Stanford Exploration Project, Report 82, May 11, 2001, pages 1–17

Velocity-stack inversion using $\mathbf{L_p}$ norms

Dave Nichols[1]

ABSTRACT

Velocity-stack inversion is the process of creating a model in velocity space that can correctly reconstruct the measured data. This is usually implemented by minimizing the L_2 norm of the data misfit and the L_2 norm of the model. Superior velocity space images, with a better separation of primaries and multiples, can be created by minimizing a different norm of the data misfit and a different norm of the model length. Norms closer to the L_1 norm appear to be better at rejecting impulsive noise and creating a spiky model.

INTRODUCTION

The velocity stack domain is a alternative domain for processing seismic data. The normal CMP-gather domain has axes of time, offset, and midpoint. The velocity-stack domain has axes of zero-offset time, velocity or slowness, and midpoint. Many data processing applications are much simpler in the velocity stack domain. E.g. multiple attenuation can be performed as a simple muting operation in the velocity-stack domain. This topic is addressed in another paper in this report (**?**). The other common use for a domain with coordinates of zero-offset time and velocity is the estimation of RMS velocities. In velocity analysis the time/offset data is transformed by stacking along many different hyperbolas and then plotting either stack power, correlation or semblance measures. The major distinguishing feature of this process is that the data is not intended to be transformed back to the time/offset domain. In contrast, the velocity-stack transform is intended to be an invertible (or approximately invertible) transform. After processing in the velocity-stack domain the data is transformed back to the original time/offset domain and further standard processing can be applied. Some transforms have analytic inverses or inverses that are simple to calculate. Time invariant transforms may have a Toeplitz structure in the frequency domain which can lead to an efficient inverse (Kostov, 1991). Unfortunately this is not true of the velocity stack transform. However, even when no analytic inverse is known it is possible to create a transform/inverse-transform pair. Once the forward transform is defined and the inverse transform can be implemented using an iterative solver. The usual process is to implement the inverse as the minimization of a least-squares problem. The least-squares solution has some attributes that may be undesirable. If the model space is overdetermined the least-squares solution will be usually be spread out over all the possible solutions. Other methods may be more useful if we desire a parsimonious representation. The inverse velocity-stack and related operators have been used for

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

many years at SEP, Thorson (**?**) used a stochastic inverse that penalized low variance models to encourage a spiky solution. Harlan (**?**) used a carefully sampled model space that avoided problems with underdetermined solutions. In this paper, I demonstrate some limitations of the least-squares criterion and I show that iterative solution using other norms can lead to more pleasing solutions.

LEAST SQUARES INVERSION

In velocity stack inversion we assume that we can implement the forward operator **H** that maps from velocity-stack space to offset space. If there is spike in velocity-stack space it will create a hyperbola in the CMP gather. The amplitude of the hyperbola is given by the height of the spike, the zero-offset time and the moveout of the hyperbola are governed by the position of the spike in the velocity-stack space. Figure 1 shows a model, **m**, with four spikes in the velocity-stack space and the corresponding data, **d**, in offset space that is created by the action of the forward operator **H**.

$$\mathbf{d} = \mathbf{H}\mathbf{m}$$

This data is used as the synthetic input for the tests of various inversion norms. For the inverse

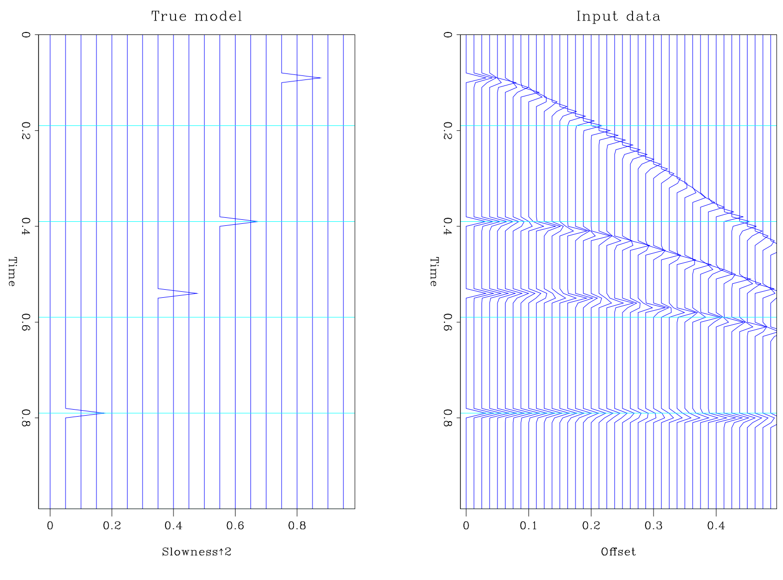

Figure 1: Model in velocity-stack space and data in offset space created by application of the forward operator. $\boxed{\text{dave2-synth1}}$ [CR]

operator we seek a solution to the problem of finding the model in velocity-stack space, **m**, given the data, **d**. This is usually posed as a least-squares minimization problem that minimizes the energy of the misfit between the modeled and measured data:

$$\min_{\mathbf{m}} |\mathbf{d} - \mathbf{H}\mathbf{m}|_2.$$

An iterative least-squares solution will start with an initial estimate of the solution $\mathbf{m_0}$. If we use a conjugate gradient type algorithm it will not only minimize the data misfit, it will also minimize the energy of the change in the model space.

$$|\mathbf{m} - \mathbf{m_0}|_2$$

This is because the adjoint operator puts no energy into the null space of the operator. If the initial solution is zero, the final solution will be the minimum energy solution, the solution with no energy in the null-space.

$$|\mathbf{m}|_2$$

Figure 2 shows the result of using a conjugate gradient solver to minimize the L_2 norm of the data misfit. The left frame displays the input data, the center frame shows the inverted data in the velocity-stack space and the right frame displays the predicted data from the model. The model predicts the input data very well. However the inverted model is not the same as the original model data in Figure 1. The difference between the two models is in the null space of the operator $\mathbf{H}$. This is easily demonstrated by taking the difference of the two models and applying the modeling operator. The result is shown in Figure 3. The original model is not the one chosen by the inversion because it contains components in the null space. The L_2

