Parsimony criteria for missing data restoration algorithms

Jeff Thorson

Introduction

A previous paper [Thorson, 1982] discussed in general terms the artifacts that arise from missing elements of data on subsequent processing of the data. Specifically, ignoring the missing data is equivalent to padding with zeros (in all linear processes), and this can give rise to undesirable artifacts that may obscure events of real interest. Considering the problem of reconstituting data never collected in the first place, it is obvious that the choice of a method for filling in the missing data is very much a subjective one. Knowing what we want the missing traces of a data set to look like implies that we already have interpreted the data to some extent, or have extracted information from it. Let us begin by making the choice more objective, or at least try to isolate the subjectivity and quantize it. The criterion chosen for a "good" restoration is that there is a model space, or space in which we can interpret the data; furthermore, information is sparse in this space. To formalize the missing data problem, we now make it subjective only in these two points: the choice of a model space, and the selection of a parsimony or minimum entropy measure in the model space. Missing data restoration then translates to maximizing the parsimony measure in model space. The known data will appear as constraints to the optimization.

As a practical matter noise in the data must be taken into account, therefore the constraints imposed by the data should not be exact. The optimization problem is most easily derived from a Bayesian point of view.

Bayes approach to the missing data problem

Derivations of functionals that measure maximum likelihood or other estimators of a model begin with Bayes' rule:

$$p(\mathbf{u}|\mathbf{d}) = \frac{p(\mathbf{d}|\mathbf{u})\, p(\mathbf{u})}{p(\mathbf{d})} \qquad (1)$$

where $p(\mathbf{u})$ denotes the probability density of **u** and $p(\mathbf{u}|\mathbf{d})$ denotes the conditional density of **u** given **d**. The densities of equation (1) are all normalized.

To interpret Bayes' rule, let the following assignments be made to **u** and **d**:

u = ideal events in model space (free variables).

d = events in data space (knowns).

Let the functional relation between the two spaces be given by

$$\mathbf{Lu} = \mathbf{d} + \mathbf{n} \tag{2}$$

where **L** is always considered to be a linear operator, and **n** is a noise term. Let us now assign meanings to each of the probability densities in equation (1):

$p(\mathbf{u}|\mathbf{d})$ = the answer, all that there is to know about **u** given **d**. In statistical estimation, it is customary to provide a cost function, or some rule to select an optimal **u** from the density $p(\mathbf{u}|\mathbf{d})$. In this case the rule shall be to pick **u** for which the probability $p(\mathbf{u}|\mathbf{d})$ is maximized.

$p(\mathbf{d}|\mathbf{u})$ = the discrepancy in the relation **Lu** = **d**. Assign this density to be the density of the noise present $p(\mathbf{n})$. All densities $p(\mathbf{u})$, $p(\mathbf{d})$, and $p(\mathbf{n})$ will be assumed to have zero mean. Data that is known exactly will have an independent zero-mean noise component with vanishing variance, while the missing data components can be assigned arbitrarily large noise variances.

$p(\mathbf{u})$ = *a priori* knowledge about **u**. Specifying this probability density will fix a parsimony measure in model space. Just as for the case of $p(\mathbf{n})$ in the data domain, constraints on the solution **u** may be included in this density. In order to set **u** constant over a portion of the model space, components of **u** need only have an independent density of vanishing variance.

$p(\mathbf{d})$ = a normalization term which is independent of model parameters **u**. It serves to normalize equation (1):

$$p(\mathbf{d}) = \int p(\mathbf{d}|\mathbf{u})p(\mathbf{u})d\mathbf{u}$$

In the optimization of (1) with respect to **u**, this term may be ignored, because it is independent of the free variables **u**.

These meanings can be assigned to equation (1) because of the relation **Lu** = **d**. The model space is the domain of the causative random process that generates a realization **u**. Data **d** is the effect, possibly modified by an independent gaussian noise process.

The densities $p(\mathbf{u})$ and $p(\mathbf{n})$ are assumed to be gaussian in form, so that maximizing equation (1) is equivalent to minimizing the negative of the logarithm of the densities. To be more specific, assume the noise **n** to be an independent gaussian process, with variances that may change from point to point in the data space:

$$p(\mathbf{n}) = \frac{1}{M_n}\exp\left[-\frac{1}{2}(\mathbf{Lu}-\mathbf{d})^T\frac{1}{\sigma_n{}^2}(\mathbf{Lu}-\mathbf{d})\right] \tag{3}$$

where $\sigma_n{}^2$ is the diagonal covariance matrix and M_n is the normalization factor for $p(\mathbf{n})$. The information about missing traces in the data set is incorporated in $p(\mathbf{n})$ by specifying the variances of those traces to be arbitrarily large. The corresponding elements of $1/\sigma_n{}^2$ are zero.

The density $p(\mathbf{u})$ is chosen to be gaussian in appearance:

$$p(\mathbf{u}) = \frac{1}{M_u}\exp\left[-\frac{1}{2}\mathbf{u}^T\mathbf{C}^{-1}\mathbf{u}\right] \tag{4}$$

but with the important stipulation that **C** is allowed to be a function of **u**. We shall make the further assumption that events in model space are independent of each other. Then $\mathbf{C}^{-1}$ is diagonal. Here, each element $c_{ii}(u_i)$ of **C** is assumed to have the same functional dependence on u_i (i is the element index of **u**):

$$c_{ii}(u_i) \equiv f(u_i)$$

With this assumption, each point in model space will statistically behave like any other, independent of the other points. Independence of events is one of the desired qualities of model space; if it is not exactly satisfied, the covariance **C** may be taken to be banded rather than diagonal. On the other hand, the operator **L** may be redefined in an attempt to diagonalize **C**.

