution

$$D' \;=\; D + \alpha\, \mathbf{F}\mathbf{M}\mathbf{F}^{-1}\, D$$

than we would be with the gradient trial solution

$$D' \;=\; D + \alpha\, \mathbf{F}\mathbf{M}\mathbf{F}^{-1}\mathbf{W}\, D$$

Studying these two expressions you can see that their dissimilarity is roughly measured by

$$\textit{Dissimilarity} \;=\; \frac{\max_{\omega} \mathbf{W}(\omega)}{\min_{\omega} \mathbf{W}(\omega)} > 1$$

Thus you might suspect that convergence of gradient methods tends to be slower when there is a wide dynamic range for the weights. Such a wide dynamic range could be caused by something almost irrelevant to the problem, such as a low level of both signal and noise near the Nyquist Frequency.

Reasoning that a compressed dynamic range would speed convergence we are motivated to consider a search direction

$$\Delta D \;=\; \alpha\, \mathbf{F M F}^{-1}\, \mathbf{W}^{\gamma}\, D$$

where γ is a numerical parameter between zero and 1. Not knowing what a good choice of γ would be, I tried several. But learning a γ value for this problem does not elucidate the general question of what γ to use. So I considered the power series

$$f(\gamma) \;=\; f(\gamma_0) + (\gamma-\gamma_0)\, \frac{df}{d\gamma}\Big|_{\gamma_0}$$

$$f \;=\; \mathbf{W}^{\gamma} \;=\; \left[e^{\log \mathbf{W}}\right]^{\gamma} \;=\; e^{\gamma \log \mathbf{W}}$$

$$\frac{df}{d\gamma} \;=\; \log\mathbf{W}\, e^{\log \mathbf{W}} \;=\; \mathbf{W}^{\gamma}\log\mathbf{W}$$

Therefore the trial solution has two free parameters α and β

$$\Delta D \;=\; \alpha\, \mathbf{F M F}^{-1}\, \mathbf{W}^{\gamma_0}\, D \;+\; \beta\, \mathbf{F M F}^{-1}\, \mathbf{W}^{\gamma_0} \log\mathbf{W}\, D$$

$$\Delta D \;=\; \alpha G_0 + \beta G_1$$

The numerical values of α and β used at each iteration were found by solving the 2-by-2 least squares system

$$\left\{\sum_{\omega} \mathbf{W} \begin{bmatrix} G_0^* \\ G_1^* \end{bmatrix} [\, G_0 \;\; G_1 \,]\right\} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \;=\; -\sum_{\omega} \mathbf{W} \begin{bmatrix} G_0^* \\ G_1^* \end{bmatrix} D$$

Also we get to see the supposed best γ, say $\gamma' = \gamma_0 + \beta/\alpha$. In practice I chose $\gamma_0 = 1/2$.

Search Direction Summary

A variety of strategies for search directions were tried, including steepest descent, conjugate gradients, and the method of the previous section. All methods ultimately led to "the answer" but none could be said to have done a very good job by the third iteration. A more drastic approach seems called for.

Falsifying the Optimization Criterion

To further speed convergence, albeit to the wrong answer, the dynamic range of the Wiener weights was compressed by square-rooting. This is shown in figure 3. At iteration 3 and at iteration 7 the answer is subjectively somewhat better in all sizes of time gaps. As expected, the ultimate answer is not so good as before.

Negative Weights

Only one method has been found which gave an impressive answer in the first iteration. This method, however may be divergent after a large number of iterations. This method is still somewhat speculative and is being presented only because the result of one iteration was so good. Consider a norm ratio

$$E \;=\; \frac{D^* \mathbf{W}_n D}{D^* \mathbf{W}_d D} \;=\; \lambda$$

This is a bit reminiscent of minimum entropy problems but the numerator and denominator are L_2 norms with different weighting functions. The variation of E is

$$\delta E \;=\; \frac{\delta Num}{Den} - \frac{Num}{Den}\frac{\delta Den}{Den} \;=\; \frac{1}{Den}\Big[\delta Num - \lambda\, \delta Den\Big]$$

This is suggestive of a computational technique in which you first compute $E = \lambda$ from the initial value of D. Then you work with the new extremalization

$$\mathrm{E}' \;=\; D^*(\mathbf{W}_n - \lambda \mathbf{W}_d) D^*$$

The effective weighting function $\mathbf{W}_n - \lambda \mathbf{W}_d$ is clearly non-positive, so we must expect something rather drastic. Drastic it is. The procedure did not appear to converge for any weights chosen. But some weights gave good answers in the first iteration. A good case is shown in figure 4 where $\mathbf{W}_n = \mathrm{I}$ and $\mathbf{W}_d = (\, \textit{Wiener Spectrum}\,)^{1/10}$.

