Solution steering with space-variant filters[a]

[a]Published in SEP report, 95, 27-42 (1997)

Robert G. Clapp, Sergey Fomel, and Jon Claerbout[1]

ABSTRACT

Most geophysical problem require some type of regularization. Unfortunately most regularization schemes produce "smeared" results that are often undesirable when applying other criteria (such as geologic feasibility). By forming regularization operators in terms of recursive steering filters, built from a priori information sources, we can efficiently guide the solution towards a more appealing form. The steering methodology proves effective in interpolating low frequency functions, such as velocity, but performs poorly when encountering multiple dips and high frequency data. Preliminary results using steering filters for regularization in tomography problems are encouraging.

INTRODUCTION

When attempting to do inversion we are constantly confronted with the problem of slow convergence. Claerbout and Nichols (1994) suggested using a preconditioner to speed up convergence. Unfortunately it is often difficult to find an appropriate preconditioner and/or the preconditioner is so computationally expensive that it negates the savings gained by reducing the number of iterations (Claerbout, 1994). Claerbout (1997) proposed designing helicon-style operators to provide a method to find stable inverses, and potentially, appropriate preconditioners (Fomel et al., 1997; Fomel, 1997).

In addition, geophysical problems are often under-determined, requiring some type of regularization. Unfortunately the simplest, and most common, regularization techniques tend to create isotropic features when we would often prefer solutions that follow trends. This problem is especially prevalent in velocity estimation. The result obtained through many inversion schemes produce a velocity structure that geologists (whose insights are hard to encode into the regression equations) find unreasonable (Etgen, 1997). Fortunately, there are often other sources of information that can be encoded into the regularization operator that allow the inversion to be guided towards a more appealing result. For example, in the case of velocity estimation, reflector dips might be appropriate.

[1]**e-mail:** bob@sep.stanford.edu, sergey@sep.stanford.edu, jon@sep.stanford.edu

We create small, space-variant, steering filters from dip or other a priori information. We use the inverse of these filters to form a preconditioner which acts as our regularization operator. We show this methodology applied to three different types of problems. In the first set of examples we interpolate well-log information using reflector dip as the basis for our steering filters. For the second set of examples we do a more traditional seismic data interpolation problem. Starting from a shot gather with a portion of the data missing. We use a velocity function to create hyperbolic paths, which in turn are used to construct steering filters. In the final example we show some preliminary results of using steering filters in conjunction with a tomographic operator to create velocity models which both satisfy the data and are geologically reasonable.

THEORY/MOTIVATION

Regularization

In general, geophysical problems are ill-posed. To obtain pleasing results we impose some type of regularization criteria such as diagonal scaling, limiting solutions to large singular values (Clapp and Biondi, 1995), or minimizing different solution norms (Nichols, 1994). The typical SEP approach is to minimize the power out of a regularization operator ($\mathbf{A}$) applied to the model ($\mathbf{m}$), described by the fitting goal

$$0 \approx \mathbf{A}\mathbf{m}. \tag{1}$$

Where $\mathbf{A}$'s spectrum will be the inverse of $\mathbf{m}$, so to produce a smooth $\mathbf{m}$, we need a rough $\mathbf{A}$(Claerbout, 1994)). The regularization operator can take many forms, in order of increasing complexity:

Laplacian operator (∇^2) The symmetric nature of the Laplacian leads to isotropic smoothing of the image.

Steering filters Simple plane wave annihilation filters which tend to orient the data in some preferential direction, chosen a priori. These filters can be simple two point filters, Figure 1, to larger filters that sacrifice compactness for more precise dip annihilation.

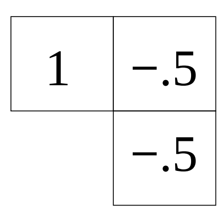

Figure 1: An example of steering filter. In this case preference is given to slopes at 45 degrees.

Prediction Error Filters (PEF) Like steering filters apply a preferential smoothing direction, but are not limited to a single dip and determine their smoothing directions from the known data (Schwab, 1997).

Preconditioning

Another important consideration is the speed of convergence of the problem. The size of most geophysical problems make direct matrix inversion methods impractical. An appealing alternative for linear problems is the family of conjugate gradient methods. Unfortunately, the operators used in seismic reflection problems are often computationally expensive. As a result it is important to minimize the number of steps it takes to get to a reasonable solution. One way that can reduce the number of iterations is by reformulating the problem in terms of some new variable ($\mathbf{x}$) with a preconditioning operator ($\mathbf{B}$). Changing a tradition inversion problem where the operator ($\mathbf{C}$) maps the model ($\mathbf{m}$) to the data ($\mathbf{d}$),

$$\mathbf{d} \approx \mathbf{C}\mathbf{m} \tag{2}$$

we can rewrite

$$\mathbf{d} \approx \mathbf{C}\mathbf{B}\mathbf{x} \tag{3}$$

where

$$\mathbf{m} = \mathbf{B}\mathbf{x}. \tag{4}$$

Helix transform

The next question is how to choose $\mathbf{B}$? We have three general requirements:

- it produces relatively smooth (by some criteria) results;
- it spreads information quickly;
- and it is computationally inexpensive.