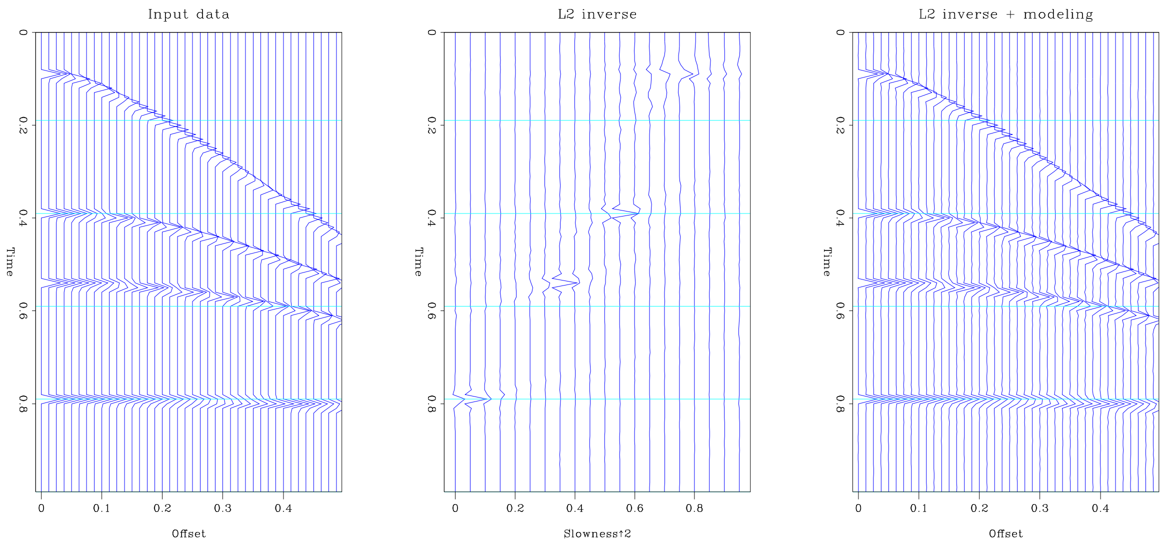

Figure 2: Input noise free data, velocity-stack inversion using L_2 norm, remodeled data dave2-l2synth [CR]

inversion is the optimal choice in the presence of Gaussian noise. In this case, the L_2 inversion gives the maximum likelihood solution to the inversion problem. Figure 4 shows the results of L_2 inversion when the data is contaminated by Gaussian noise with zero mean and variance of 1/10 of the true model amplitude. The inversion does a satisfactory job of recovering the true model. If the noise is not Gaussian then the L_2 inversion may not be a good choice. Figure 5 illustrates the behavior of L_2 inversion in the presence of impulsive noise. Four spikes with an amplitude ten times that of the hyperbolas have been added to the input data. The inversion spreads the effect of the spike across the model space. This is especially true of the spikes at the far offset. When the data is remodeled the spikes are not perfectly recreated, they have been smeared into the adjacent traces.

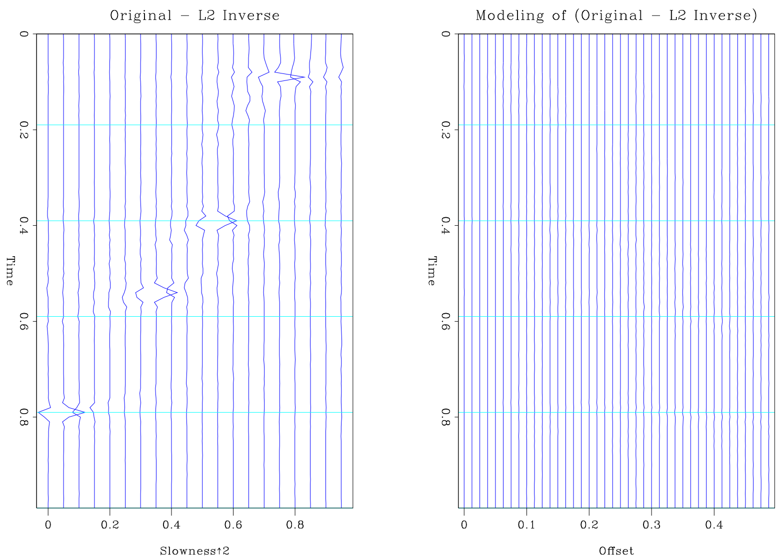

Figure 3: Difference between the original model and the result of L_2 inversion. The result is in the null space of the operator. When the operator is applied the result is zero. dave2-synth-null [CR]

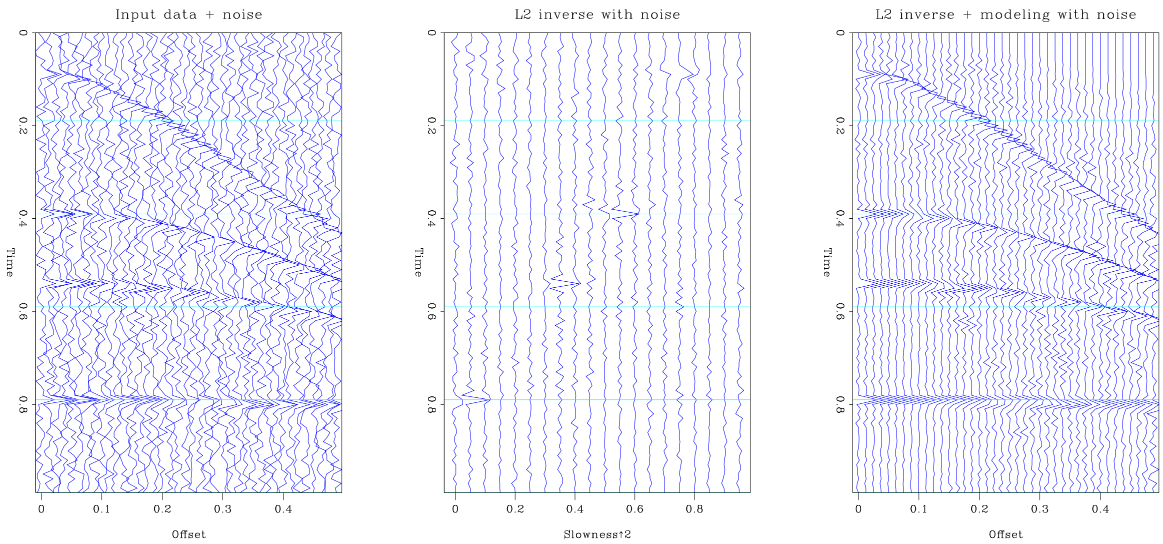

Figure 4: Input data with random noise, velocity-stack inversion using L_2 norm, remodeled data dave2-l2synthn [CR]

