Stanford Exploration Project, Report 105, September 5, 2000, pages 41–53

Stanford Exploration Project, Report 105, September 5, 2000, pages 41–53

Short Note

Multiple realizations using standard inversion techniques

Robert G. Clapp[1]

INTRODUCTION

When solving a missing data problem, geophysicists and geostatisticians have very similar strategies. Each use the known data to characterize the model's covariance. At SEP we often characterize the covariance through Prediction Error Filters (PEFs) (Claerbout, 1999). Geostatisticians build variograms from the known data to represent the model's covariance (Isaaks and Srivastava, 1989). Once each has some measure of the model covariance they attempt to fill in the missing data. Here their goals slightly diverge. The geophysicist solves a global estimation problem and attempts to create a model whose covariance is equivalent to the covariance of the known data. The geostatistician performs kriging, solving a series of local estimation problem. Each model estimate is the linear combination of nearby data points that best fits their predetermined covariance estimate. Both of these approaches are in some ways exactly what we want: given a problem give me 'the answer'.

The single solution approach however has a couple significant drawbacks. First, the solution tends to have low spatial frequency. Second, it does not provide information on model variability or provide error bars on our model estimate. Geostatisticians have these abilities in their repertoire through what they refer to as 'multiple realizations' or 'stochastic simulations'. They introduce a random component, based on properties (such as variance) of the data, to their estimation procedure. Each realization's frequency content is more accurate and by comparing and contrasting the equiprobable realizations, model variability can be assessed.

In this paper I present a method to achieve the same goal using a formulation that better fits into geophysical techniques. I modify the model styling goal, replacing the zero vector with a random vector. I show how the resulting models have a more pleasing texture and can provide information on variability.

[1]**email:** bob@sep.stanford.edu

MOTIVATION

Regularized linear least squares estimation problems can be written as minimizing the quadratic function

$$Q(m) = \|\mathbf{d} - \mathbf{Lm}\|^2 + \epsilon^2 \|\mathbf{Am}\|^2 \tag{1}$$

where **d** is our data, **L** is our modeling operator, **A** is our regularization operator and we are inverting for a model **m**. Alternately, we can write them in terms of fitting goals,

$$\begin{array}{rcl} \mathbf{d} & \approx & \mathbf{Lm} \\ \mathbf{0} & \approx & \epsilon \mathbf{Am}, \end{array} \tag{2}$$

For the purpose of this paper I will refer to the first goal as the *data fitting goal* and the second as the *model styling goal*. Normally we think of *data fitting goal* as describing the physics of the problem. The *model styling goal* is suppose to provide information about the model character. Ideally **A** should be the inverse model covariance. In practice we don't have the model covariance so we attempt to approximate it through another operator. At SEP the regularization operator is typically one of the following:

Laplacian or gradient a simple operator that assumes nothing about the model

Prediction Error Filter (PEF) a stationary operator estimated from known portions of the model or some field with the same properties as the model (Claerbout, 1999)

steering filter a non-stationary operator built from minimal information about the model (Clapp et al., 1997)

non-stationary PEF a non-stationary operator built from a field with the same properties as the model (Crawley, 2000).

A problem with the first three operators is that while they approximate the model covariance, they have little concept of model variance. As a result our model estimates tend to have the wrong statistical properties.

MISSING DATA

The missing data problem is probably the simplest to understand and interpret results. We begin by binning our data onto a regular mesh. For **L** in fitting goals (2) we will use a selector matrix **J**, which is '1' at locations where we have data and '0' at unknown locations. As an example, let's try to interpolate a day's worth of data collected by SeaBeam (Figure 1), which measures water depth under and to the side of a ship (Claerbout, 1999). Figure 2 shows the result of estimating a PEF from the known data locations and then using it to interpolate the entire mesh. Note how the solution has a lower spatial frequency as we move away from the recorded data. In addition, the original tracks of the ship are still clearly visible.

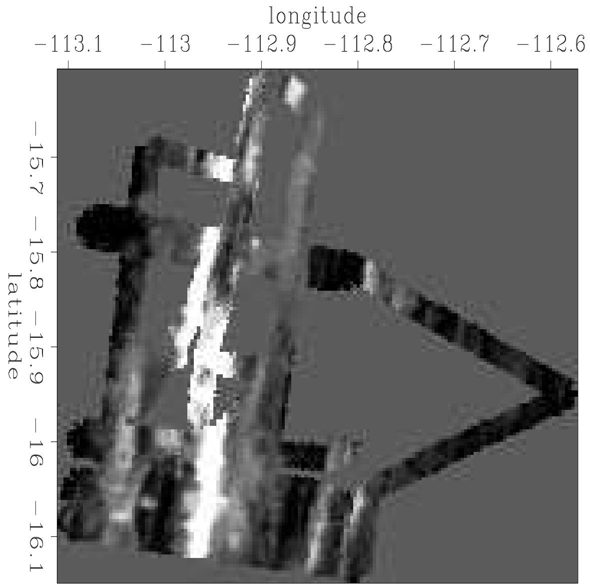

Figure 1: Depth of the ocean under ship tracks. bob3-sea.init [ER]