By defining our operators via the helix method (Claerbout, 1997) we can meet all of these requirements. The helix concept is to transform N-Dimensional operators into 1-D operators to take advantage of the well developed 1-D theory. In this case we utilize our ability to construct stable inverses from simple, causal filters. We can set $\mathbf{B}$, from equation (4) to

$$\mathbf{B} = \mathbf{A}^{-1}, \tag{5}$$

where $\mathbf{A}$ is the roughening operator from fitting goal (1), and $\mathbf{B}$ is simulated using polynomial division. If $\mathbf{A}$ is a small roughening operator, $\mathbf{B}$ is a large smoothing operator without the heavy costs usually associated with larger operators.

Steering Filters

At this point a discussion of steering filters is appropriate. Plane waves with a given slope on a discrete grid can be predicted (destroyed) with compact filters (Schwab, 1997). Inverting such a filter by the helix method, we can create a signal with a given arbitrary slope extremely quickly. If this slope is expected in the model, the described procedure gives us a very efficient method of preconditioning the model estimation problem, fitting goal (2).

How can a plane prediction (steering) filter be created? On the helix surface, the plane wave $P(t,x) = f(t - px)$ translates naturally into a periodic signal with the period of $T = N_t + \sigma$, where N_t is the number of points on the t trace, and $\sigma = \frac{p\triangle x}{\triangle t}$, where σ is the plane slope, [2] and $\triangle x$ and $\triangle t$ correspond to the mesh size. If we design a filter that is two columns long (assuming the columns go in the t direction), then the *plane prediction* problem is simply connected with the *interpolation* problem: to destroy a plane wave, shift the signal by T, interpolate it, and subtract the result from the original signal. Therefore, we can formally write

$$\mathbf{P} = I - \mathbf{S}(\sigma) \, , \tag{6}$$

where $\mathbf{P}$ denotes the steering filter, $\mathbf{S}$ is the shift-and-interpolation operator, and $\mathbf{I}$ is the identity operator.

Different choices for the operator $\mathbf{S}$ in (6) produce filters with different length and prediction power. A shifting operation corresponds to the filter with the Z-transform $\Sigma(Z) = Z^T$, while the operator $\mathbf{S}$ corresponds to an approximation of $\Sigma(Z)$ with integer powers of Z. One possible approach is to expand $\Sigma(Z) Z^{-N_t}$ using the Taylor series around the zero frequency ($Z = 1$). For example, the first-order approximation is

$$S_1(Z) = Z^{N_t}\left(1 + \sigma(Z-1)\right) = (1-\sigma) Z^T + \sigma Z^{T+1} \, , \tag{7}$$

which corresponds to linear interpolation and leads in the two-dimensional space to the steering filter $\mathbf{P}$ of the form

1	
$\sigma - 1$	$-\sigma$

(8)

Filter (8) is equivalent to the explicit first-order upwind finite-difference scheme on the plane wave equation

$$\frac{\partial P}{\partial x} + p\,\frac{\partial P}{\partial t} = 0 \, . \tag{9}$$

An important property of filter (8) is that it produces an exact answer for $\sigma = 0$ and $\sigma = 1$. The values of $\sigma > 1$ lead to unstable inversion. For negative σ, the filter is reflected:

$\mathbf{P}_1 =$

	1
σ	$-\sigma - 1$

(10)

The top panel in Figure 2 shows a plane wave, created by applying the helix inverse of filter (8) on a single spike (unit impulse) for the value of $\sigma = 0.7$. We see a noticeable frequency dispersion, caused by the low order of the approximation.

[2] In computational physics, the dimensionless number σ is sometimes referred to as the CFL (Courant, Friedrichs, and Lewy) number (Sod, 1985).