Under the assumption of independence, Bayes' estimator is connected to the minimum entropy property. The density $p(\mathbf{u})$ may be rewritten in the form

$$p(\mathbf{u}) = \prod_{i=1}^{N} q(u_i) \quad \text{where} \quad q(u_i) = \frac{1}{M_u}\exp\left[-\frac{u_i{}^2}{2f(u_i)}\right]$$

N is the number of points in model space. The relative entropy of **u**, apart from a constant term arising from the factor $1/M_u$, is estimated by

$$Entropy(\mathbf{u}) \approx -\mathrm{E}[\log q(u_i)]$$

$$Entropy(\mathbf{u}) \approx \sum_{i=1}^{N}\frac{u_i{}^2}{f(u_i)} \equiv \mathbf{u}^T\mathbf{W}\mathbf{u} \tag{5}$$

where **W** is a diagonal weighting function. The ensemble expectation of **u** in equation (5) can be replaced by the sum over i because of the assumption of independence. Under the constraint that the energy of **u** be fixed, $\|\mathbf{u}\|$ = constant, maximizing the joint density $p(\mathbf{u})$ is equivalent to minimizing the entropy estimator (equation (5)), depending on the form of the variance $f(u_i)$. When $p(\mathbf{u})$ is gaussian, $f(u_i)$ is constant, and from (5), the entropy is invariant as long as $\|\mathbf{u}\|$ is constant.

We can now introduce a basic criterion (or definition) for parsimony, that is,

(P1): $f(u_i)$ must be a positive monotonic increasing function of u_i. Corresponding realizations of the random process are *parsimonious*.

This assumption makes the variance of $p(\mathbf{u})$ increase with increasing u_i, making large amplitude outliers in model space more probable than in the gaussian case (constant variance). By minimizing the entropy in equation (5) under constraints, The model space will tend to collect outliers, with the remainder of the points in the space having a much smaller variance. Condition P1 thus satisfies our intuition on what a parsimonious distribution should be like in model space. Deeming [1981] discusses further the relation between condition P1 and minimum entropy functionals, the case of minimum entropy deconvolution.

One example of such an *a priori* choice for $f(u_i)$ is the following function (figure 1(a)):

$$f(u_i) = \begin{cases} \sigma_o^2 & |u_i| < u_c \\ \left[\dfrac{2(m_o - m_1)}{u_i^2} + \dfrac{1}{\sigma_1^2}\right]^{-1} & |u_i| > u_c \end{cases} \tag{6}$$

The logarithm of the single-variable probability density function $q(u_i)$ is therefore proportional to the function displayed in figure 1(b):

$$\log q(u_i) = \begin{cases} m_o - \dfrac{u_i^2}{2\sigma_o^2} + constant & |u_i| < u_c \\ m_1 - \dfrac{u_i^2}{2\sigma_1^2} + constant & |u_i| > u_c \end{cases} \tag{7}$$

This density, similar to a *normal mixture* [Godfrey, 1979], claims that there are two gaussian populations of events in **u**; the majority with variance σ_o^2, and the few with variance σ_1^2. This distribution also satisfies our intuition concerning parsimony. The specification of m_o and m_1 in equation (7) can be absorbed into the function $\sigma(u_i)$, with σ still being monotonic in u_i.