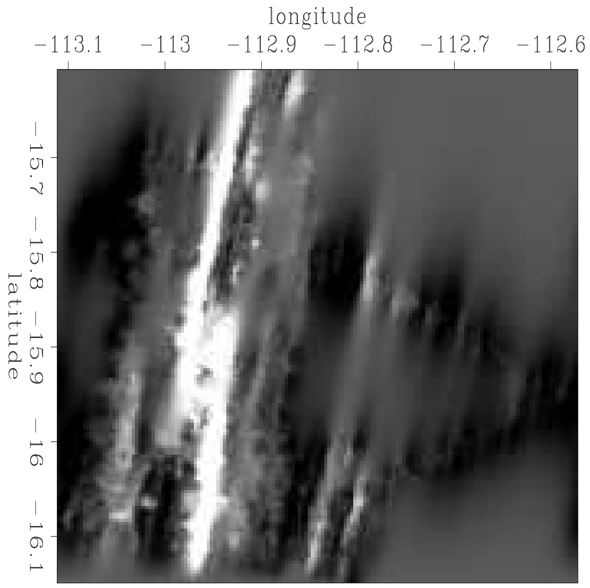

Figure 2: Result of using a PEF to interpolate Figure 1, taken from GEE (Claerbout, 1999). bob3-sea.pef [ER,M]

If we look at a histograms of the known data and our estimated data we can see the effect of the PEF. The histogram of the known data has a nice Gaussian shape. The predicted data is much less Gaussian with a much lower variance. We want estimated data to have the same statistical properties as the known data (for a Gaussian distribution this means matching the mean and variance).

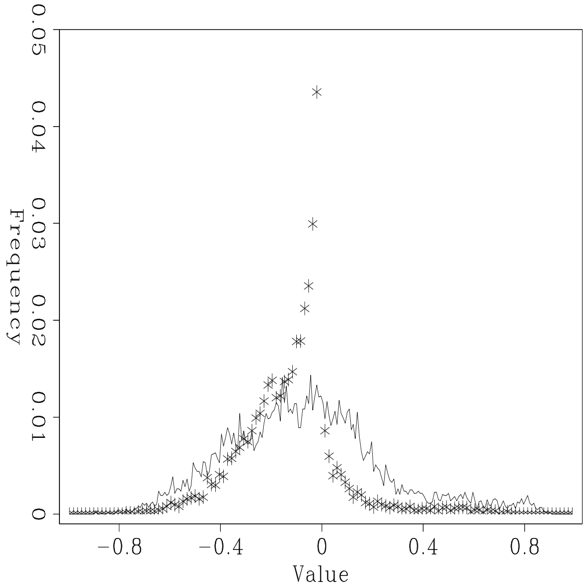

Figure 3: Histogram for the known data (solid lines) and the estimated data ('*'). Note the dissimilar shapes. bob3-pef.histo [ER,M]

Geostatisticians are confronted with the same problem. They can produce smooth, low frequency models through kriging, but must add a little twist to get model with the statistical properties as the data. To understand how, a brief review of kriging is necessary. Kriging estimates each model point by a linear combination of nearby data points. For simplicity lets assume that the data has a standard normal distribution. The geostatistician find all of the points $m_1....m_n$ around the point they are trying to estimate m_0. The vector distance between all data points $\mathbf{d_{ij}}$ and each data point and the estimation point $\mathbf{d_{i0}}$ are then computed. Using the predefined covariance function estimate C, a covariance value is then extracted between all known point pairs C_{ij} and between known points and estimation point C_{i0} at the given distances $\mathbf{d_{ij}}$ and $\mathbf{d_{i0}}$ (Figure 4). They compute the weights $(w_1...w_n)$ by solving the set of equations implied by

$$\begin{bmatrix} C_{11} & ... & C_{1n} & 1 \\ . & ... & . & . \\ . & ... & . & . \\ . & ... & . & . \\ C_{n1} & ... & C_{nn} & 1 \\ 1 & ... & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ . \\ . \\ . \\ w_n \\ \mu \end{bmatrix} = \begin{bmatrix} C_{10} \\ . \\ . \\ . \\ C_{n0} \\ 1 \end{bmatrix}. \tag{3}$$