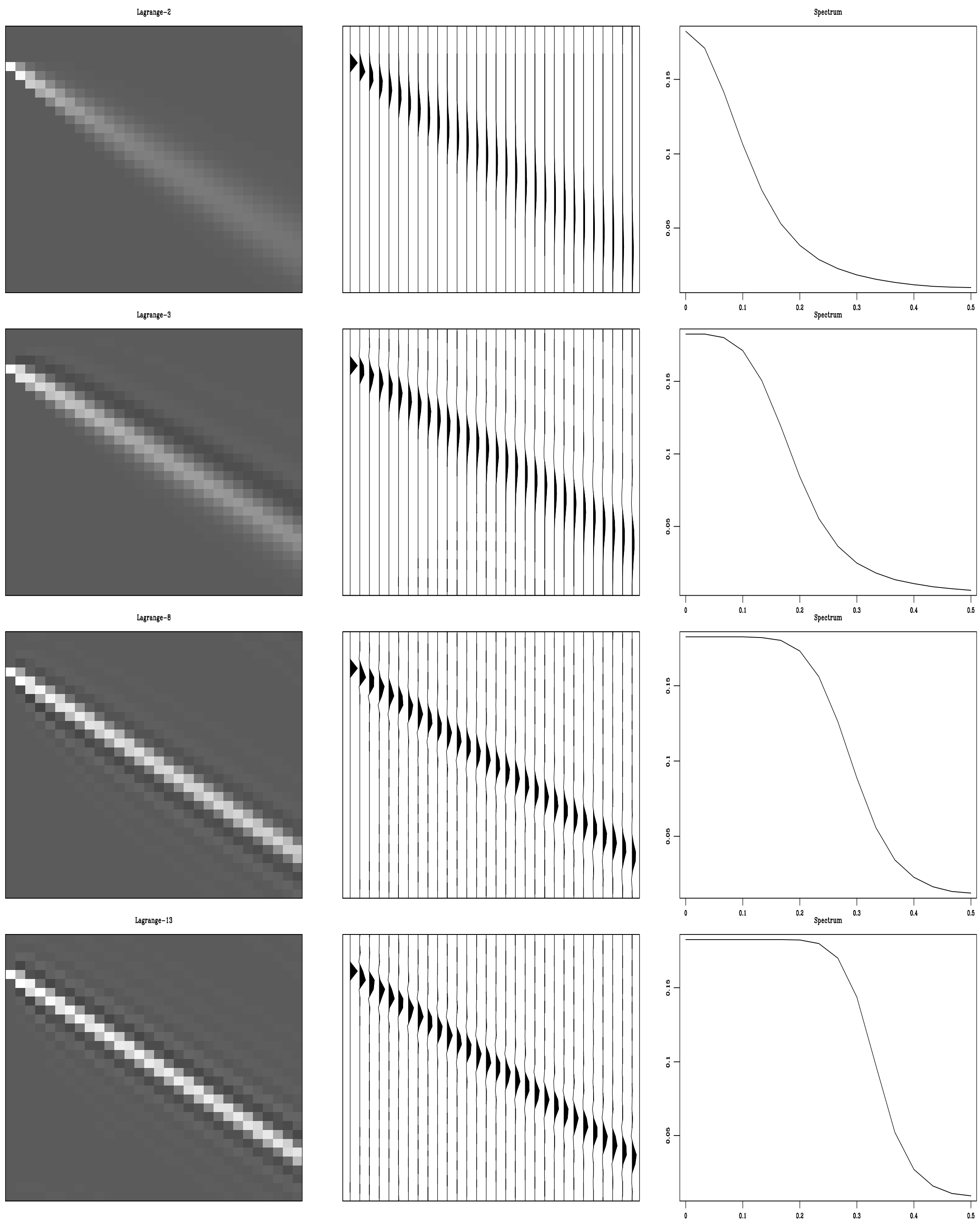

Figure 2: Steering filters with Lagrange interpolation. The left and middle plots show the impulse responses of steering filters: the top panel corresponds to linear interpolation (two-point Lagrange, upwind finite-difference); the second top plot, the three-point Lagrange filter (Lax-Wendroff scheme); the two bottom plots, the 8-point and 13-point Lagrange filters. The right plots in each panel show the corresponding average spectrum. The spectrum flattens and the prediction get more accurate with an increase of the filter size.

The second-order Taylor approximation yields

$$\begin{aligned} S_2(Z) &= Z^{N_t-1}\left(1+\sigma(Z-1)\frac{(\sigma-1)\,\sigma\,(Z-1)^2}{2}\right) = \\ & \frac{\sigma\,(\sigma-1)}{2}\,Z^{T-1}+\left(1-\sigma^2\right)\,Z^T+\frac{\sigma\,(\sigma+1)}{2}\,Z^{T+1}\ , \end{aligned} \tag{11}$$

which corresponds to the 2-D filter

$$\mathbf{P}_2 = \begin{array}{|c|c|c|} \hline & 1 & \\ \hline \frac{\sigma\,(1-\sigma)}{2} & (\sigma^2-1) & -\frac{\sigma\,(\sigma+1)}{2} \\ \hline \end{array} \tag{12}$$

and is equivalent to the Lax-Wendroff finite-difference scheme of equation (9). The interpolation, implied by filter (10) is a local three-point polynomial (Lagrange) interpolation. The correspondence of the Taylor series method, described above, and the Lagrange interpolation can be proved by induction. In general, the filter coefficients for the second row of the N-th order Lagrangian filter are given by the explicit formula

$$a_k = \prod_{i \neq k} \frac{(\sigma - \left[\frac{N}{2}\right] - i)}{(k-i)}\ , \tag{13}$$

where the k and i range from 0 to N. Such a filter has a stable inverse for $-\frac{N}{2} \leq \sigma \leq \frac{N+1}{2}$ and additionally produces an exact answer for all integer σ's in that range. We would have arrived at the same conclusion if instead of expanding the Z-transform of the filter $\mathbf{S}$ around $Z=1$, expanded its Fourier transform around the zero frequency. The latter case corresponds to the "self-similar" construction of Karrenbach (1995). The impulse responses for the helix inverses of different-order Lagrangian filters are shown in Figure 2.

