GEOPHYSICS. VOL. 77, NO. 1 (JANUARY-FEBRUARY 2012); P. V1–V9, 8 FIGS.
10.1190/GEO2011-0099.1

Hyperbolic estimation of sparse models from erratic data

Yunyue Li[1], Yang Zhang[1], and Jon Claerbout[1]

ABSTRACT

We have developed a hyperbolic penalty function for image estimation. The center of a hyperbola is parabolic like that of an l_2 norm fitting. Its asymptotes are similar to l_1 norm fitting. A transition threshold must be chosen for regression equations of data fitting and another threshold for model regularization. We combined two methods: Newton's and a variant of conjugate gradient method to solve this problem in a manner we call the hyperbolic conjugate direction (HYCD) method. We tested examples of (1) velocity transform with strong noise (2) migration of aliased data, and (3) blocky interval velocity estimation. For the linear experiments we performed in this study, nonlinearity is introduced by the hyperbolic objective function, but the convexity of the sum of the hyperbolas assures the convergence of gradient methods. Because of the sufficiently reliable performance obtained on the three mainstream geophysical applications, we expect the HYCD solver method to become our default method.

INTRODUCTION

In the world of geophysics, conjugate gradient methods are widely used for their simplicity, reliability, and fast convergence. Traditionally, we use l_2 norm to measure the data fitting and modeling regularization.

When least-squares (l_2) data-fitting is changed to least absolute values (l_1) data-fitting, infinite outliers may be tolerated. This is called "robustness" (Huber, 1964; Claerbout and Muir, 1973; Darche, 1989; Nichols, 1994; Guitton, 2005; Candés et al., 2006). At the same time, model regularization using l_1 norm leads to sparse models. (Valenciano et al., 2004; Donoho, 2006b).

Despite numerous l_1 optimization algorithms and their applications in the community of compressive sensing and computer science (Schmidt et al., 2007; Candés et al., 2006; Donoho, 2006a), we realize that for most of the geophysical applications, pure l_1-norm objective function is not desirable because tiny residuals always have as large an effect as giant ones. Instead, we seek merely to preserve the desirable l_1 characteristics to solutions of large problems such as image estimation. This led us to consider the hyperbolic penalty function that is l_2-like for small residuals and l_1-like for large ones. This penalty function has also been called the "hybrid norm" (Bube and Langan, 1997).

Previously, we solved problems requiring robustness and sparseness by the method of iteratively reweighted least squares (IRLS) (Gersztenkom et al., 1986; Guitton and Verschuur, 2004; Daubechies et al., 2010), a method that is cumbersome because parameters related to numerical analysis are required, although we have little theoretical guidance how to choose them, with each application requiring experimentation to learn. Another widely used standard optimization package for large-scale optimization problem, limited memory variation of the Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm (Liu and Nocedal, 1989) has recently included the Orthant-Wise limited-memory quasi-Newton (QWL-QN) method (Andrew and Gao, 2007) to meet the needs for l_1-type regularization in the model space. However, L-BFGS requires a differentiable function to measure the data fitting, which makes l_1 data fitting objective not welcomed by this family of methods.

Our experience shows that we need two different hyperbolic penalty functions, one for the data fitting objective function, the other for the model styling objective function. In this paper, we use the terminology — model styling — instead of model regularization to honor the subjectivity when choosing the regularizor. Each objective function requires a threshold of residual, let us call it R_d for the data fitting, and R_m for the model styling. Instead of being the result of numerical analysis, the meaning of the thresholds R_d and R_m is quite physical. Here are two examples: For a shot gather with about 30% of the area saturated with ground roll, choose R_d around the 70th percentile of the fitting residual. Sometimes, geologists prefer earth to be blocky with different lithologies. Therefore, we seek earth models that are as blocky as the geological requirements. In other words, we seek earth models whose

Manuscript received by the Editor 11 March 2011; revised manuscript received 3 August 2011; published online 1 February 2012.

[1] Stanford University, Department of Geophysics, Stanford, California, USA. E-mail: yunyue.li@sep.stanford.edu; yang@sep.stanford.edu; jon@sep.stanford.edu.

derivatives are spiky. For blocks about 20 mesh points long the spikes should average about 20 points apart. Thus about 95% of the residuals should be in the l_2 area with only about 5% in the l_1 area, allowing 5% of the spikes to be of unlimited size. This is an R_m at about the 95th percentile of model styling residual. The subjectively best R_d and R_m can be found within a limited interval around these physical interpretations. These examples also enable us to conclude that in a wide variety of practical examples fitting goals for data and model need not go far from the usual l_2 norm, but they do need to incorporate some residual values out in the l_1 zone, possibly very far out in it.

In our paper, we propose a new numerical method inspired by two old ones, Newton's and a variant of conjugate gradients (known as conjugate directions). Because the objective function is in general defined by two different hyperbolas, we name our method hyperbolic conjugate direction (HYCD) method. HYCD keeps the simplicity in methodology of the conjugate gradients methods, and only adds a little bit of cost to each conjugate direction iteration. The convexity of the hyperbolas assures the convergence. Experiments on three different applications: (1) velocity transform with strong noise, (2) migrating-aliased data, and (3) blocky interval velocity estimation demonstrate the utility and robustness of our HYCD solver.

THEORY

Two aspects of the new proposed HYCD method should be emphasized here: First, the hyperbolic penalty function defined by a threshold of the residual is the key to the l_1 characteristics; second, our combined HYCD method shares the outstanding convergence properties of the Newton and the conjugate gradient methods.