Estimating m_0 is then simply,

$$m_0 = \sum_{i=1}^{n} w_i m_i. \tag{4}$$

To guarantee that the matrix in equation (3) is invertible geostatisticians approximate the covariance function through a linear combination of a limited set of functions that guarantee that the matrix in equation (3) is positive-definite and therefore invertible. The smooth models

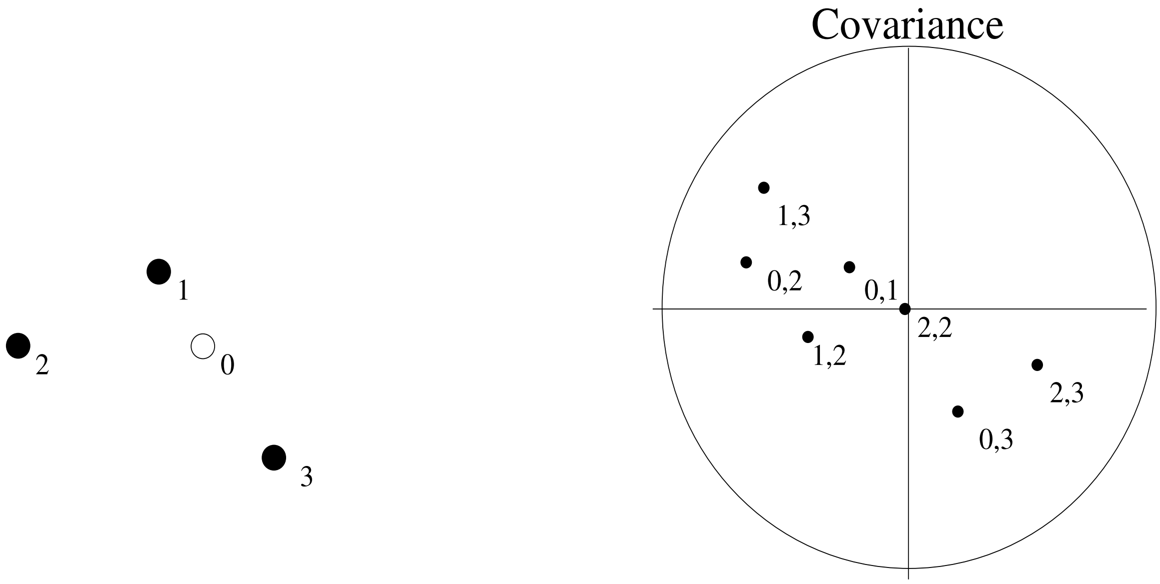

Figure 4: Definition of the terms in equation (3). A vector is drawn between two points. The covariance at the angle and distance describing the vector is then selected. bob3-covar-def [NR]

provided by kriging often prove to be poor representations of earth properties. A classic example is fluid flow where kriged models tend to give inaccurate predictions. The geostatistical solution is to perform Gaussian stochastic simulation, rather than kriging, to estimate the field (Deutsch and Journel, 1992). There are two major differences between kriging and simulation. The primary difference is that a random component is introduced into the estimation process. Stochastic simulation, or sequential Gaussian simulation, begins with a random point being selected in the model space. They then perform kriging, obtaining a kriged value m_0 and a kriging variance σ_k. Instead of using m_0 for the model value we select a random number β from a normal distribution. We use as our model point estimate m_i,

$$m_i = m_0 + \sigma_k \beta. \tag{5}$$

We then select a new point in the model space and repeat the procedure. To preserve spatial variability, a second change is made: all the previously estimated points are treated as 'data' when estimating new points guaranteeing that the model matches the covariance estimate. By selecting different random numbers (and/or visiting model points in a different order) we will get a different, equiprobable model estimate. The advantage of the models estimated through simulation is that they have not only the covariance of they data, but also the variance. As a result the models estimated by simulation give more realistic fluid flow measurements compared to a kriged model. In addition, by trying different realizations fluid flow variability can be assessed.

The difference between kriging and simulation has a corollary in our least squares estimation problem. To see how let's write our fitting goals in a slightly different format,

$$\begin{aligned} \mathbf{r_d} &\approx \mathbf{d} - \mathbf{J}\mathbf{m} \\ \mathbf{r_m} &\approx \epsilon \mathbf{A}\mathbf{m}, \end{aligned} \tag{6}$$

where $\mathbf{r_d}$ is our data residual and $\mathbf{r_m}$ is our model residual. The model residual is the result of applying our covariance estimate $\mathbf{A}$ to our model estimate. The larger the value of a given $\mathbf{r_m}$, the less that model point makes sense with its surrounding points, given our idea of covariance. This is similar to kriging variance. It follows that we might be able to obtain something similar to the geostatistician's simulations by rewriting our fitting goals as

$$\begin{aligned} \mathbf{d} &\approx \mathbf{J}\mathbf{m} \\ \sigma \mathbf{v} &\approx \epsilon \mathbf{A}\mathbf{m}, \end{aligned} \tag{7}$$

where $\mathbf{v}$ is a vector of random normal numbers and σ is a measure of our estimation uncertainty[2]. By adjusting σ we can change the distribution of $\mathbf{m}$. For example, let's return to the SeaBeam example. Figure 5 shows four different model estimations using a normal distribution and various values for the variance. Note how the texture of the model changes significantly. If we look at a histogram of the various realizations (Figure 6), we see that the correct distribution is somewhere between our second and third realization.