If instead of Taylor series in Z, we use a rational (Padè) approximation, the filter will get more than one coefficient in the first row, which corresponds to an implicit finite-difference scheme. For example, the [1/1] Padè approximation is

$$S_1^1(Z) = \frac{1+\frac{1+\sigma}{2}\,(Z-1)}{1+\frac{1-\sigma}{2}\,(Z-1)} = \frac{1-\sigma+(1+\sigma)Z}{1+\sigma+(1-\sigma)Z} \tag{14}$$

which leads to the filter

$$\mathbf{P}_1^1 = \begin{array}{|c|c|} \hline 1 & \frac{1-\sigma}{1+\sigma} \\ \hline \frac{\sigma-1}{1+\sigma} & -1 \\ \hline \end{array} \tag{15}$$

and corresponds to the Crank-Nicolson implicit scheme.

It is interesting to note that a space-variant convolution with inverse plane filters can create signals with different shape, which remains planar only locally. This situation corresponds to a variable slowness p in the one-way wave equation (9). Figure 3 shows an example: predicting hyperbolas with a 7-point Lagrangian filter.

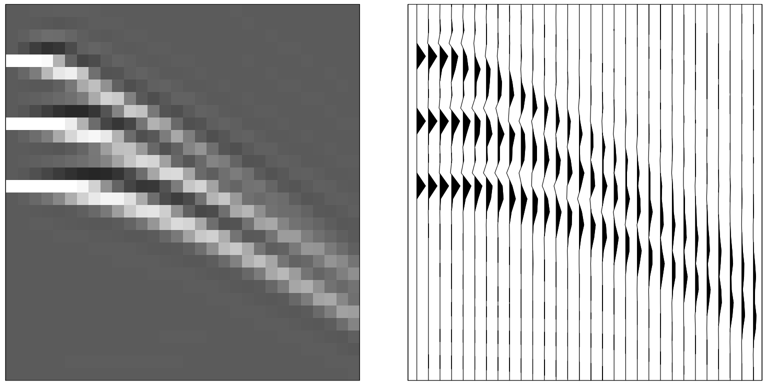

Figure 3: Creating hyperbolas with a variant plane-wave prediction: the impulse response of the inverse 7-point time-and-space-variant Lagrangian filter.

Space variable filters

Steering filters are effective in spreading information along a given direction, but are limited to a single dip. If it is inappropriate to apply a single smoothing direction to the entire model there are two general courses of action:

Patching (Claerbout, 1992b; Schwab and Claerbout, 1995) Redefine our problem into a series of problems, each on a small subset of the data where the stationarity assumption is valid, then recombine the data. This approach leads to problems in determining subsets where the stationarity condition is satisfied and how to effectively remove patching boundaries from the final output.

Space varying filters Filters that vary with location but are spatially smooth. In many ways this is the a more appealing approach. In the past, space varying filters have not been used because they impose significant memory issues (a filter at every location) and must be spatially smooth. By choosing steering filters for our regularization operator and using helix enabled polynomial division, these weaknesses are significantly diminished. We can construct and store relatively small filters which are much easier to smooth (smoothing the preferential dip direction is sufficient). In addition the polynomial division produced inverse filters will have an even higher level of smoothness because each filter spreads information over large, overlapping regions at each iteration.

WELL LOG/DIP INTERPOLATION

To illustrate the effectiveness of this method imagine a simple interpolation problem. Following the methodology of (Fomel et al., 1997) we first bin the data, producing a model $\mathbf{m}$, composed of known data $\mathbf{m}_k$ and unknown data $\mathbf{m}_u$. We have an operator $\mathbf{J}$ which is simply a diagonal masking operator with zeros at known data locations and ones at unknown locations. We can write $\mathbf{m}_k$ and $\mathbf{m}_u$ in terms of $\mathbf{m}$ and $\mathbf{J}$,

$$\mathbf{m}_k \approx (\mathbf{I}-\mathbf{J})\mathbf{m} \tag{16}$$

$$\mathbf{m}_u \approx \mathbf{J}\mathbf{m} \tag{17}$$

where $\mathbf{I}$ is the identity matrix. We have the preconditioning operator $\mathbf{B}$, which applies polynomial division using the helix methodology. In this case we have a single equation in our estimation problem,

$$\mathbf{m}_k \approx (\mathbf{I}-\mathbf{J})\mathbf{B}\mathbf{x}. \tag{18}$$

So the only question that remains is what to use for $\mathbf{B}$, or more specifically $\mathbf{B}^{-1}$, $\mathbf{A}$.

For this experiment we create a series of well logs by subsampling a 2-D velocity field. We use as our a priori information source, reflector dips, to build our steering filters, and thus our operator $\mathbf{A}$. For this test we pick our dips from our "goal", left portion of Figure 4. We define areas in which we believe each of these dips to be approximately correct, and smooth the overall field (right portion of Figure 4).