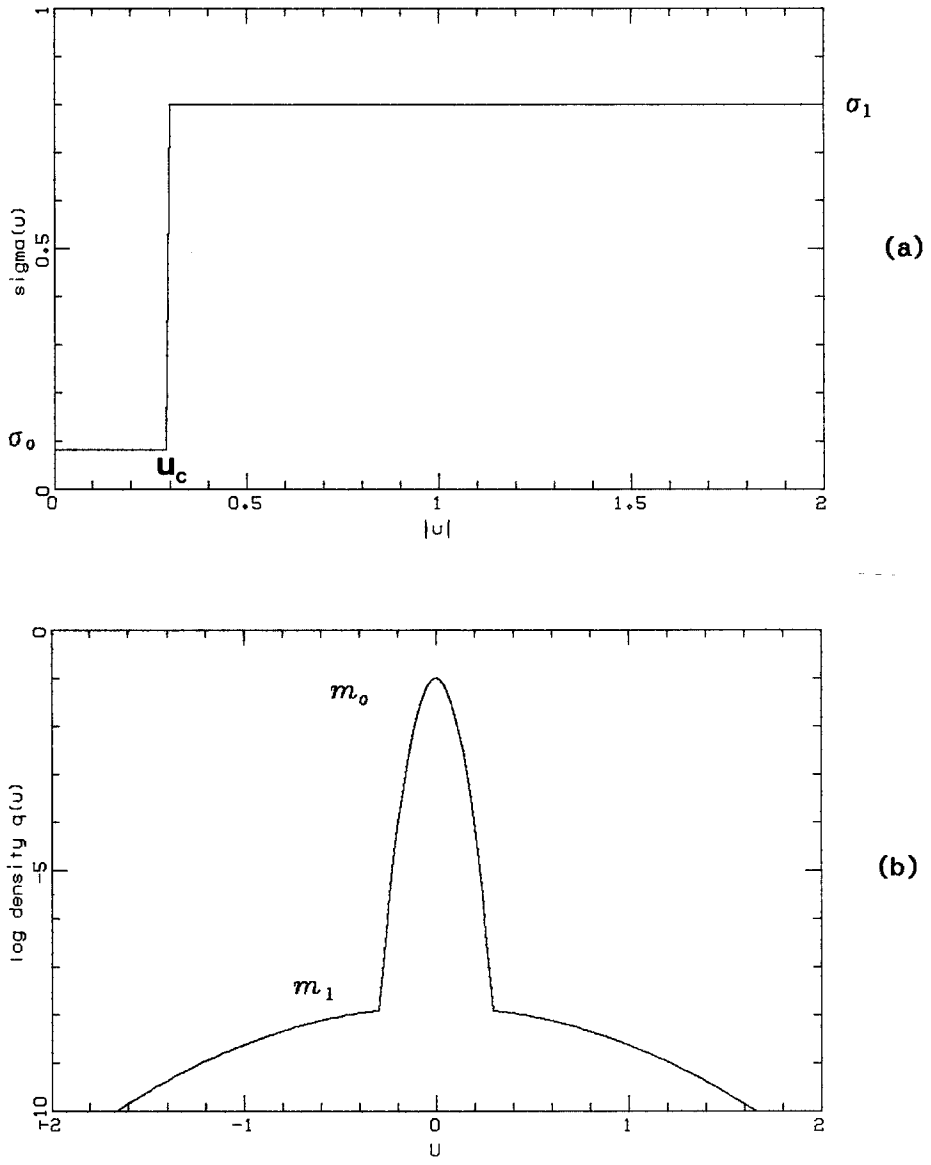

FIG. 1. Mixture of two normal distributions. (a) the monotonic function $\sigma(u_i)$ has two values σ_0 and σ_1. (b) Log density q of the distribution with the variances in (a). The relative values of m_0 and m_1 determine the relative sizes of the two gaussian populations. m_0 and m_1 can be incorporated into a monotonic function f (equation (6)).

With the form of the probability density (7) defined, taking the logarithm of equation (1) defines the parsimony functional (P2) that is to be minimized to solve **u**:

$$\min_{\mathbf{u}} \mathbf{u}^T \frac{1}{\sigma^2}\mathbf{u} + (\mathbf{Lu}-\mathbf{d})^T \frac{1}{\sigma_n{}^2}(\mathbf{Lu}-\mathbf{d}) \qquad \text{(P2)}$$

This functional has the data constraints (the second term) included as a penalty term to the

parsimony measure, the first term. If the data noise has constant variance, the noise variance acts as a penalty factor. The lower the noise, the more tightly the data constraints are applied in the model domain. The algorithm (discussed below) designed to minimize the nonlinear functional (P2) is an iterative algorithm: at each iteration, the current variance is estimated from **u**, which then is assumed constant for the remainder of the iteration step. This effectively ignores the derivative $d\sigma(u_i)/du_i$.

A nonlinear system of equations can be derived from (P2) by differentiating with respect to **u**, ignoring the derivative $d\sigma/du_i$:

$$\left[\frac{1}{\sigma^2} + \mathbf{L}^T \frac{1}{{\sigma_n}^2} \mathbf{L}\right] \mathbf{u} = \frac{1}{{\sigma_n}^2} \mathbf{L}^T \mathbf{d} \tag{8}$$

The noise variance may be assumed a constant ${\sigma_n}^2$, except at points with no data, where the variance is infinite. If this is the case, ${\sigma_n}^2$ may be pulled through the operator $\mathbf{L}^T$, leaving behind a projection that annihilates the missing data part of **d**. The projector may be incorporated into **L** without loss of generality. Multiplying through by σ^2 will give the equivalent nonlinear system

$$\mathbf{Q}\mathbf{u} = \mathbf{b} \tag{9a}$$

$$\mathbf{Q} \equiv \mathbf{I} + \frac{\sigma^2(\mathbf{u})}{{\sigma_n}^2} \mathbf{L}^T \mathbf{L} \tag{9b}$$

$$\mathbf{b} \equiv \frac{\sigma^2(\mathbf{u})}{{\sigma_n}^2} \mathbf{L}^T \mathbf{d} \tag{9c}$$

This is the system of equations that will form the basis of the descent algorithm used in this paper. Throwing away the derivative $d\sigma/du_i$ does no harm; it implies that when a solution **u** is found, its variances are locally independent of **u**.

Parsimony criteria in model space

Now that a model domain has been chosen through the operator **L**, let us examine choices for the *a priori* variance $\sigma^2(u_i)$ that defines a measure of parsimony. Under the constraint P1, that $\sigma(u_i)$ must be a monotonic function of u_i, there is a wide freedom of choice for $\sigma(u_i)$. If the statistics of the solution **u** to equation (9) are known, these can be used. This is generally not the case, so it is desirable to parameterize the function $\sigma(u_i)$ with as few parameters as necessary. The curve of figure 1 is defined by three values: σ_0, σ_1, and the cutoff value u_c that determines the boundary between the two limiting gaussian cases. The function of figure 1(a) is a fundamental one, and far from being an extreme

example, may be one of the most useful to apply to obtain parsimonious solutions to the non-linear system (9). Let us take a closer look at it.

With σ defined by figure 1, note there are three possible ranges for the solution to lie in:

(1) The entire population of **u** has the variance σ_0. If this is true, the distribution of points **u** is gaussian, equations (9) become linear, and the problem turns into a linear least squares solution for **u**, biased toward low energy in the model domain. Those points that rise above the cutoff value u_c are sparse enough not to affect the statistics of the data.

(2) The entire population of **u** in the model domain has the variance σ_1 assigned to it. This is similar to the previous case, only that many u_i values must be larger than the cutoff value u_c to properly be assigned the variance σ_1. Relatively few values of u_i can be allowed to be below the cutoff, so that they do not perturb the estimation of a gaussian distribution with variance σ_1. This implies that σ_1 must be much greater than u_c. An exception occurs when the determination of σ is made from a smoothed version of $|u|$ (see the section below on smoothing). Then certain values of u below u_c may genuinely belong to a population with a variance of ${\sigma_1}^2$, if nearby values are larger than u_c. Smoothing allows points in a common neighborhood to belong to the same population. But the problem is still gaussian, and reduces to a biased least squares estimation for **u**.

(3) The values of u_i in the model space straddle the cutoff u_c, to give an estimation of the one-dimensional density $q(u_i)$ that is distinctly non-gaussian. Two populations of points exist in the model domain, and to satisfy the intuitive criterion of parsimony or "sparseness", many points should fall into the population of low values (call this population "grass" [Rocca, 1982]). Relatively few points should fall into the high-valued population (the population of "trees"). If the model is partitioned into these two populations, each set of points will have their own separate distributions.