We can get an estimate of σ, or in the case of the missing data problem $\frac{\sigma}{\epsilon}$, by applying fitting goals (6). If we look at the variance of the model residual $\sigma(\mathbf{m_r})$ and $\sigma(\mathbf{d})$ we can get a good estimate of σ,

$$\sigma = \frac{\sigma(\mathbf{r_m})}{\sigma(\mathbf{d})}. \tag{8}$$

Figure 7 shows eight different realizations with a random noise level calculated through equation (8). Note how we have done a good job emulating the distribution of the known data. Each image shows some similar features but also significant differences (especially note within the 'V' portion of the known data).

A potentially attractive feature of setting up the problem in this manner is that it easy to have both a space-varying covariance function (a steering filter or non-stationary PEF) along with a non-stationary variance. Figure 8 shows the SeaBeam example again with the variance increasing from left to right.

SUPER DIX

In general the operator $\mathbf{L}$ in fitting goals (7) is much more complex than the simple masking operator used in the missing data problem. One of the most attractive potential uses for a

[2]For the missing data problem ϵ could be used exclusively. As our data fitting goal becomes more complex, having a separate σ and ϵ becomes useful.

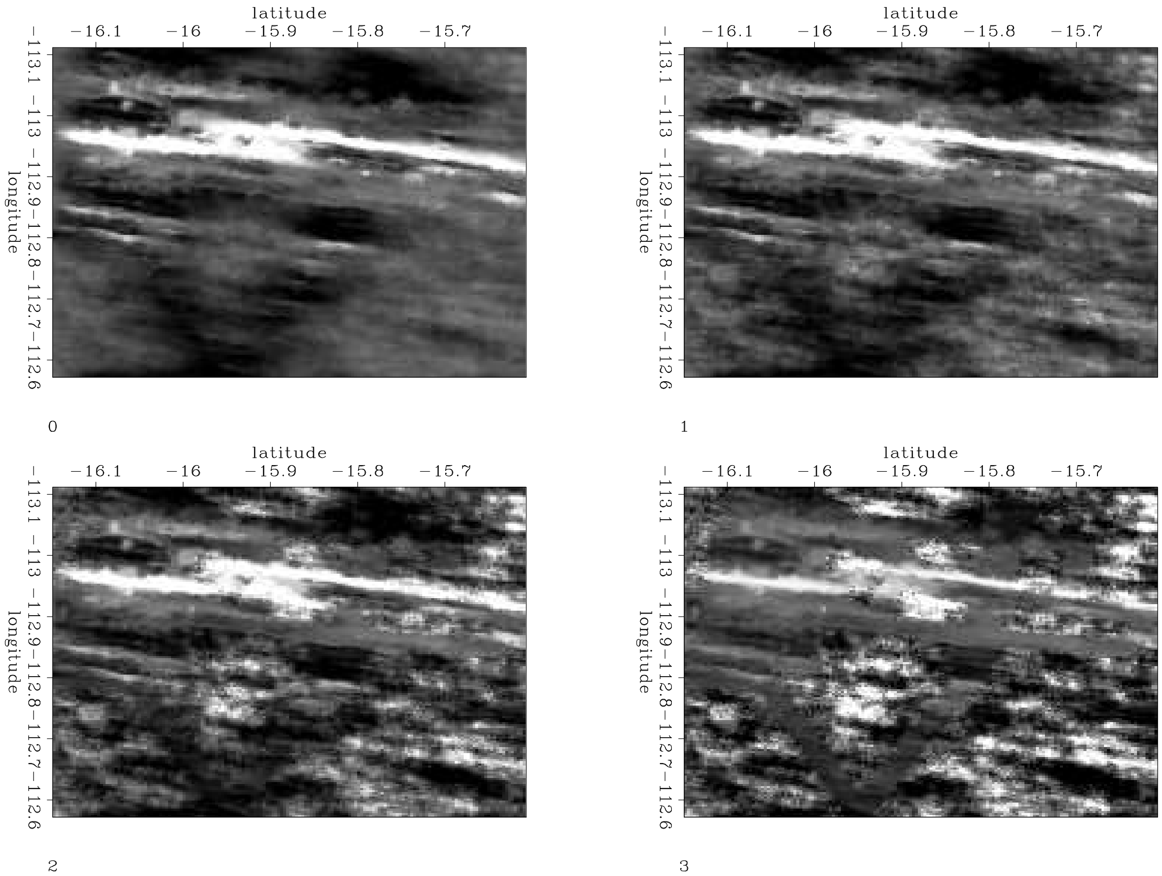

Figure 5: Four different realizations with increasing σ in fitting goals (7). bob3-distrib [ER,M]