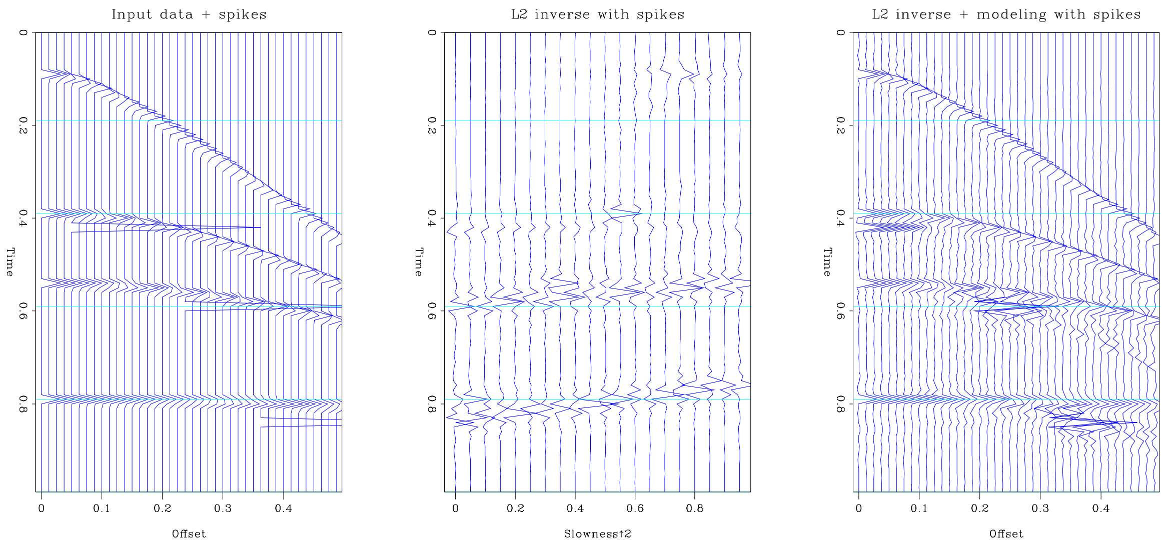

Figure 5: Input data with impulsive noise, velocity-stack inversion using L_2 norm, remodeled data dave2-l2synths [CR]

WEIGHTING THE RESIDUAL

The simple L_2 norm of the residual is not the only function that we can choose to minimize. We can also choose to minimize a weighted residual

$$\min_{\mathbf{m}} |\mathbf{W_L d} - \mathbf{W_L H m}|_2.$$

Where $\mathbf{W_L}$ is a diagonal operator in the data space. This is equivalent to solving the weighted problem

$$\mathbf{W_L H m} = \mathbf{W_L d}.$$

Any weighting operator can be chosen as long as it does not have any zeros on the diagonal. Tarantola (1987) would choose $\mathbf{W_L}$ to be a "diagonal inverse noise covariance estimate". This noise covariance represents an estimate of the reliability of each data sample. If the variance is high then the data sample is unreliable and the residual at that location should influence the inversion less than that at a location where the variance is low.

The L_p norm

A particular choice for $\mathbf{W_L}$ is one that results in minimizing the L_p norm of the residual. Choose the i^{th}diagonal element of $\mathbf{W_L}$ to be a function of the i^{th} component of the data vector.

$$\mathbf{W_L}_{ii} = |r_i|^{(p-2)/2}$$

This can be thought of as a method that estimates the noise variance from the residual after the inversion. The weighted residual is then

$$\mathbf{r}^\dagger \mathbf{W_L}^\dagger \mathbf{W_L r}$$

$$= \Sigma_i r_i^\dagger |r_i|^{(p-2)} r_i$$
$$= \Sigma_i |r_i|^p = |\mathbf{r}|_p^p$$

The function that is minimized is the L_p norm of the data misfit. This method is valid for all norms $1 < p \leq 2$. In this article I will often refer to the L_1 norm. However since this method is not valid for the true L_1 norm, I am really referring to the $L_{1+\epsilon}$ norm where ϵ is very small. When the "L_1" norm is desired the weighting is $\mathbf{W}_{\mathbf{L}ii} = |r_i|^{-1/2}$. The gradient direction is given by

$$\mathbf{g} = \mathbf{H}^\dagger \mathbf{W}_\mathbf{L} \mathbf{r}$$

This gradient down-weights the importance of large residuals, and concentrates the model on improving the fit to the data that is already well estimated. Since the final residual is not known during the solution of the problem, the minimization is solved using a sequence of weighted least-squares problems. This method is called "Iteratively reweighted least-squares". The method has been used before at SEP for deconvolution (**?**) but not for velocity-stack inversion. The cost is a little higher than a regular least-squares solver but not excessive. In practice the reweighting operator is modified slightly to avoid dividing by zero. A damping parameter ϵ is chosen and the weighting operator is modified to be:

$$\mathbf{W}_{\mathbf{L}ii} = |r_i|^{(p-2)/2} \qquad ; \ |r_i| > \epsilon$$

$$\mathbf{W}_{\mathbf{L}ii} = |\epsilon|^{(p-2)/2} \qquad ; \ |r_i| < \epsilon$$

The correct choice of this damping operator is related to the condition number of the operator, but a practical choice appears to be

$$\epsilon = \max |\mathbf{d}|/100.$$

The algorithm then works as follows:

1. Calculate current residual.
2. Construct weighting operator.
3. Solve least-squares problem with new weighting operator.
4. Go to step 1.

It can be shown (Gersztenkorn et al., 1986) that this method converges to the correct L_p norm solution. In fact, we do not even need to completely solve the reweighted least-squares problem at each step. I use a few iterations of a conjugate gradient scheme before calculating a new weighting function and restarting the iterations. Figure 6 shows the result of minimizing a $L_{1.1}$ data misfit when the data is contaminated by Gaussian noise, and Figure 7 shows the result when the noise is impulsive. The reconstruction of the data with Gaussian noise is slightly inferior to the L_2 norm. The noise has been concentrated into a few significant events rather than remaining random. The second example is the type of problem that L_1 methods are best at handling. The algorithm has completely ignored the spikes in the input data and the reconstructed data is completely noise free. However, even though we are estimating a model that minimizes the L_1 norm of the data misfit, we will still be finding the minimum energy model that satisfies this misfit criterion. The operator has the same null space as the original operator $\mathbf{H}$ so no energy is added to the null space of the operator. The inverted data still does not have the spikiness present in the original model.