Because the minimization of the functional (P2) biases the solution of **u** toward zero, the energy of **u** is determined by the constraints $\mathbf{Lu} \approx \mathbf{d}$ and by the penalty factor ${\sigma^2}/{{\sigma_n}^2}$. The selection of the noise variance, the cutoff u_c, and the variances ${\sigma_0}^2$, ${\sigma_1}^2$ are crucial to the success of the optimization. They must be chosen so that the solution **u** falls into two populations: many points into the "grass" and a few large-value points into the "trees". If u_c is chosen too high or too low, **u** reverts to a least-squares solution which may not have the desirable property of sparseness. If the operator $\mathbf{L}^T$ preserves energy from data space to model space (or approximately so), the choices of parameters in figure 1 may be made more precise.

Let λ be the ratio of the expected area of the σ_1 population to that of the rest of the model domain: $\lambda = N_1/N$. Most of the energy from the data should fall in the σ_1 population, thus an estimator of σ_1 is

$$\sigma_1^2 \approx \frac{1}{N_1}\sum_{i=1}^{N_1} u_i^2 = \frac{1}{N_1}\sum_{j=1}^{N_d} d_j^2 \approx \frac{N_d}{N_1}\sigma_d^2 \tag{10}$$

where N_1 = points in the σ_1 population,

N = points in model space,

N_d = points in data space,

σ_d^2 = an estimate of variance in the data domain.

Now σ_o should be a small fraction of σ_1. It may depend on the noise variance, if known, assuming $\mathbf{L}^T$ will spread noise over model space in a gaussian way:

$$\sigma_o^2 \approx \frac{1}{N-N_1} N_d \sigma_n^2 \tag{11}$$

Finally the cutoff u_c should be put as far as possible below σ_1, but far enough above σ_o to satisfy the ratio λ. That is, the ratio of the number of points in the σ_o population that happen to rise above the cutoff should be much smaller than λ.

$$\text{erf}\left[\frac{u_c}{\sigma_o}\right] > 1 - \lambda \tag{12}$$

The effect of smoothing u on the determination of σ

Making σ a function of a smoothed or linear-filtered version of **u** has a profound effect on the solution of equation (9). It is usually the case that the shapes of the σ_1 population in model space are known *a priori*. As an example, suppose a vertical seismic profile is given as **d** and define **L** to be a 2-D Fourier transform. Linear events will line up on rays that pass through the origin in the Fourier domain. However the populations sort themselves out in the Fourier domain, points lying in straight lines passing through the origin will have a tendency to belong to the same population, with the same variance controlling them. Define smoothing windows that smooth in this radial direction. So to get a better estimate of which population a clump of energy belongs in, take the average of values $|u_i|$ over the window in order to select $\sigma(u_i)$. This smoothing process has no relation to the process of assigning positive correlations among nearby points, but rather assigns the same statistics to nearby points (those in the same window). As a consequence they will behave similarly in the solution of the system (9). Any correlation in **u** may come only from the data through the operator $\mathbf{L}^T$.

The variance of figure 1 can be selected to be a continuous function by defining it at the points $(\sigma, |u_i|)$ and interpolating linearly. But by neglecting the derivative of σ with respect to u_i in the derivation of the nonlinear system (9), the variances were assumed to be piecewise constant. This in effect generalizes the two-population case to many populations. The advantage of having more populations is not a clear one, but making $\sigma(u_i)$ continuous makes gradients continuous, which is an important factor for descent algorithms.