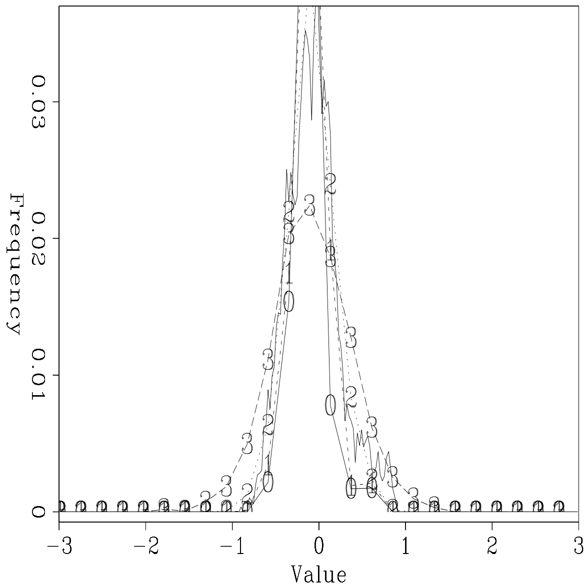

Figure 6: Histogram of the known data (solid line) and the four different realizations of Figure 5. bob3-movie.distir [ER,M]

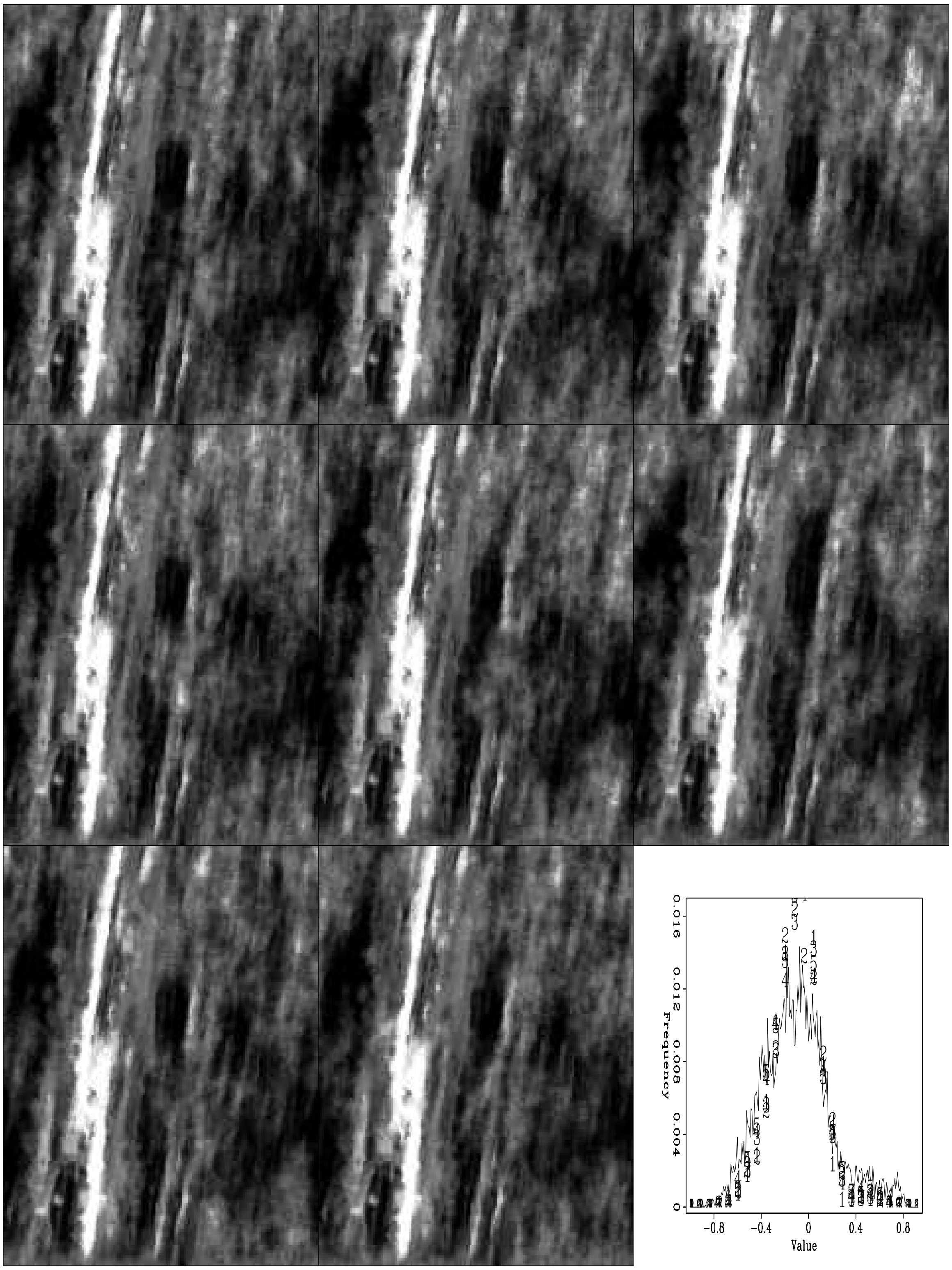

Figure 7: Eight different realizations of the SeaBeam interpolation problem and their histograms. Note how the realizations vary away from the known data points. bob3-sea.movie [ER,M]

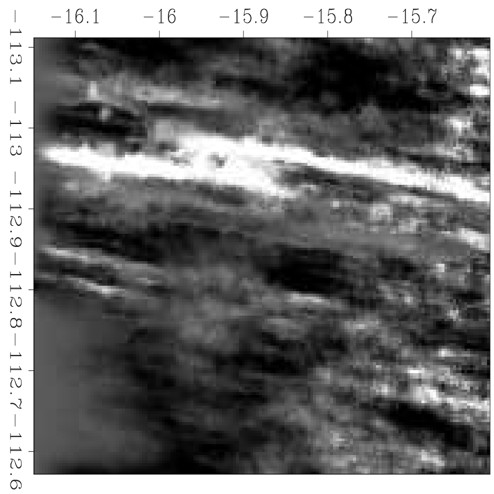

Figure 8: Realization where the variance added to the image increases from left to right. bob3-non-stat [ER,M]

range of equiprobable models is in velocity estimation. As a result I decided to next test the methodology on one of the simplest velocity estimation operators, the Dix equation (Dix, 1955).

Following the methodology of Clapp et al. (1998), I start from a CMP gather $q(t,i)$ moveout corrected with velocity v. A good starting guess for our RMS velocity function is the maximum "instantaneous stack energy",

$$\text{stack}(t,v) = \sum_{i=0}^{n} \text{NMO}_\text{v}(q(t,i)). \tag{9}$$