$$\min_{\mathbf{m}} |\mathbf{d} - \mathbf{H}\mathbf{m}|_1$$

and

$$|\mathbf{m}|_2$$

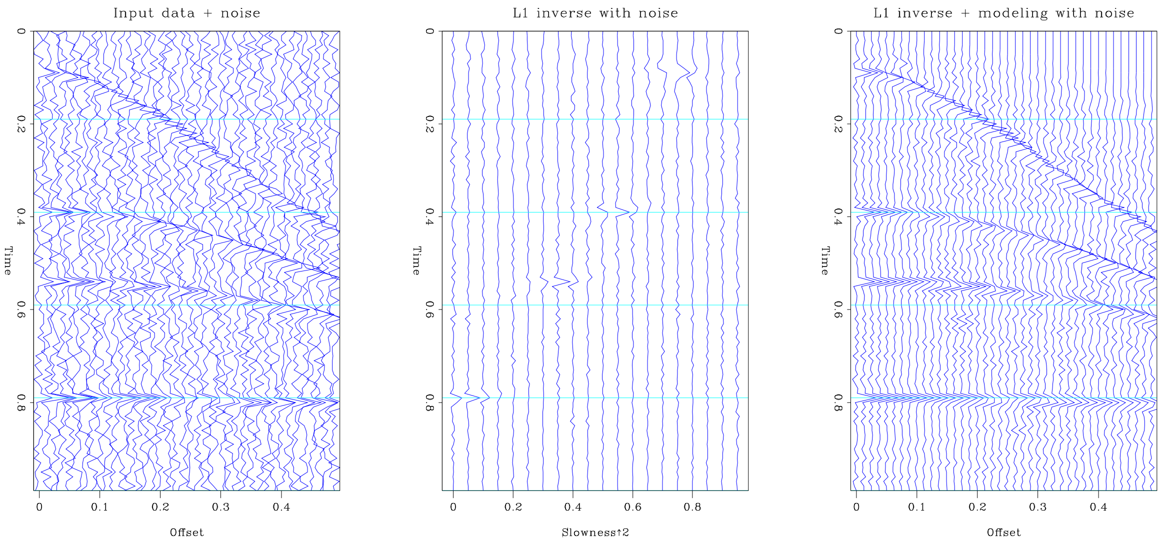

Figure 6: Input data with random noise, velocity-stack inversion using $L_{1.1}$ norm, remodeled data dave2-l1synthn [CR]

WEIGHTING THE MODEL

One way to address the loss of spikiness in the model is to apply a model space weight. Now we solve the problem

$$\mathbf{H}\mathbf{W_R}\hat{\mathbf{m}} = \mathbf{d},$$

followed by

$$\mathbf{m} = \mathbf{W_R}\hat{\mathbf{m}}.$$

The iterative solution of this system minimizes the energy of the new model parameters $\hat{\mathbf{m}}$.

$$\begin{aligned} |\hat{\mathbf{m}}|^2 &= \hat{\mathbf{m}}^\dagger\hat{\mathbf{m}} \\ &= \mathbf{m}^\dagger \mathbf{W_R^{-1}}^\dagger \mathbf{W_R^{-1}} \mathbf{m} \end{aligned}$$

The iterative solution will have no energy in the null space of the composite operator $\mathbf{H}\mathbf{W_R}$.

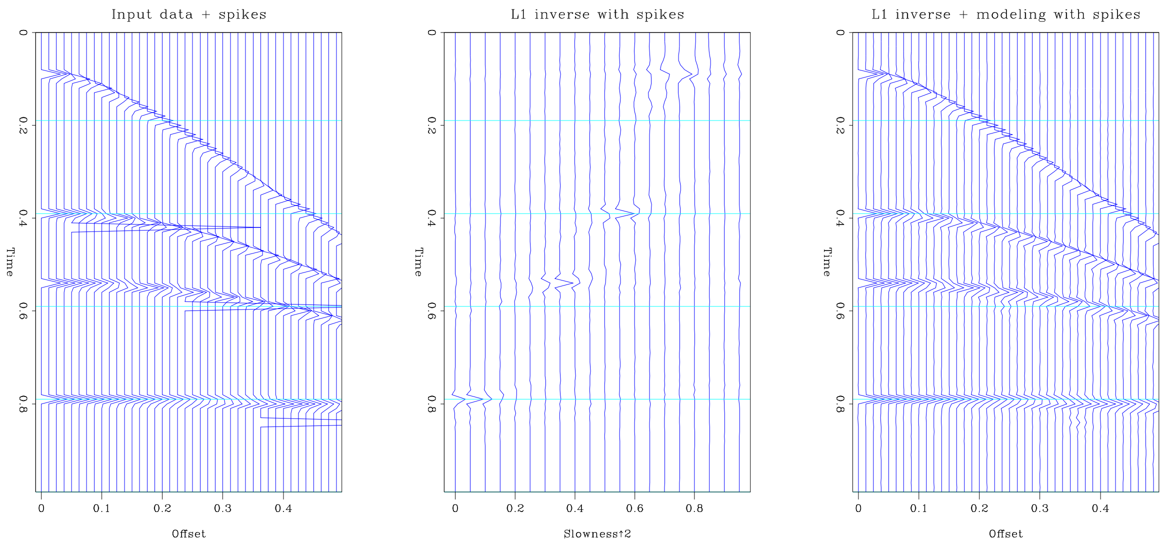

Figure 7: Input data with impulsive noise, velocity-stack inversion using $L_{1.1}$ norm, remodeled data $\boxed{\text{dave2-l1synths}}$ [CR]

Minimizing the L_p norm of the model

In the same vein as the previous section we can choose the diagonal elements of $\mathbf{W_R}$ to be

$$\mathbf{W_R}_{ii} = |\mathbf{m}_i|^{(2-p)/2}$$

The weighted model energy function that is minimized is now

$$\Sigma \mathbf{m}_i^\dagger |\mathbf{m}_i|^{(p-2)} \mathbf{m}_i = |\mathbf{m}|_p$$

Which is the L_p norm of the model. For the L_1 norm the weight is $|\mathbf{m}_i|^{1/2}$. The gradient direction is then

$$\mathbf{g} = \mathbf{W_R H^\dagger r}$$

This gives more energy to the model points that already have a large energy and thus tends to make a "spikier" model space. A practical implementation of the model weighting involves a damping factor to prevent multiplication by zero.

$$\mathbf{W_R}_{ii} = |\mathbf{m}_i|^{(2-p)/2} \qquad ; \quad |\mathbf{m}_i| > \nu$$

$$\mathbf{W_R}_{ii} = |\nu|^{(2-p)/2} \qquad ; \quad |\mathbf{m}_i| < \nu$$

One choice for the damping factor that has proven successful is

$$\nu = \max_i |\mathbf{m}_i|/100.$$

By combining model space weights and data space weights we can minimize the L_p norm of both the data misfit and the model space. We solve the following problem:

$$\mathbf{W_L H W_R \hat{m}} = \mathbf{W_L d},$$

followed by

$$\mathbf{m} = \mathbf{W_R}\hat{\mathbf{m}}.$$

This method is implemented using an iteratively reweighted least-squares algorithm that is similar to the previous scheme.