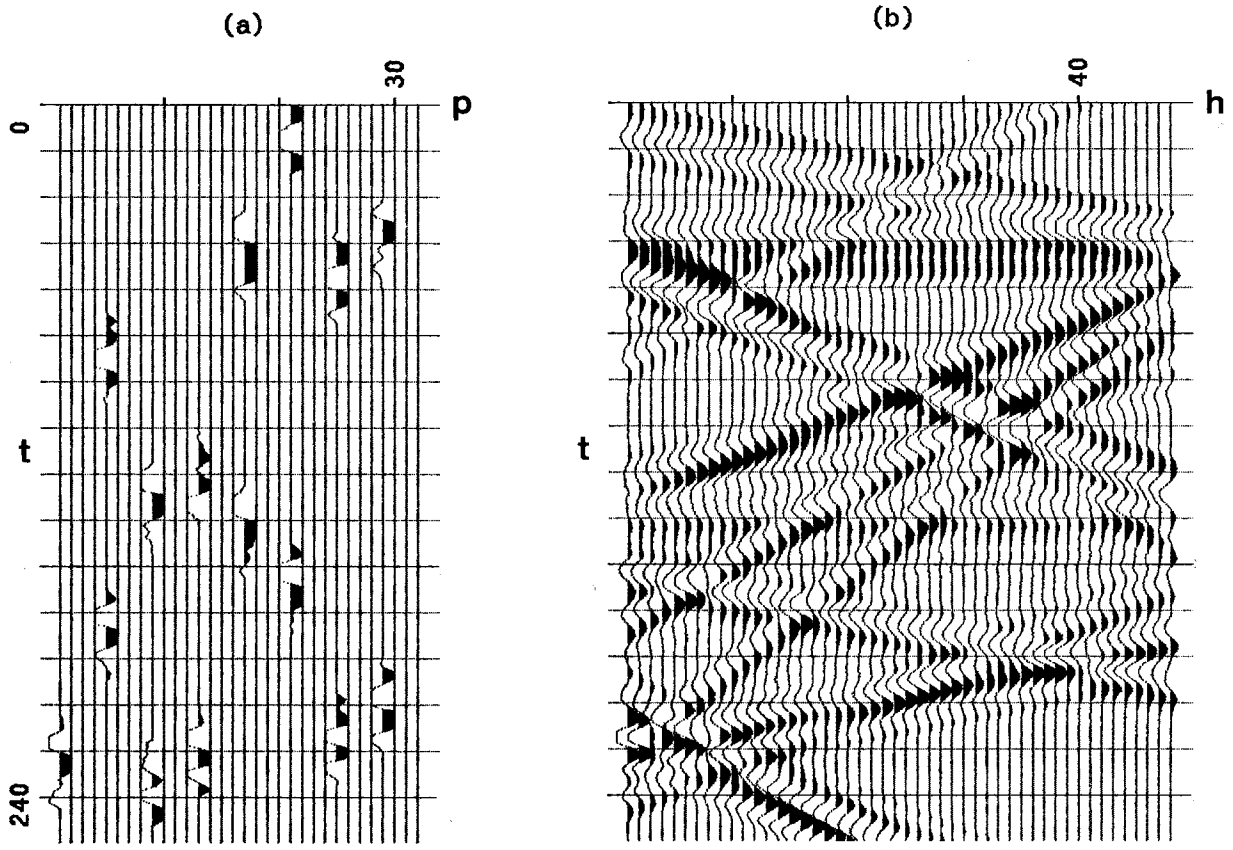

FIG. 2. (a) A sparse function in model space. A number of simple waveforms are randomly placed in **u**. The horizontal p axis represents ray parameter, though it is not necessary here to put dimensions on units. p ranges from a minimum of -3.2 to a maximum of 3.0. Δp is 0.2 and N_p, the number of traces, is 32. Data (b) is the result of applying the slant stack operator L to (a). The offset (h) axis ranges from 0 to 47, Δh = 1.0, and N_h is 48. If (b) is used as input to the system of equations (9), (a) will be the anticipated solution.

A synthetic case -- slant stacking

As a test for solving the nonlinear equations (9), consider L to be a slant stack operator. The operator, equivalent to the operations of linear moveout and stack, is defined to be

$$d(h,t) = \mathbf{L}u(p,t) = \frac{1}{\sqrt{N_p}} \sum_{p}^{N_p} u(p,t-ph) \tag{13}$$

Since **d** and **u** are two-dimensional wavefields, the indices (p,t) and (h,t) now are used in place of the single index i. The sum is taken over those p values where the index $t-ph$ is within bounds of the minimum and maximum values of the t index. The scale factor $1/\sqrt{N_p}$ makes the transformation nearly energy-preserving, if it were not for the bounds on the t axis in model space. Besides having bounds, the t index is also discrete; in practice $u(p, t-ph)$ is found by linear interpolation of the adjacent t-sampled points of u. The transpose of L is simply a slant stack in the other domain with dips of the opposite sign:

$$u(p,t) = \mathbf{L}^T d(h,t) = \frac{1}{\sqrt{N_h}} \sum_{h}^{N_h} d(h,t+ph) \tag{14}$$

With the scale factor $1/\sqrt{N_h}$, equation (14) is not precisely the transpose of (13). The factors were chosen to preserve energy from one domain into another.

When the operator L (equation 13) is applied to the sparse model of figure 2(a) the resulting data is shown in figure 2(b). The data consists of a number of discrete events with linear moveout, each event possessing the corresponding wavelet found in the model of figure 2(a). The initial data for this test is that of figure 2(b) with a small amount of independent gaussian noise added to it (figure 3). The objective is to restore the original model of figure 2(a) from the noisy data. Selection of the model parameters $\sigma(u_i)$ is the first step.

Looking at the desired solution (figure 2(a)) it is apparent that it satisfies the mixed gaussian model of figure 1. A histogram of the model of figure 2(a), displayed in figure 4(a), bears this out. The low-end variance ${\sigma_0}^2$ is zero, while the variance ${\sigma_1}^2$ can easily be estimated. However, to make σ continuous in u, the function $\sigma(u)$ of figure 4(b) was chosen as the *a priori* information in the model domain. Figure 4(a) compares the fit of the continuous and discontinuous functions $\sigma(u)$ to the histogram of the model. It must be remembered that in this test case we are taking advantage of information that would not ordinarily be available. We have access to the statistics of the desired solution, and use them to select $\sigma(u)$.

One exception was made to the continuity requirement for σ. At very small values of $u(p,t)$, the variance is chosen to be zero. This will clip the low values of u to zero, and in

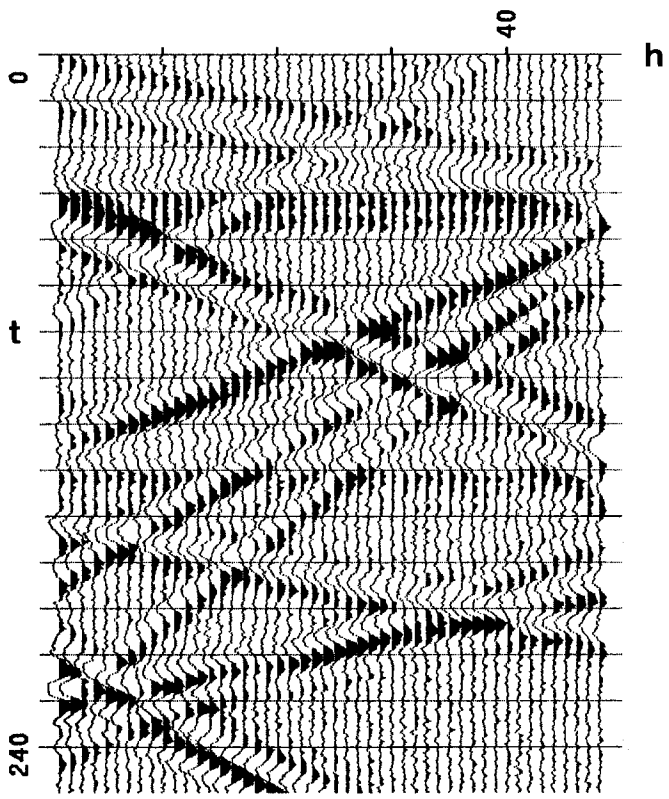

FIG. 3. The data of figure 2(b) with independent gaussian noise added. The variance of the data is 0.15^2. This is approximate, for the data is non-gaussian. The variance of the noise is 0.5^2.

the iterative algorithm to be described, effectively prevents those points from varying any further. The clip will gradually reduce the degrees of freedom of u in model space, because once a point is clipped to zero, it will remain zero. One measure of the progress of the algorithm is the ratio of zero points u to all points in model space. This is effectively a measure of the σ_0 population of **u**.