Not all times have reflections so we don't weight each $v_{\text{rms}}(t)$ equivalently. Instead we introduce a diagonal weighting matrix, $\mathbf{W}$, found from stack energy at each selected $v_{\text{rms}}(t)$. Our data fitting goal becomes

$$\mathbf{0} \quad \approx \quad \mathbf{W}[\mathbf{Cu}-\mathbf{d}]. \tag{10}$$

We are multiplying our RMS function by our time τ so must make a slight change in our weighting function. To give early times approximately the same priority as later times, we need to multiply our weighting function by the inverse,

$$\mathbf{W}' = \frac{\mathbf{W}}{\tau}. \tag{11}$$

Next we need to add in regularization. I define a steering filter operator $\mathbf{A}$ that influences the model to introduce velocity changes that follow structural dip. I replace the zero vector with a random vector and precondition the problem (Fomel et al., 1997) to get

$$\begin{aligned} \mathbf{0} &\approx \mathbf{W}'(\mathbf{CA}^{-1}\mathbf{p}-\mathbf{d}) \\ \sigma\mathbf{v} &\approx \epsilon\mathbf{p}. \end{aligned} \tag{12}$$

To test the methodology I took a 2-D line from a 3-D North Sea dataset provided by Unocal. Figure 9 shows four different realizations with varying levels of σ. I then chose