1. Calculate current residual.
2. Construct left weighting operator using current residual.
3. Construct right weighting operator using current model.
4. Solve least-squares problem with new weighting operators.
5. Transform current solution to unweighted model space.
6. Go to step 1.

We now have a scheme that minimizes the L_p norm in both the data misfit and the model space.

$$\min_{\mathbf{m}} |\mathbf{d} - \mathbf{H}\mathbf{m}|_p$$

and

$$|\mathbf{m}|_p$$

Figures 8 and 9 show the result of using the $L_{1.1}$ norm in both domains. This method retains the ability of the $L_{1.1}$ data misfit to ignore impulsive noise but it now also creates a spiky model. The data with Gaussian noise still has undesirable artifacts in the reconstructed data but, as we expected, the model in velocity-stack space is a much better match to the original spiky model. Both the model and the reconstructed data in figure 9 are excellent matches to the original model and data in Figure 1.

REAL DATA EXAMPLE

The following figures show the effects of using the various inversion methods on one CDP gather from the dataset that was provided for the Mobil AVO workshop. The aim of the inversion is to create an image in the velocity-stack space in which the primary energy and the multiple energy is well separated. In addition the modeled data must be a good match to the primary-only part of the original data. Since this dataset is intended to be used for AVO analysis, it is important that the amplitudes of the events are correctly modeled. Figure 10 displays four stages in the processing of a CMP gather from the Mobil dataset. The first frame is the raw CMP gather, the second frame is its L_2 velocity-stack inverse, the third frame is the data remodeled from the inverse, and the final frames is the difference between the input CMP and the remodeled data. The raw data is heavily contaminated by multiples and there are high amplitude bursts at the far offsets. The noise bursts produce long curved artifacts in the velocity-stack space. They are the dominant feature in the inversion and it is difficult

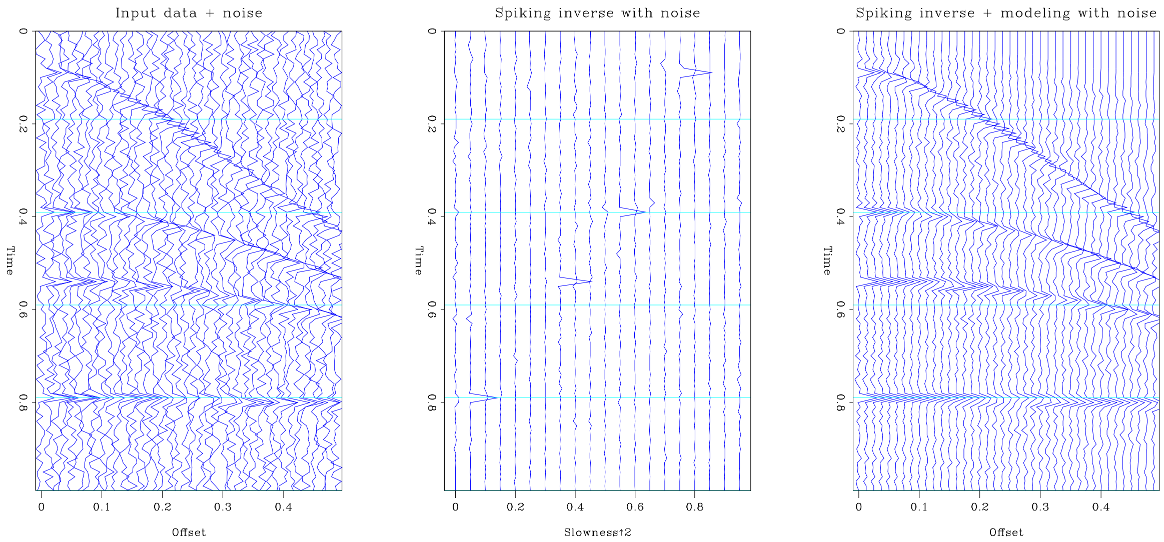

Figure 8: Input data with random noise, velocity-stack inversion using L_1 norm in both spaces, remodeled data dave2-spksynthn [CR]

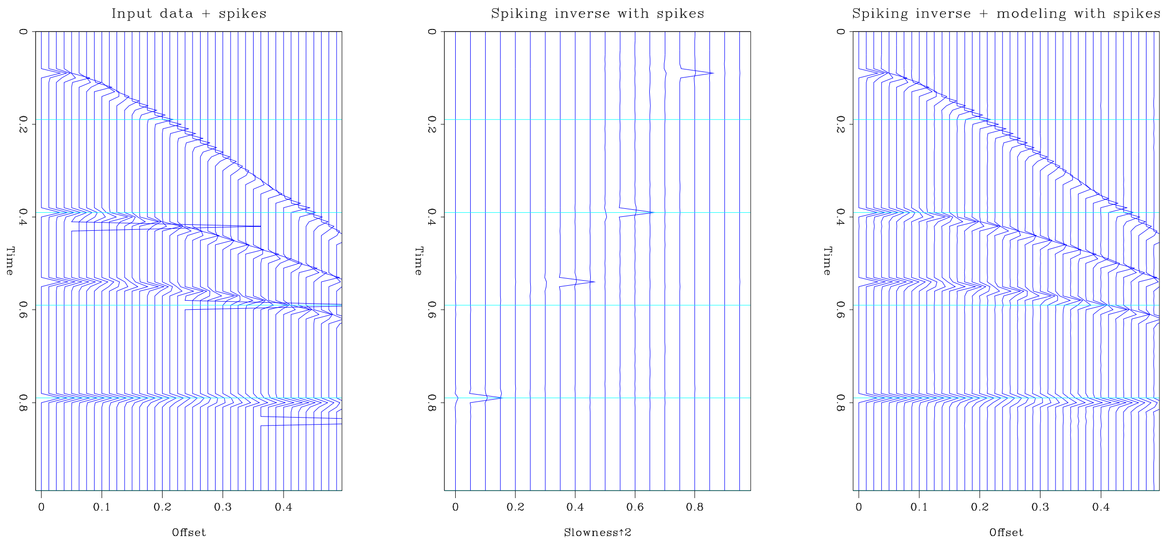

Figure 9: Input data with impulsive noise, velocity-stack inversion using L_1 norm in both spaces, remodeled data dave2-spksynths [CR]