The next step is to choose a smoothing operator on **u**. The criterion is: what geometry of points in model space are expected to lie in the same population? In our case the answer is easy. The desired solution consists of time (t) sampled wavelets. Estimating $\sigma(u)$ is best done by averaging over the same population, or averaging in time. For this example, the algorithm applied a ten-point smoother on $|u|$ in the time direction and used the smoothed value of $|u|$ to select σ.

The simplest descent method was used to solve the nonlinear system (9): steepest descent. See figure 5 for a description of the algorithm. The only departure from linearity is

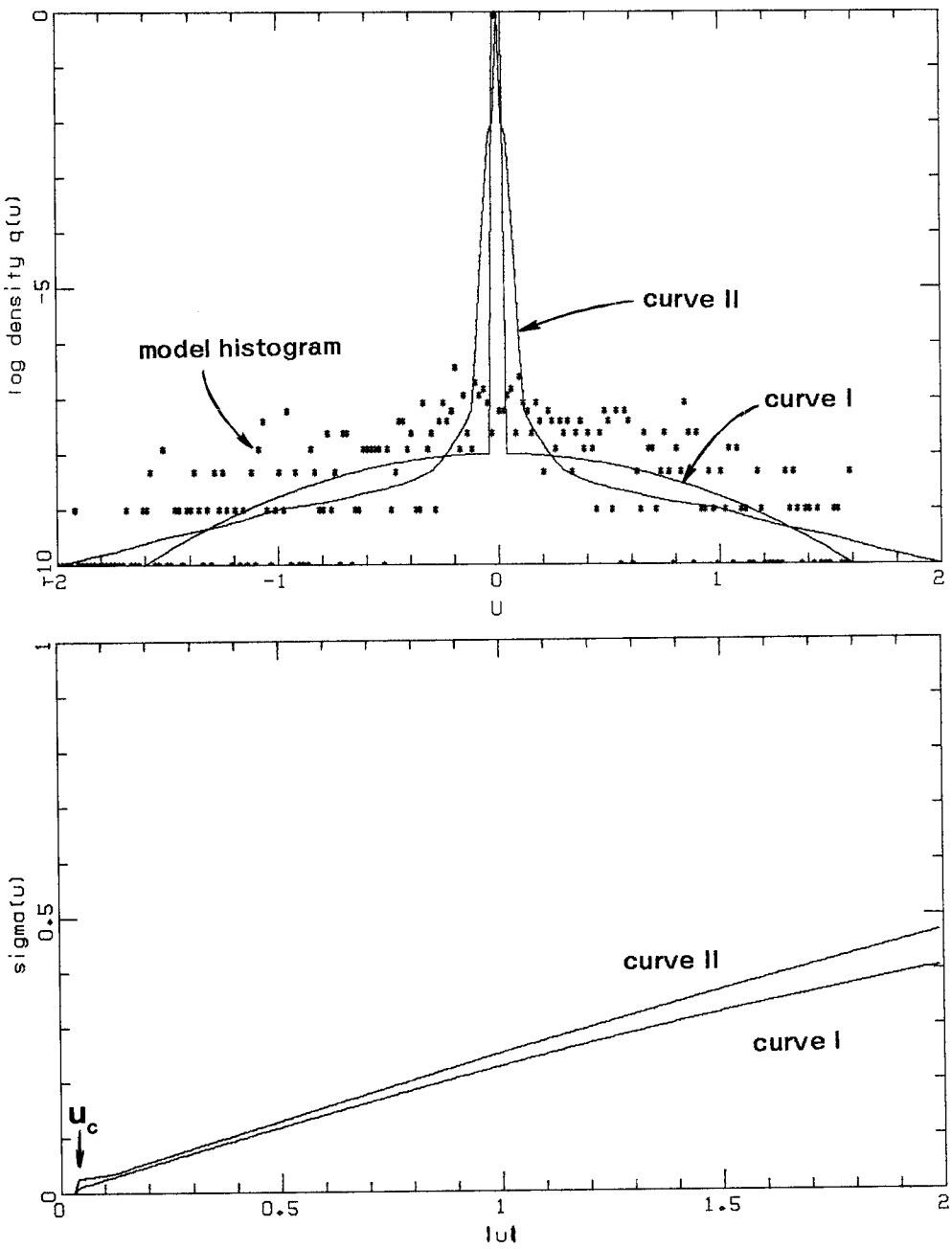

FIG. 4. Selection of the variance function $\sigma(u)$ for the test. Graph (b) is a plot of two possible functions to use for σ in the parsimony measure (P2).
Curve I is a member of the mixed-gaussian family depicted in figure 1, and actually represents $\sqrt{f(u)}$ rather than the actual variances σ_0 and σ_1. Curve I satisfies equation (6) with $\sigma_0 = 0$, $\sigma_1 = 0.8$, and $u_c = 0.035$. The corresponding log density is labeled as curve I on graph (a).
Curve II in (b) is the function actually used in the optimization. It is a piecewise linear function of $|u|$, and its log density is labeled as curve II on graph (a). It is continuous except for the clip imposed at the smallest values of $|u|$; see text for discussion of the clip.
The histogram of figure 2(a), the desired answer, is also displayed as the scattered points in graph (a). Curves I and II were designed to fit the histogram as well as possible.