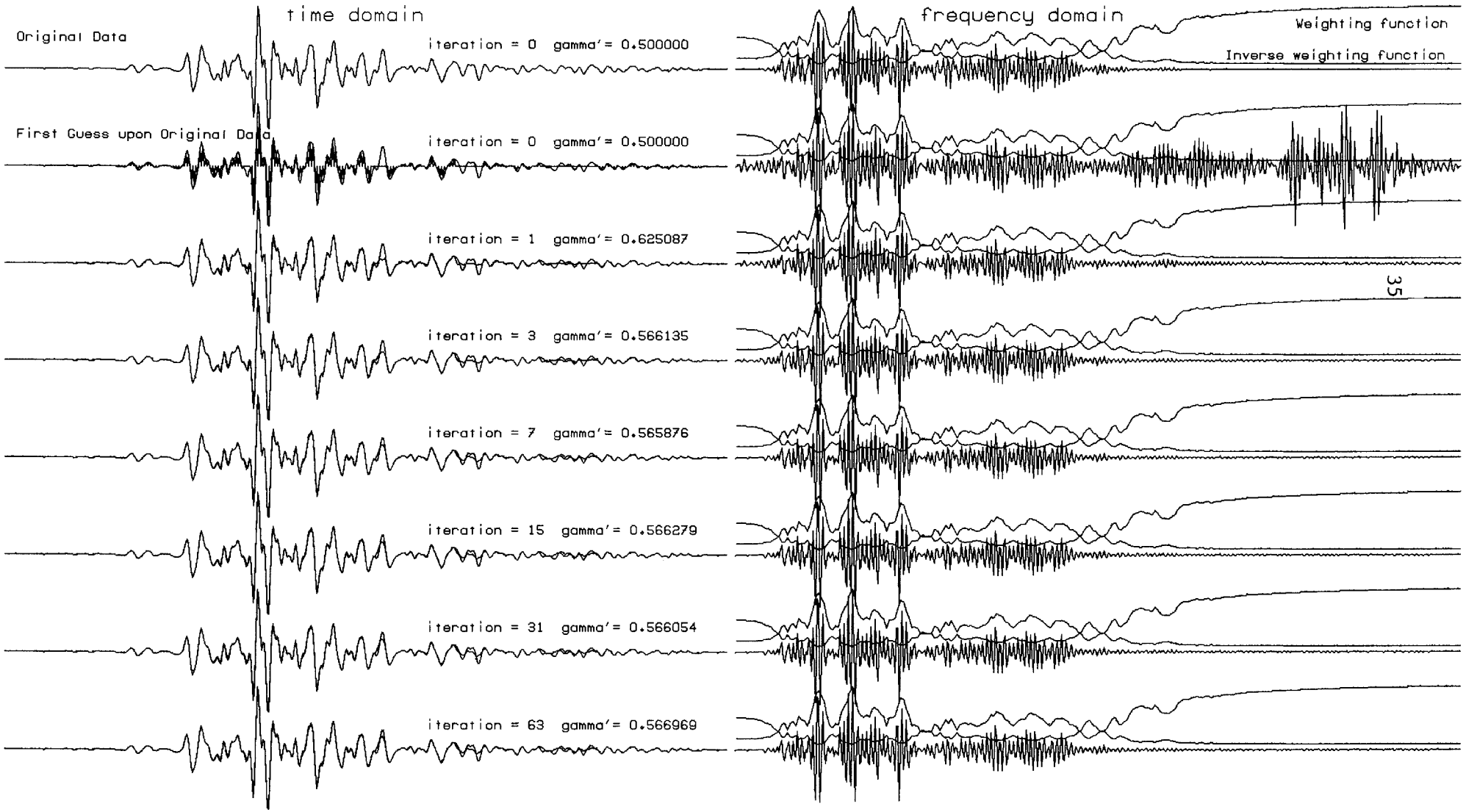
As of this report deadline, I have been unable to gather all these ideas together to create a convergent method which makes such good progress in the first iteration.