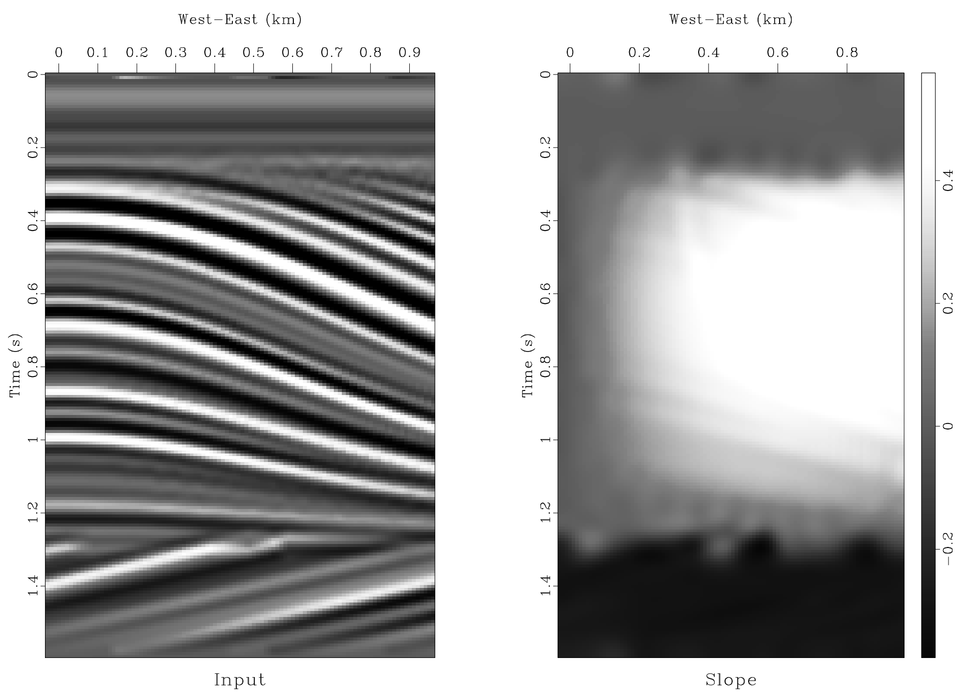

Figure 4: Left, a synthetic seismic section with four picked reflectors indicated by '*'; right; the dip field constructed from the picked reflectors.

For the first test, we simulate nine well logs along the survey (Figure 5). We use equation (18) as our fitting goal and a conjugate gradient solver to estimate $\mathbf{x}$. Within 12 iterations we have a satisfactory solution(Figure 5). If you look closely, especially near the bottom of the section you can still see the well locations, but in general the solution converges quickly to something fairly close to the correct velocity field (Figure 4).

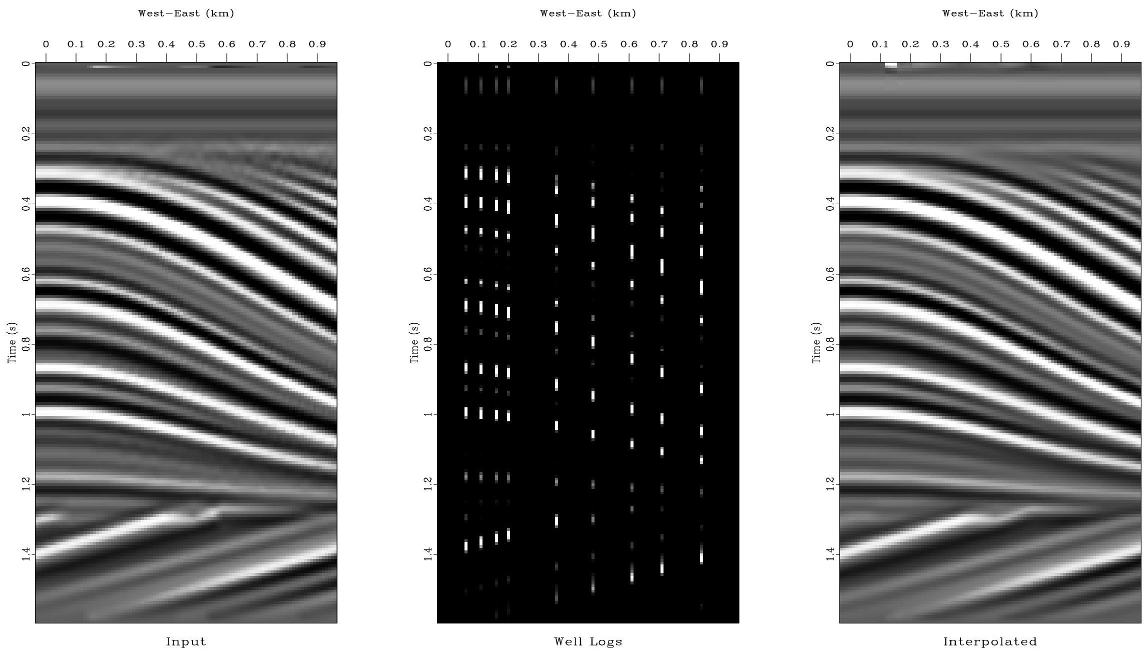

Figure 5: Left, correct velocity field; middle, well subset selected as input; right, velocity field resulting from interpolation.