Solve: $\mathbf{Qu} = \left(\mathbf{I} + \frac{\sigma^2(\mathbf{u})}{\sigma_n^2}\mathbf{L}^T\mathbf{L}\right)\mathbf{u} = \frac{1}{\sigma_n^2}\mathbf{L}^T\mathbf{d}$

Steepest Descent Algorithm

Initial $\mathbf{u}_0 = 0$

for $k = 0,1,2, \cdots$

estimate $\sigma(\mathbf{u}_k)$

$$\mathbf{g}_k = \left(\mathbf{u} + \frac{\sigma^2(\mathbf{u}_k)}{\sigma_n^2}\mathbf{L}^T(\mathbf{L}\mathbf{u}_k - \mathbf{d})\right)$$

$$\alpha_k = \frac{\mathbf{g}_k' \mathbf{g}_k}{\mathbf{g}_k' \mathbf{Q} \mathbf{g}'_k}$$

$$\mathbf{u}_{k+1} = \mathbf{u}_k - \alpha_k \mathbf{g}_k$$

next k

FIG. 5. Steepest descent algorithm. The modification to a linear steepest descent algorithm is in the estimation of σ at the beginning of each iteration. Otherwise it is the same, even the step size α has the same form as in the linear case.

the modification of $\sigma(u)$ at each step by the process of smoothing u and applying the function of figure 4(b). Each step can be considered one iteration of a corresponding linear system, with σ fixed. As a matter of fact, the process is expected to settle into a linear steepest descent as $1/\sigma^2$ in (P2) converges to an invariant weighting factor.

The initial estimate $\mathbf{u} = \mathbf{L}^T\mathbf{d}$ is shown in figure 6(a). It has all the artifacts inherent in inverse slant stacks. Streaks of energy at large constant dip represent truncation effects at the edge of the data grid. Figures 6(b), 6(c), and 6(d) display the fifth, tenth and last iterations respectively of the procedure. It is apparent that the method converges most rapidly in the first few iterations, then for the remaining iterations slowly chips away at the remainder of the error energy. The last iterate is very close in appearance to the desired solution (figure 2(a)).

Figure 7 is a plot of the number of zero points in $\mathbf{u}$ as a function of iteration number. This parameter converges to the value that the desired solution has. As explained in the section on parsimony, the zero count is a measure of the size of the σ_0 population of u. It is interesting to note that though the two-variance model of figure 1 was not used for the descent, the solution reached is in agreement with this model.