Solution Dependent Weights

It was mentioned earlier that the Wiener weight example was a bit of a cheat since the spectrum was known and used. Ordinarily it would have to be estimated from the data itself. This could be done from the zero padded raw data. A question is whether we can do better by basing the weight function $\mathbf{W}$ on the solution being attained. Of course if the fraction

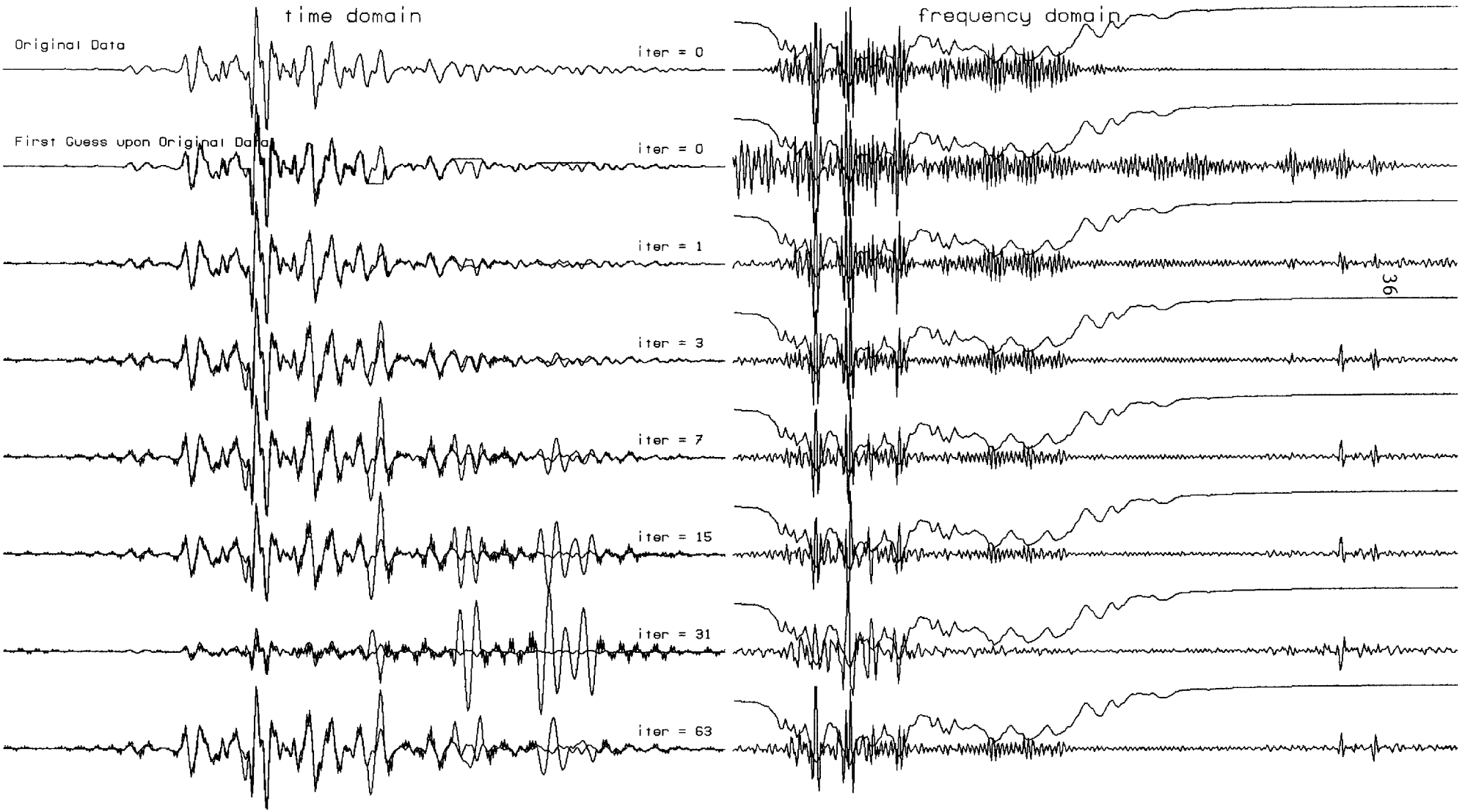
Claerbout
Algorithm: delta D = alpha***gamma
Notebook: March 8,1981
lag for smoothing noise, lag = 300.000008
exponent for noise estimate, pow= 0.500000
noise fraction at max, noise= 10000.000271