to identify the primary velocity trend. However, despite the unpleasant appearance in the velocity-stack space the reconstructed data is a good match to the input and the residual is small over the whole domain. This is the characteristic behavior of L_2 inversion, it attempts to minimize the global energy in the residual. Figure 11 shows the same four stages of processing when the $L_{1.1}$ norm of the data misfit is minimized. The model in velocity stack space is still contaminated by the noise streaks but not quite as badly as the L_2 inversion. The remodeled data is still a good match to most of the input data but the algorithm has treated the high energy bursts at the far offsets as noise and it has rejected them. The residual energy is higher than in the L_2 case, but it is all concentrated in the far offset noise bursts. Figure 12 displays the effect of spiking inversion in the model space. The inversion minimized the $L_{1.1}$ norm of the data misfit and the $L_{1.1}$ norm of the model. The events in velocity-stack space are much more compact and the noise streaks do not cross the trend of the primary energy. This is a very desirable result, if we intend to mute the data in velocity-stack space. It is much easier to separate the primary trend (to the left of the figure) from the multiples and the effects of the noise bursts. The differing concentration of the energy can be seen more clearly be examining the envelope of the data in velocity-stack space. Figure 13 compares the envelope of the data for the three inversions. The result of spiking inversion shows a much better separation of primary and multiple data.

DISCUSSION

The choice of a suitable norm for data misfit is dependent on your expectations of the the noise character. If the noise is known to be Gaussian and uncorrelated, it is best to minimize the L_2 norm of the data misfit. If the noise is expected to be impulsive, it is better to minimize a lower order norm. I have shown that for noise that is pure spikes the $L_{1.1}$ norm was very successful: it completely eliminated the spikes. The same considerations apply to the norm of the model. If the model is expected to be smooth, then a L_2 norm may be appropriate. If the model that you desire is spiky, a $L_{1.1}$ norm may be more to your taste.

An advertisement for C++

All of the iterative solver routines developed for this work were written within the “CLOP” framework (**?**). This means that the solvers are now available for use with *any* operator written within that framework. The same routine can be used with velocity-stack operators, convolutional operators, migration operators etc. This is in sharp contrast to many problems coded in Fortran for which a new solver often needs to be implemented for each new operator.

REFERENCES

Gersztenkorn, A., Bednar, J. B., and Lines, L. R., 1986, Robust iterative inversion for the one-dimensional acoustic wave equation: Geophysics, **51**, no. 2, 357–368.

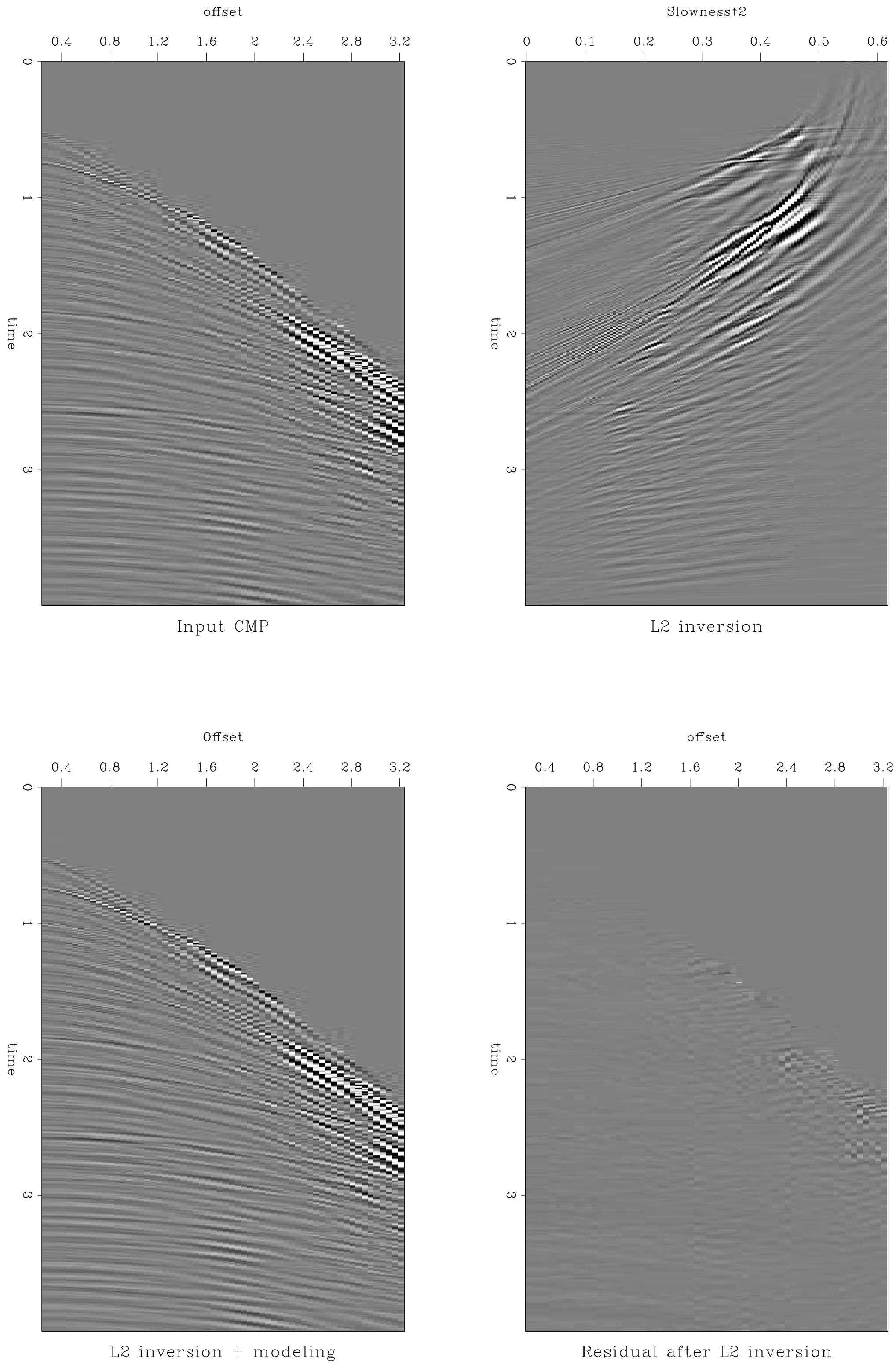

Figure 10: L2 inverse processing. Top left: input data. Top right: velocity-stack inverse using L_2 norm of data misfit. Bottom left: reconstructed input data. Bottom right: residual data not modeled. dave2-real-l2 [CR]