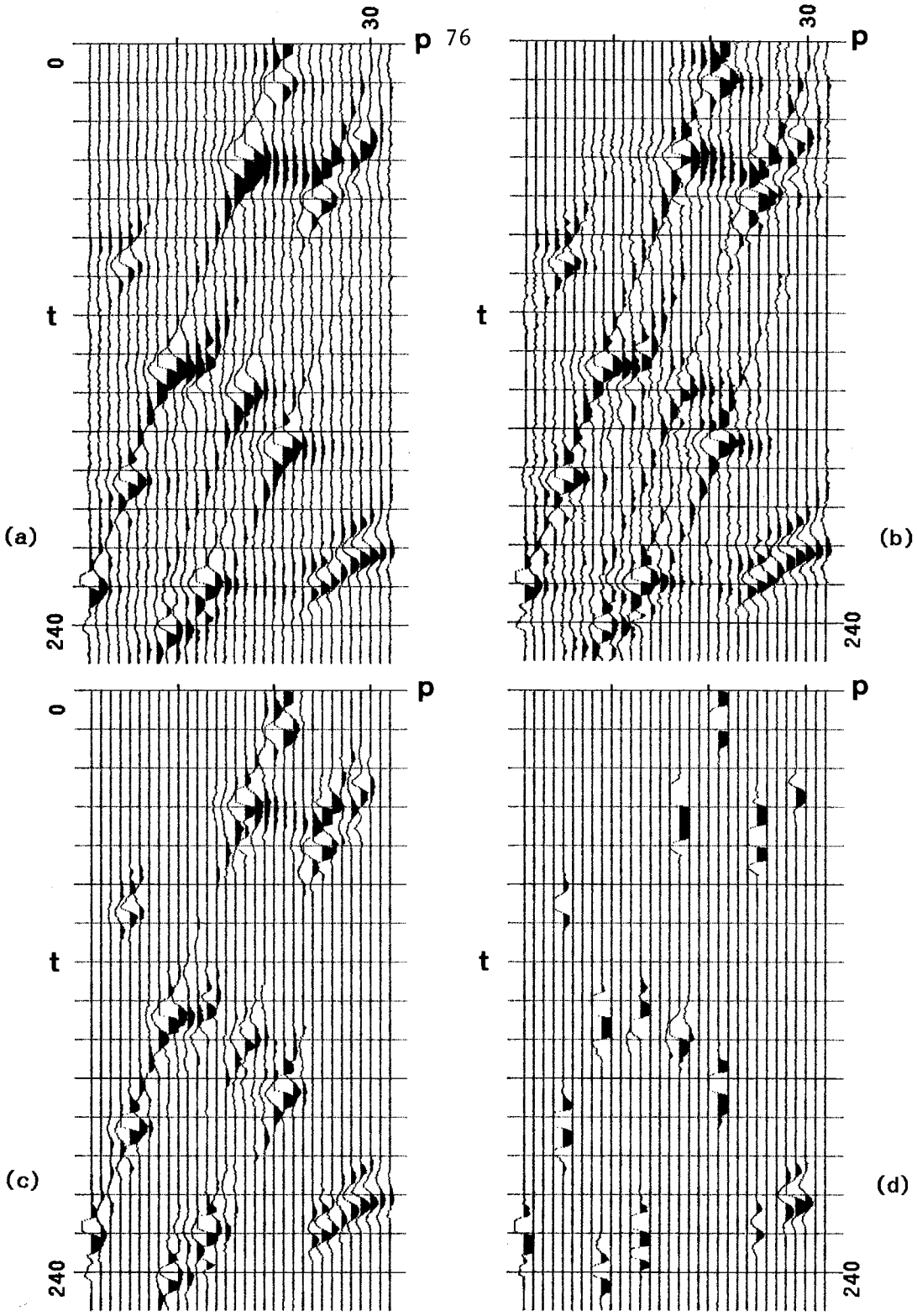

FIG. 6. (a) The initial estimate $\mathbf{u} = \mathbf{L}^T\mathbf{d}$. It has all the common artifacts seen on inverse slant stacks. The streaked events at high positive dip are truncation effects from the right edge of the data set (figure 3).
(b) Fifth iteration estimate of **u**.
(c) Tenth iteration estimate of **u**.
(d) Last (40th) iteration estimate of **u**. It has converged to the desired solution, with minor exceptions.

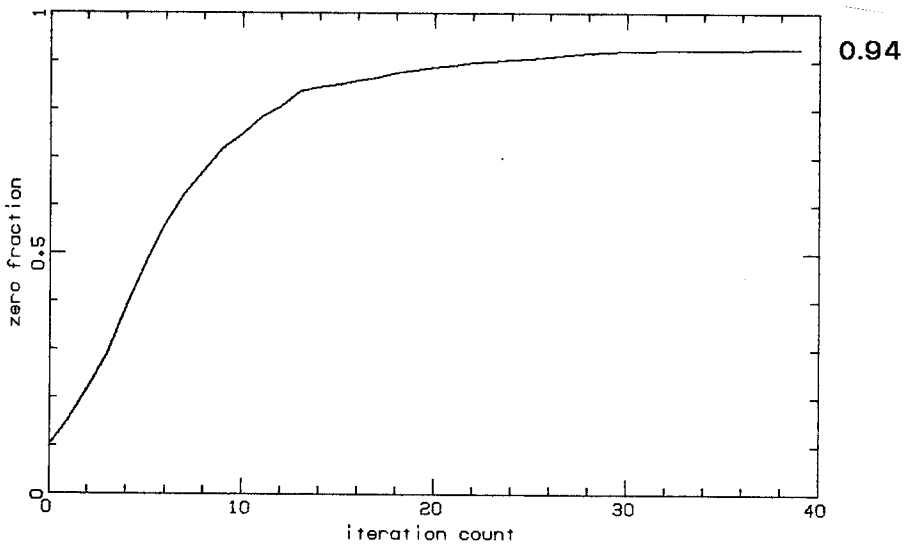

FIG. 7. Zero count of **u** in the model space as a function of iteration. The zero count converges to (and exceeds) the value of 0.94 given by the desired answer of figure 2(a). The zero count is the proportion of identically zero points u_i to the number of points $N = n_t n_p$ in model space.

Because of the biased nature of the functional (P2), the energy in the last iterate of **u** is slightly smaller than in the original model. If the solved model **u** were subsequently applied to restore missing traces in **d**, the restored traces may have to be scaled to the original data.

REFERENCES

Deeming, T., 1981, Deconvolution and reflection coefficient estimation using a generalized minimum entropy principle: Paper presented at the 51st annual SEG meeting, Los Angeles, October 1981.

Godfrey, R., 1979, A stochastic model for seismogram analysis: Doctoral thesis, Stanford University, August 1979. Also SEP volume 17.

Rocca, F., 1982, "Grass and trees", personal communication.

Thorson, J., 1982, Weighting and extrapolation schemes for stacking: SEP volume 30, pp. 77-94.

47. A and B are both knights and C is a knave. Therefore, C is the werewolf.

48. A is the only knight and thus the werewolf.

49. The portrait is in the silver casket.

51. The portrait is in the lead casket.

50. 64 and 15,625.

52.

Center field: Jake	Left field: Mike	Right field: Luke
Third base: Ken	Second base: Oakie	First base: Punky
Shortstop: Quick	Pitcher: Nick	Catcher: Rick

53. Saucer 54. Six 55. Five bananas.

56. The numbers one through nine are embedded in the names in numerical order.

57. Zero. If three letters match the envelopes, so will the fourth.

58. The boy maximizes his chance of drawing a ten-dollar bill by putting a single ten-dollar bill in one hat and the rest of the bills in the other hat. His overall chance of success is 14/19.

59. Let 1A stand for the insides of the first pair of gloves, 1B for the outsides, 2A for the insides of the second pair of gloves, and 2B for the outsides. Dr. X wears both pairs, the second on top of the first. Sides 1A and 2B may become contaminated. Sides 1B and 2A remain sterile. Dr. Y wears the second pair, with the sterile sides 2A touching his hands. Dr. Z turns the first pair inside out before putting them on. Sterile sides 1B will then be touching his hands. He then puts on the second pair of gloves with 2B to the outside. In this way, sides 2B are the only sides that touch Ms. H.

60. Number each bag from one to ten. From bag one, take out one coin. From bag two, take two coins. From bag three, take out three coins, and so on. Place all of the coins on the penny scale and weigh all of them at once. If all of the coins were made of fool's gold, then they would weigh 110 ounces. If bag one contained the real gold coins, they would all weigh 111 ounces. If bag two contained the real gold coins, they would all weigh 112 ounces. The number of the bag containing the real gold coins would be the difference between the weight on the scale and 110 ounces.

61. 27. 62. Forty.

63. Each of the numbers contains an 'i' in their names. The next number is 31.

64. N. The letters are the first letters of the digits from 0 to 9.

65. Eleven crossings:

1) A and his wife cross,	2) A returns,	3) B's and C's wives cross,
4) A's wife returns,	5) B and C cross,	6) B and his wife return,
7) A and B cross,	8) C's wife returns,	9) A's and B's wives cross,
10) C return,	11) C and his wife return.	