For a more difficult test, we decreased the number of wells, and give them varying lengths. In Figure 6 you see that in a few iterations we achieve a result quite similar to our goal. In addition, in areas far away from known data the method still followed the general dip direction simply at a lower frequency level.

SHOT-GATHER BASED INTERPOLATION

Another possible application for using recursive steering filters is to interpolate seismic data. As an initial test we chose to interpolate a shot gather. We used a $v(z)$ velocity function to construct hyperbolic trajectories, which in turn were used to construct our dip field (similar to the seismic dips used in the previous section).

For a first test we created a synthetic shot gather using a $v(z) = a + bz$ model as input to a finite difference code. We then cut a hole in this shot gather and attempted to recover the removed values. As Figure 7 shows we did a good job recovering the amplitude within a few iterations.

Left, synthetic shot gather; center, holes cut out of shot gather; right, inversion result after 15 iterations.

To see how the method reacted when it was given data that did not fit its model (in this case hyperbolic moveout) we used a dataset with significant noise problems (ground roll, bad traces, etc.). Using the same technique as in Figure 7 we ended up

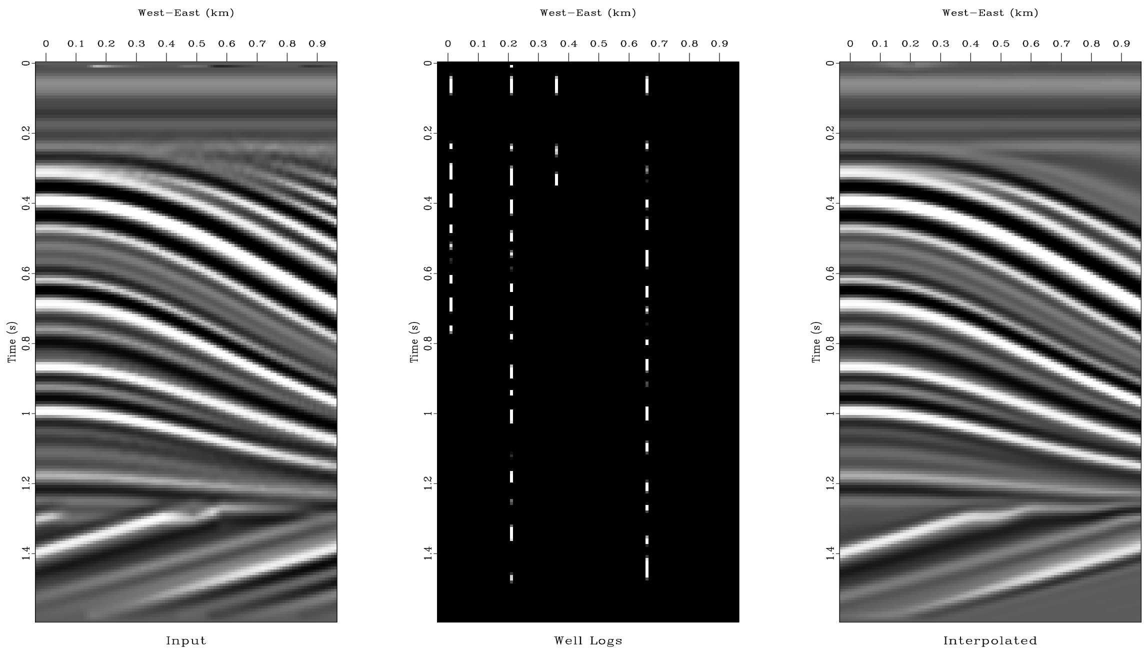

Figure 6: Left model (our goal), middle well logs, and right estimated model after 12 iterations.