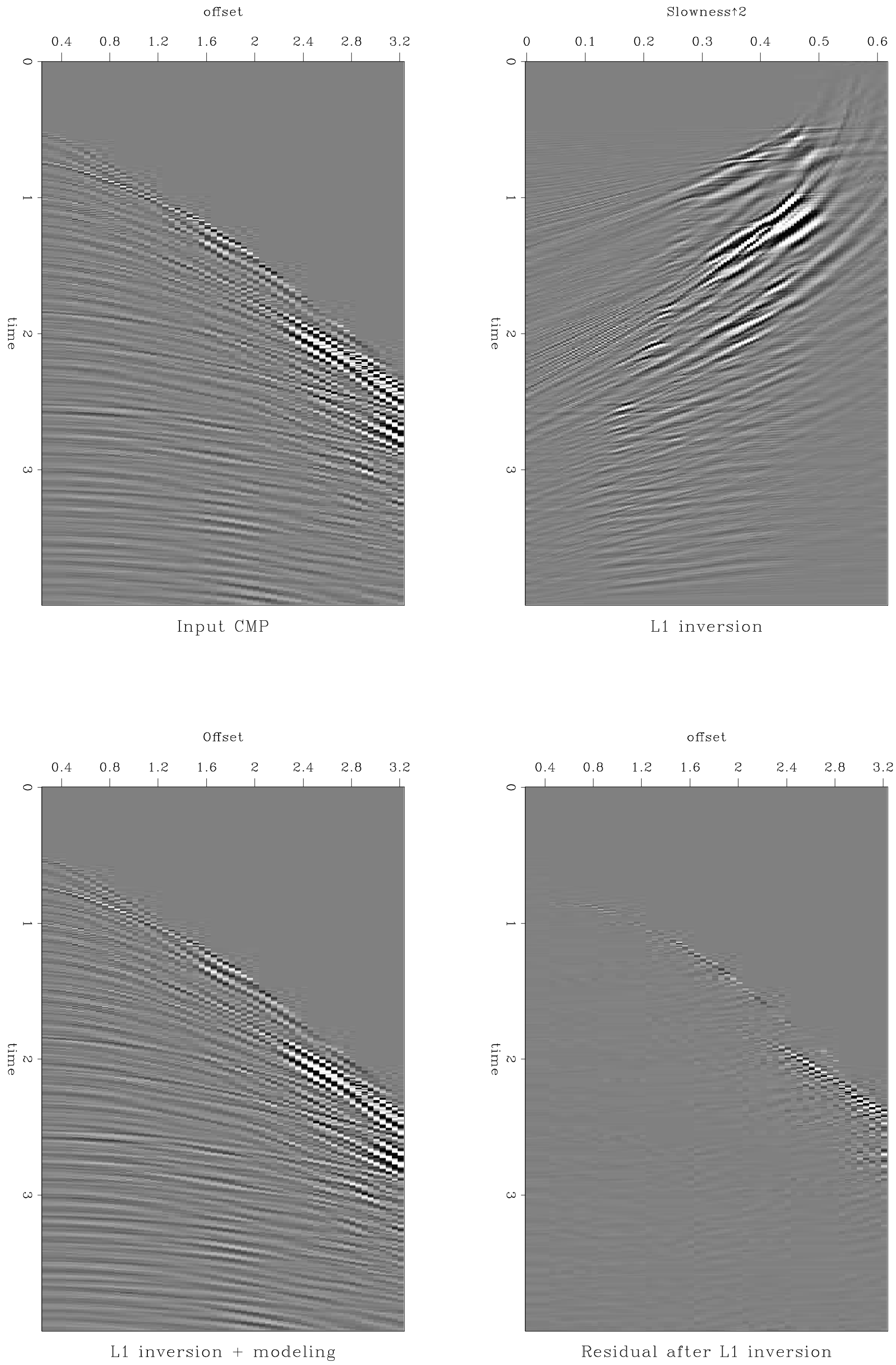

Figure 11: $L_{1.1}$ inverse processing. Top left: input data. Top right: velocity-stack inverse using $L_{1.1}$ norm of data misfit. Bottom left: reconstructed input data. Bottom right: residual data not modeled. $\boxed{\text{dave2-real-l1}}$ [CR]