Compressed Wiener Weights



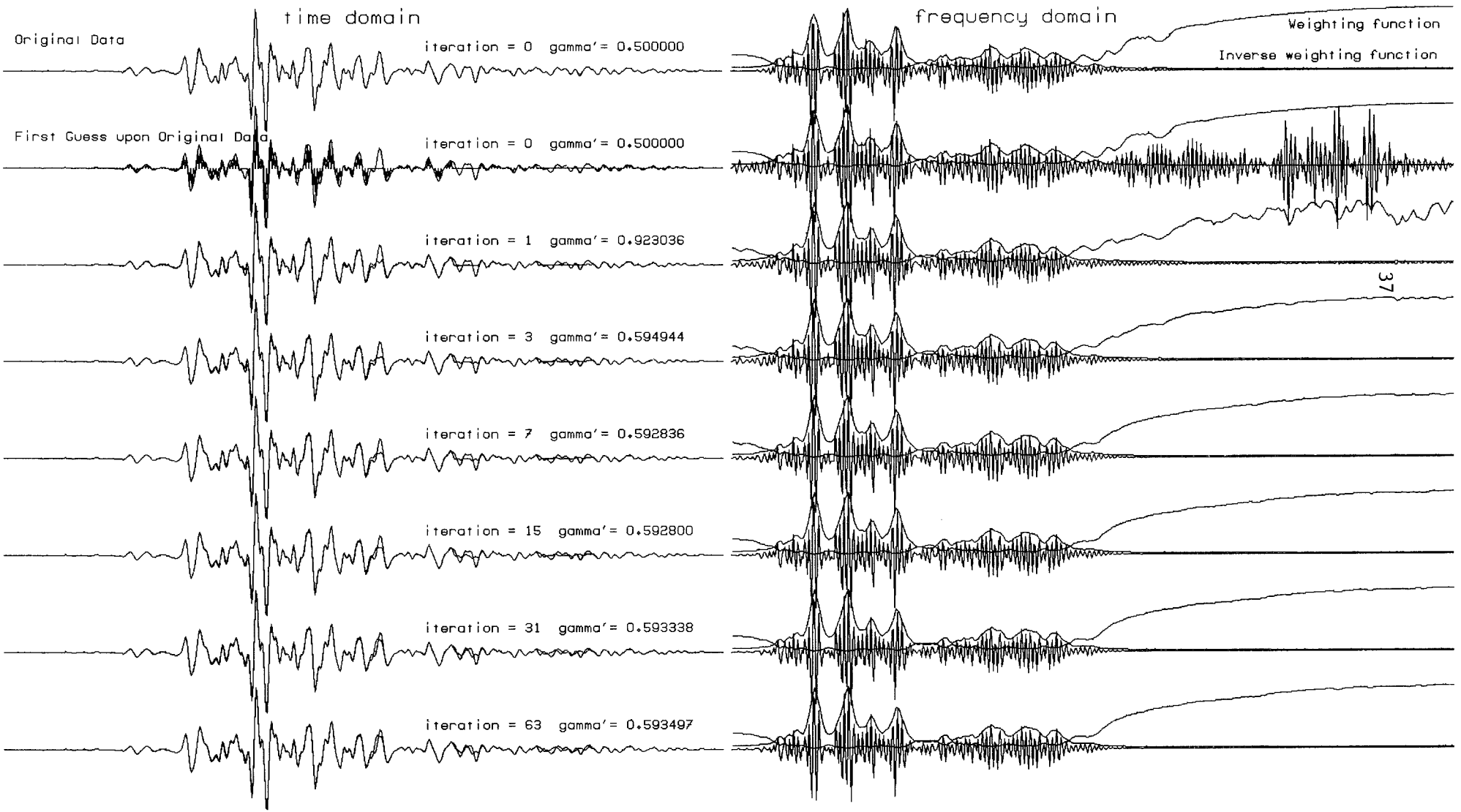
Claerbout
Algorithm: D' = D+ alpha*X+beta*G
Notebook: March 6,1981
lagn=300.000000 pown=0.000000
lagd=300.000007 powd=1.000000
lamda=1.000000

Norm Ratio Weights



Claerbout
Algorithm: delta D = alpha***gamma
Notebook: March 8,1981
lag for smoothing noise, lag = 150.000004
exponent for noise estimate, pow= 1.000000
noise fraction at max, noise= 10000.000271

Updating the Weights



of data which was actually missing was small, there should be little difference. However, should a large fraction of data be missing, we expect substantially different results if we update the weight. Of course smoothing the weight is an essential part of preventing degeneracy to an exercise of circular reasoning. The results shown in figure 5 are quite respectable. of course one always faces the practical tradeoff between resolution and sample variance.

One can regard the Burg spectral estimation procedure as updating of the weights between each reflection coefficient estimation. At each stage one supposes that the prediction residual spectrum is white.

A final attempt was made to achieve fast convergence. Notice that the costly part is the $\mathbf{F}\mathbf{M}\mathbf{F}^{-1}$ calculation. Thus is is economical to reestimate the weighting function several times before selection of a final α and β. This was tried, but there was no perceptible improvement.

REFERENCE

Khinchin, A.I.,1957, Mathematical Foundations of Information Theory, Dover Books