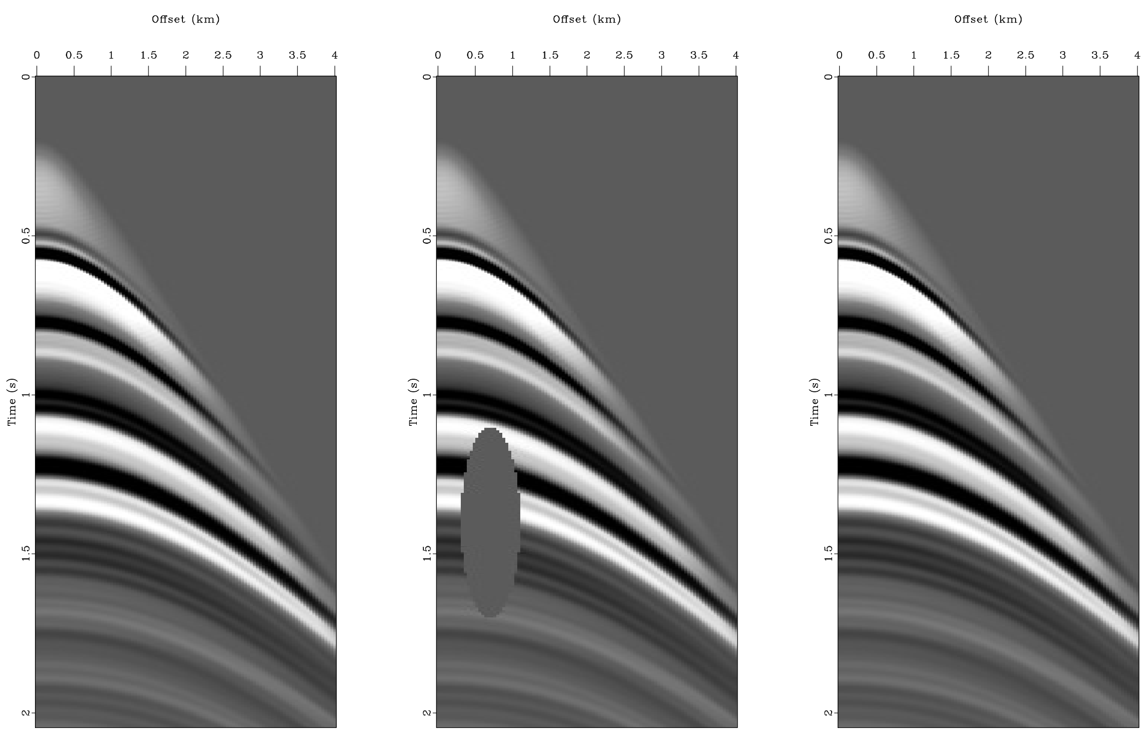

Figure 7:

with a result which did a fairly decent job fitting portions of the data where noise content was low, but a poor job elsewhere (Figure 8). Even where the method did the best job of reconstructing the data, it still left a visible footprint. A more esthetically pleasing result can be achieved by using the above method followed a more traditional interpolation problem using the operator **A** and the fitting goal

$$\mathbf{Am} \approx 0, \tag{19}$$

where **m** is initialized with the result of our previous inversion problem and not allowed to change at locations where we have data. The bottom right panel in Figure 8 shows the result of applying a few iterations of fitting goal (19) to the bottom left result in Figure 8. By using both methodologies the interpolated data does a much better job blending into its surroundings but still is a poor interpolation result.

FUTURE WORK AND CONCLUSIONS

We show that by using helicon enabled inverse operators built from small steering filters we can quickly obtain esthetically pleasing models. Tests on smooth models, with a single dip at each location proved successful. The methodology does not adequately handle models with multiple dips at each location and presupposes some knowledge of the desired final model. A different approach would be to estimate the steering filters (**S**) from the experimental data (**m**). Generally, this leads to a system of non-linear equations

$$\mathbf{P}(\sigma)\mathbf{m} = (\mathbf{I} - \mathbf{S}(\sigma))\mathbf{m} = \mathbf{0} \;, \tag{20}$$

which need to be solved with respect to σ. One way of solving system (20) is to apply the general Newton's method, which leads to the iteration

$$\sigma_k = \sigma_{k-1} + \frac{\mathbf{P}(\sigma_{k-1})\mathbf{m}}{\mathbf{S}'(\sigma_{k-1})\mathbf{m}} \;, \tag{21}$$

where the derivative $\mathbf{S}'(\sigma)$ can be computed analytically. It is interesting to note that if we start with $\sigma = 0$ and apply the first-order filter (8), then the first iteration of scheme (21) will be exactly equivalent to the slope-estimation method of Claerbout (1992a), used by Bednar (1997) for calculating coherency attributes. Finally, the steering filter regularization methodology needs to be tried in conjunction with a variety of operators and applied to real data problems.

REFERENCES

Bednar, J. B., 1997, Least squares dip and coherency attributes, *in* SEP-95: Stanford Exploration Project, 219–225.

Claerbout, J., 1997, Multidimensional recursive filters via a helix, *in* SEP-95: Stanford Exploration Project, 1–13.