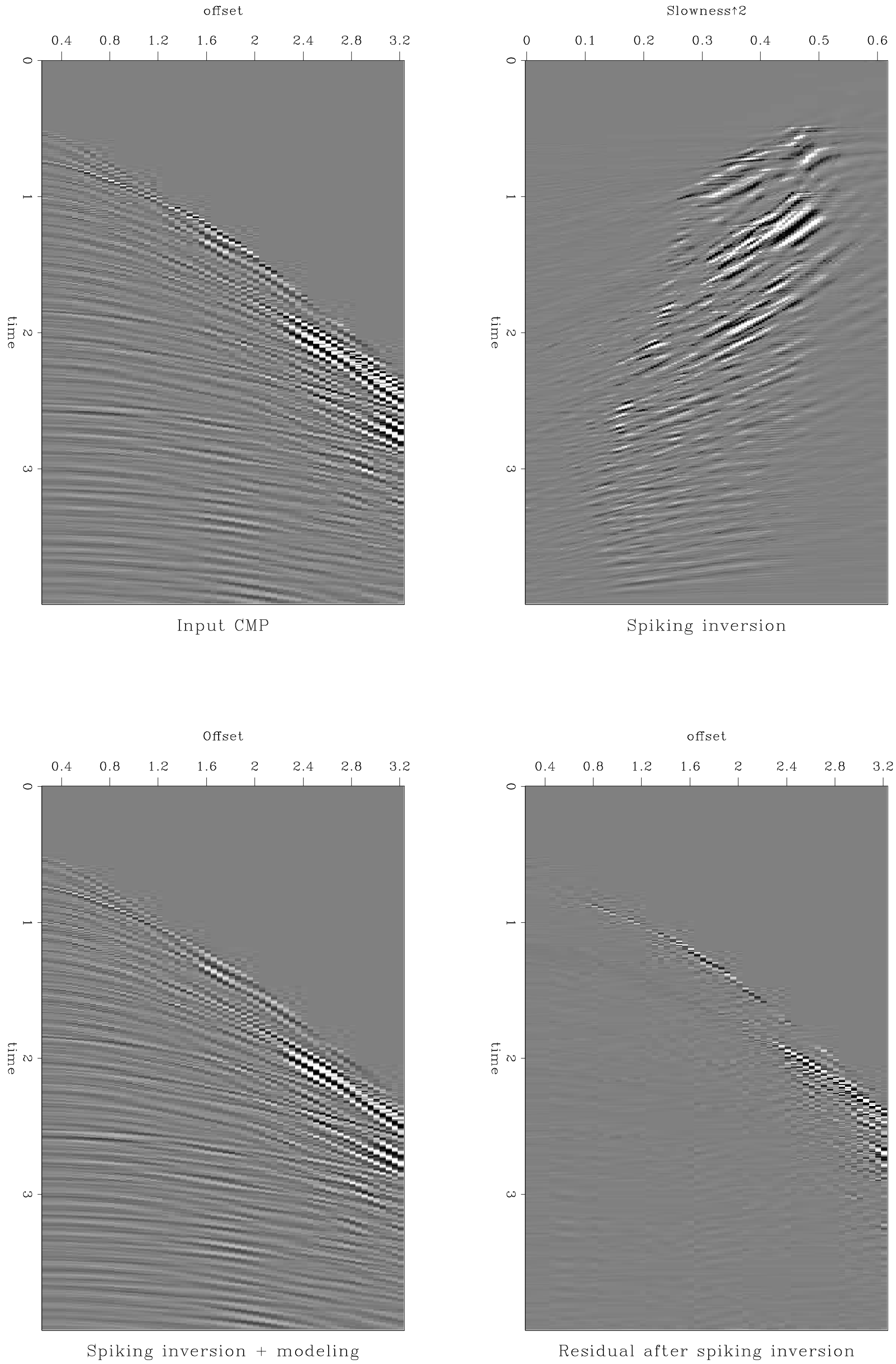

Figure 12: Spiking inverse processing. Top left: input data. Top right: velocity-stack inverse using $L_{1.1}$ norm of data misfit and $L_{1.1}$ norm of model length. Bottom left: reconstructed input data. Bottom right: residual data not modeled. dave2-real-spk [CR]

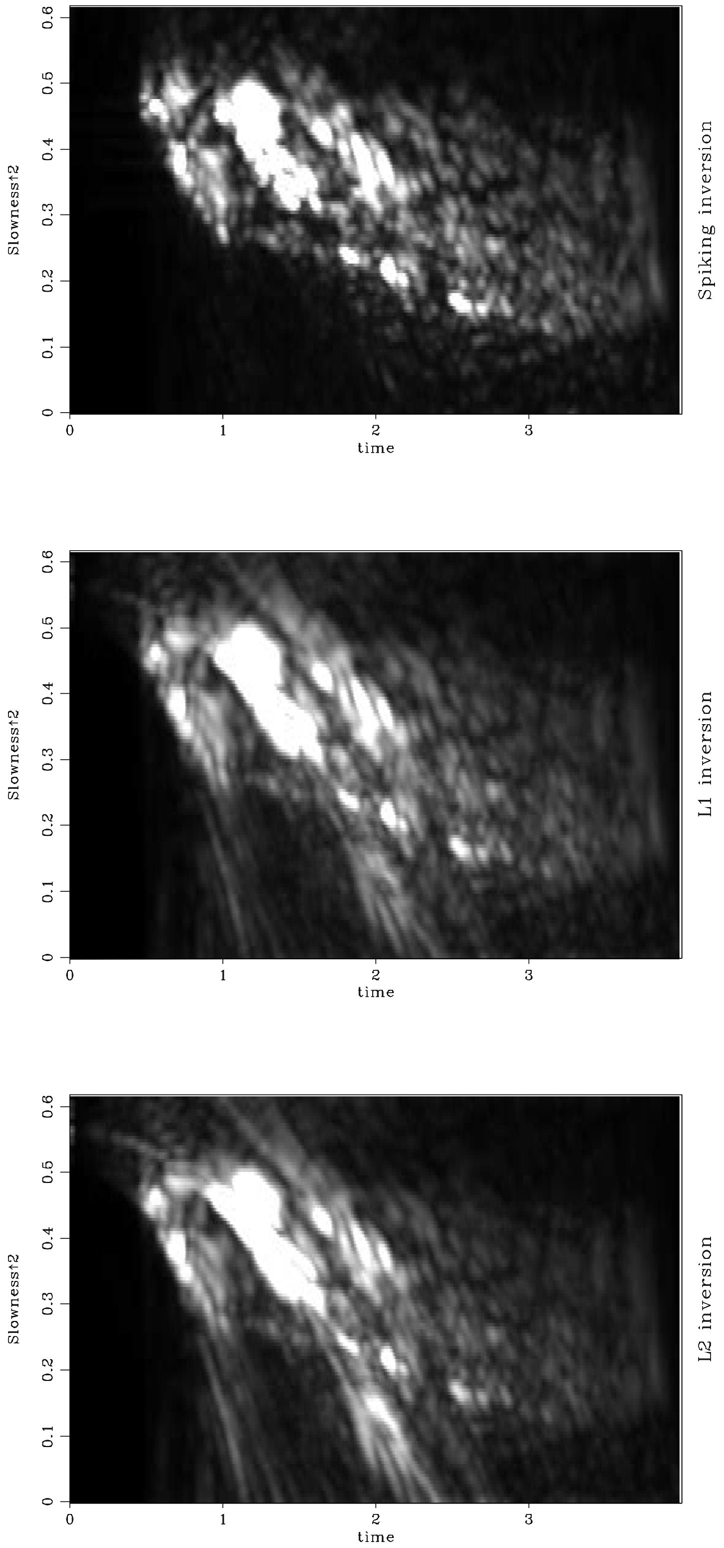

Figure 13: Envelope of the inverted data in velocity-stack space. The spiking inversion has the most cleanly separated primary and multiple trends. dave2-realenv [CR,M]

Kostov, C., 1991, Toeplitz structure in slant-stack inversion *in* Gardner, G. H. F., and Lu, L., Eds., Slant-stack processing:: Soc. Expl. Geophys., 452–455.

Tarantola, A., 1987, Inverse problem theory: Elsevier.