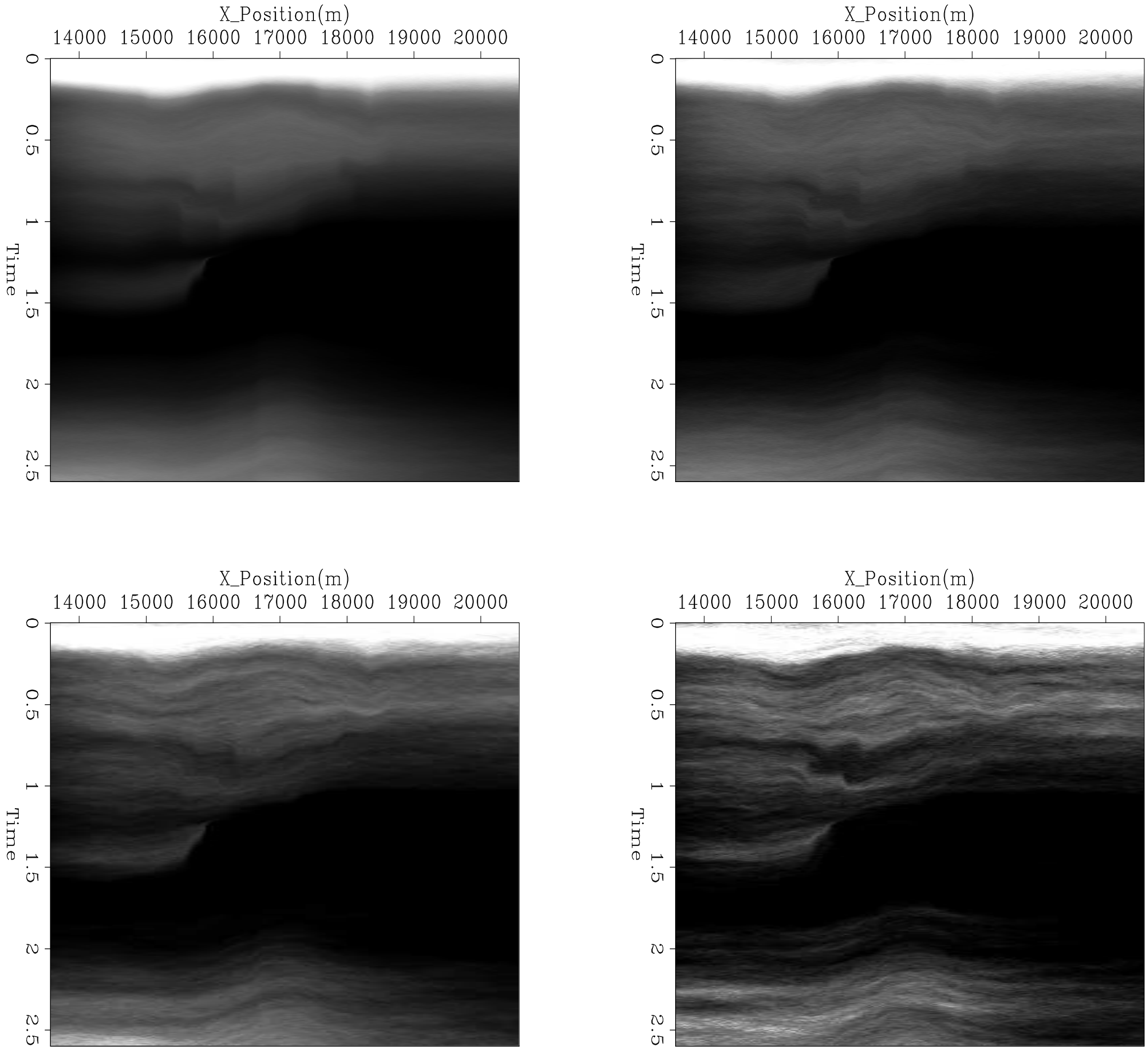

Figure 9: Four different realization of fitting goals (12) with increasing levels of Gaussian noise in **v**. bob3-scale.10.x2 [ER,M]

what I considered a reasonable variability level, and constructed ten equiprobable models (Figure 10). Note that the general shapes of the models are very similar. What we see are smaller structural changes. For example, look at the range between .7s and 1.1s. Generally each realization tries to put a high velocity layer in this region, but thickness and magnitude varies in the different realizations.