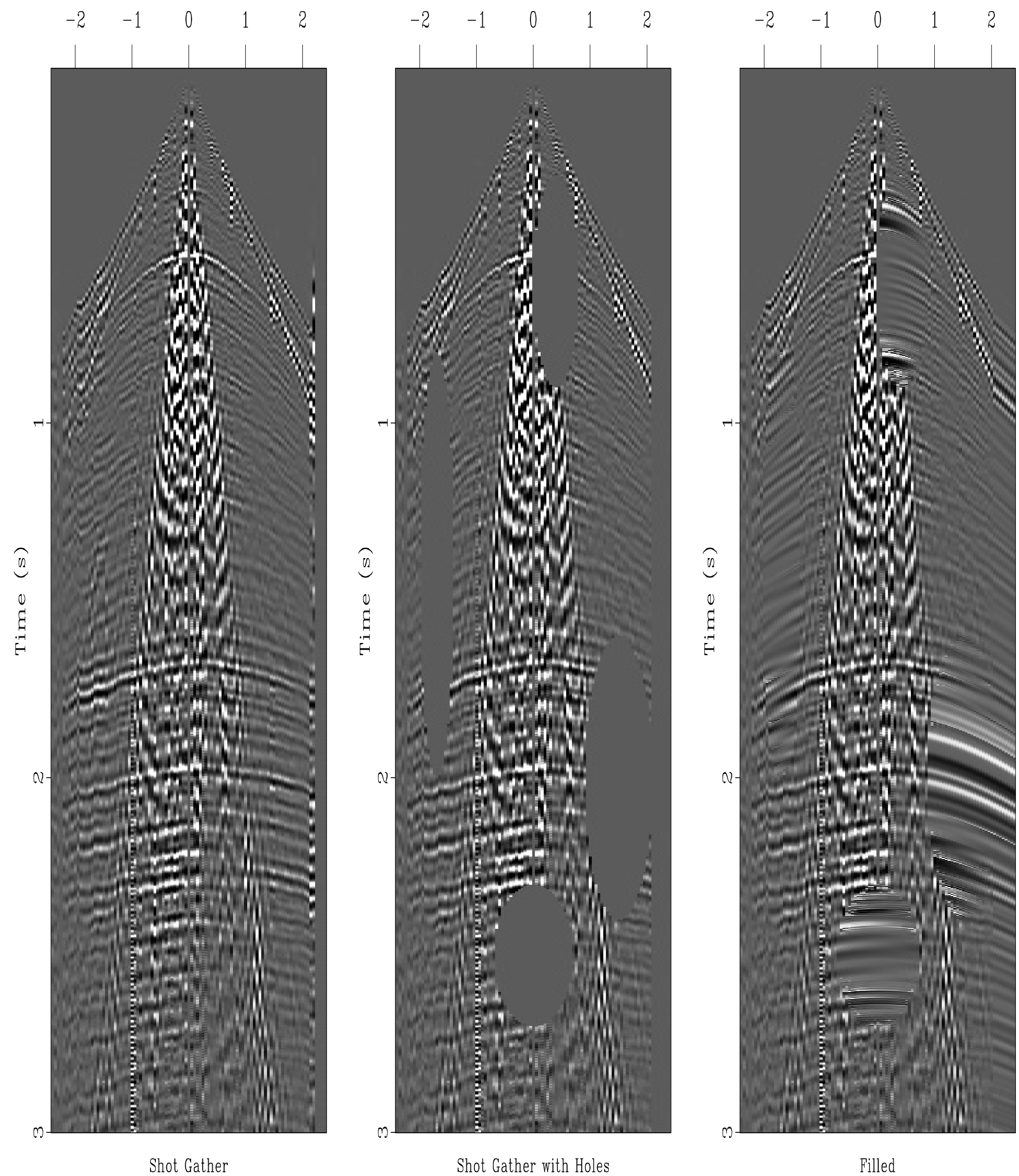

Figure 8: Top left, original shot gather; top right, gather with holes (input); bottom left, result applying equation 18, bottom right, result after applying equation (18) followed by (19).

Claerbout, J., and D. Nichols, 1994, Spectral preconditioning, *in* SEP-82: Stanford Exploration Project, 183–186.

Claerbout, J. F., 1992a, Earth Soundings Analysis: Processing Versus Inversion: Blackwell Scientific Publications.

———, 1992b, Nonstationarity and conjugacy: Utilities for data patch work, *in* SEP-73: Stanford Exploration Project, 391–400.

———, 1994, Applications of Three-Dimensional Filtering: Stanford Exploration Project.

Clapp, R. G., and B. Biondi, 1995, Multi-azimuth velocity estimation, *in* SEP-84: Stanford Exploration Project, 75–88.

Etgen, J., 1997, Problems and prospects in interval velocity estimation. Presented on Stanford Exploration Project Annual Meeting.

Fomel, S., 1997, On model-space and data-space regularization: A tutorial, *in* SEP-94: Stanford Exploration Project, 141–164.

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

Karrenbach, M., 1995, Elastic tensor wave fields: PhD thesis, Stanford University.

Nichols, D., 1994, Velocity-stack inversion using $\mathbf{L_p}$ norms, *in* SEP-82: Stanford Exploration Project, 1–16.

Schwab, M., 1997, Cross product operator detects plane reflectors, *in* SEP-94: Stanford Exploration Project, 181–188.

Schwab, M., and J. Claerbout, 1995, The interpolation of a $3-D$ data set by a pair of $2-D$ filters, *in* SEP-84: Stanford Exploration Project, 271–278.

Sod, G. A., 1985, Numerical methods in fluid mechanics: Cambridge University Press.