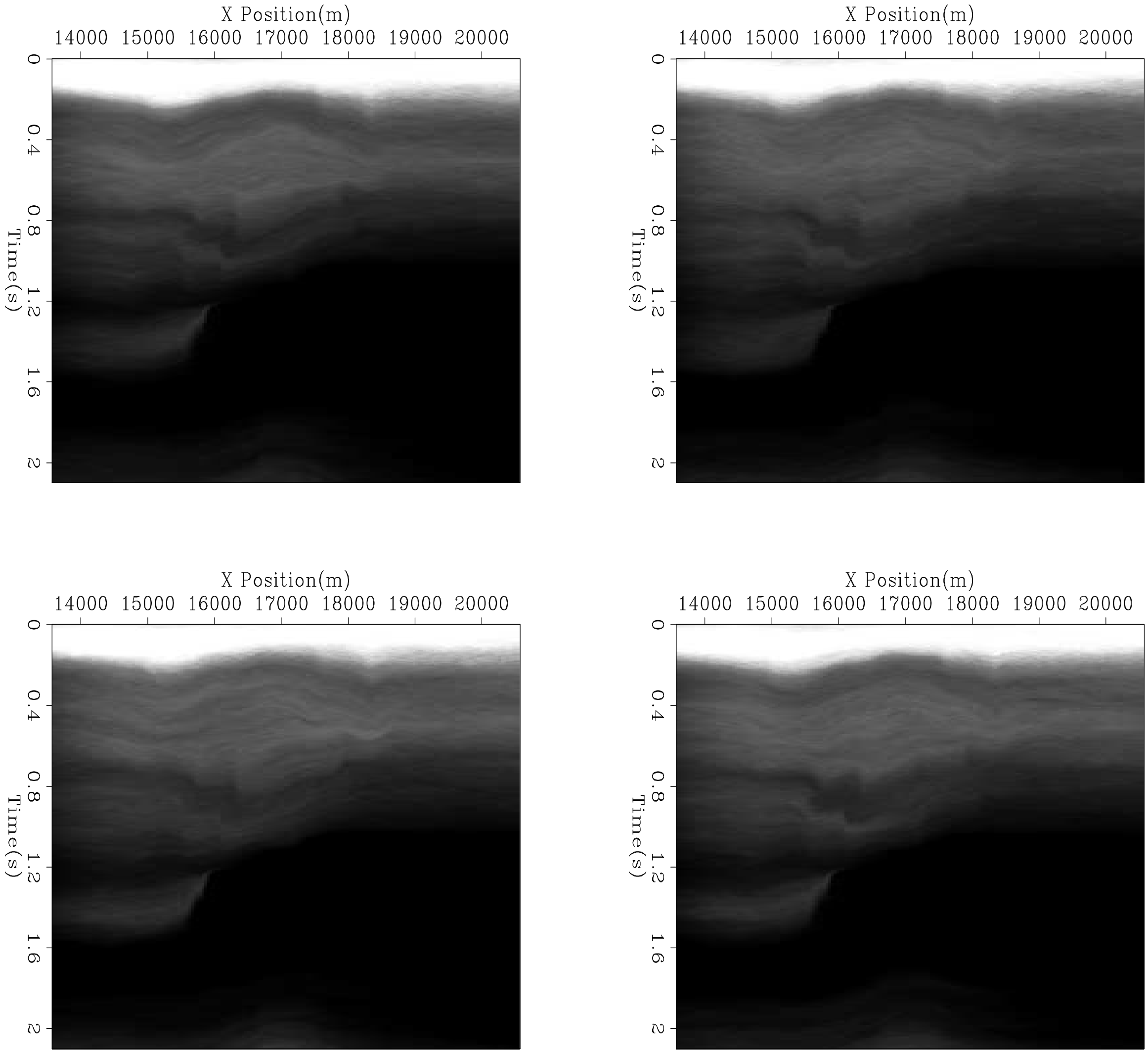

Figure 10: Four of the ten different realization of fitting goals (12) with constant Gaussian noise in **v**. bob3-dix-real [ER,M]

FUTURE WORK

In this paper I glossed over several problems. First, σ should be a space-varying function rather than the constant I proposed. A bootstrap approach (using the model residual at one non-linear iteration as our guess at a space-varying σ) might prove effective but hasn't been tested. How to calculate σ for the non-missing data problem is an open question. In the generic geophysical operator **L**, we often don't know which model components are estimated through the data fitting goal and which are estimated by the model styling goal. Finally, I made the assumption that I was dealing with models with a normal distribution. Whether replacing **v** with another distribution or using something similar to the geostatistician's *normal-score transform* would be effective in correctly modeling these distributions is unknown.

CONCLUSIONS

I have demonstrated a new method for creating equiprobable realizations using standard geophysical inversion techniques. The character of resulting models is much more consistent than models derived by standard techniques.

REFERENCES

Claerbout, J., 1999, Geophysical estimation by example: Environmental soundings image enhancement: Stanford Exploration Project, http://sepwww.stanford.edu/sep/prof/.

Clapp, R. G., Fomel, S., and Claerbout, J., 1997, Solution steering with space-variant filters: SEP–**95**, 27–42.

Clapp, R. G., Sava, P., and Claerbout, J. F., 1998, Interval velocity estimation with a null-space: SEP–**97**, 147–156.

Crawley, S., 2000, Seismic trace interpolation with nonstationary prediction-error filters: SEP–**104**.

Deutsch, C. V., and Journel, A., 1992, GSLIB: Geostatistical software library and user's guide: Oxford University Press.

Dix, C. H., 1955, Seismic velocities from surface measurements: Geophysics, **20**, no. 1, 68–86.

Fomel, S., Clapp, R., and Claerbout, J., 1997, Missing data interpolation by recursive filter preconditioning: SEP–**95**, 15–25.

Isaaks, E. H., and Srivastava, R. M., 1989, An Introduction to Applied Geostatistics: Oxford University Press.