Hyperbolic penalty function

A circle $t^2 = z^2 + x^2$ seen in (t, x) space is a hyperbola with a parameter z. This suggests the penalty function $H_i^2 = R^2 + r_i^2$ where r_i is the ith residual, R is the universal constant threshold parameter, and $H(r) = \sum_i H_i$ is the penalty. Customarily, there is no penalty when the residual vanishes, so to accommodate that custom (making no fundamental change) we subtract the constant R from H. Thus, the hybrid penalty function promoted here is the origin-shifted hyperbola $H_i(r) = \sqrt{R^2 + r_i^2} - R$. The hyperbolic penalty function and its first two derivatives are

$$H_i = R\left(\sqrt{1 + r_i^2/R^2} - 1\right) = \sqrt{R^2 + r_i^2} - R, \qquad (1)$$

$$H_i' = \frac{r_i/R}{\sqrt{1 + r_i^2/R^2}} \quad (softclip), \qquad (2)$$

$$H_i'' = \frac{1}{R(1 + r_i^2/R^2)^{3/2}} > 0. \qquad (3)$$

Various scalings are possible. Here, we chose H to have the same physical units as R. With this scaling, the l_1 and l_2 limits are

$$H_i(r) = \begin{cases} |r_i| - R, & \text{if } R \ll |r| \\ r_i^2/(2R), & \text{if } R \gg |r|. \end{cases} \qquad (4)$$

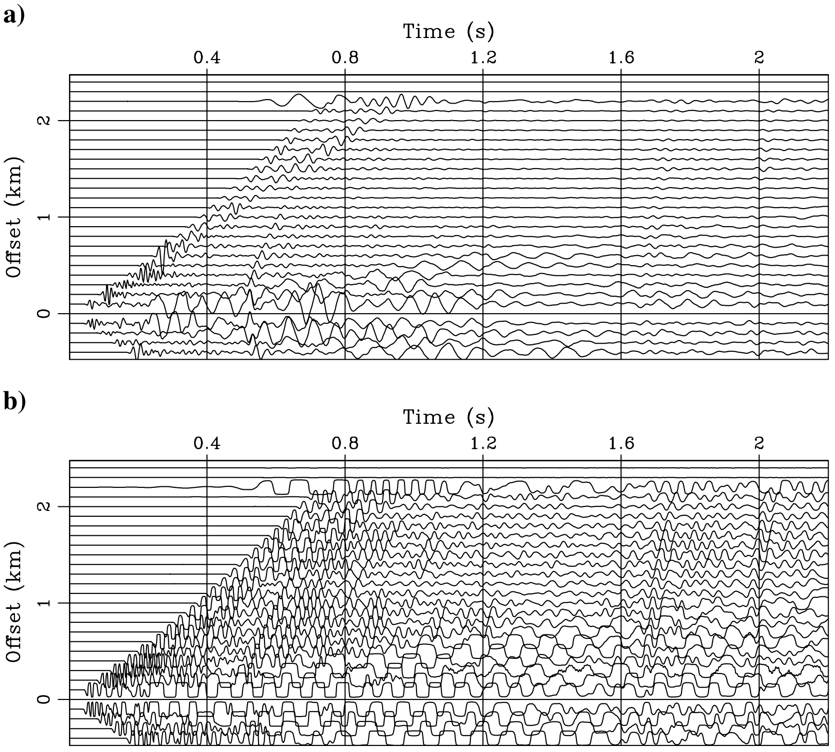

Figure 1. Reflection data $\mathbf{d}$ before (a) and after (b) soft clip $H'(\mathbf{d})$. Clipping large amplitudes enables small ones to be seen.

The second derivative $H_i''(r)$ is always positive, which assures us a unique minimum.

We often call the first derivative $H'(r)$ the "softclip" function. Equation 2 at small $|r/R|$ behaves as scaled l_2, namely, $H'(r) = r/R$. At large $|r/R|$, it behaves as l_1, namely, $H'(r) = \mathrm{sgn}(r)$. Over its whole range, $H'(r)$ behaves as a clip function, although with a softer transition around $|r/R| = 1$. As a demonstration of the soft clip function, a family of seismic reflection signals $\mathbf{d}$ shown in Figure 1 is passed through $H'(\mathbf{d})$. The intended satisfactory result is that large portions of signal are clipped (turned into "soft" rectangle functions), allowing a gain increase bringing smaller, more sinusoidal signals up into view (and up to where data fitting codes will notice them).

Conjugate directions with hyperbolic penalty function

The numerical method we use here is a synthesis of two old ones, Newton's and a variant of conjugate gradients, what we call conjugate directions (CD). It says at each iteration to descend in the plane of the gradient and the previous step. This variant is equivalent to conjugate gradients on the normal equation (Hestenes and Stiefel, 1952; M. Saunders, personal communication, 2011). Therefore, it would solve a linear regression exactly in N iterations.

When the hyperbolic penalty function is merged into the conjugate directions, we deal with the nonparabolicity by the Newton's method. Within each CD iteration, we make a quadratic Taylor approximation to the hyperbolic penalty function, and move to the predicted minimum. We iterate this Taylor approximation and small minimization process until converge. See Appendix A for details about the quadratic approximation and Appendix B for details about the plane search.

When might we get in trouble? Recall each residual has its own Taylor series. Even a residual far out in the l_1 area may move a significant distance. It is a bad fit if the residual jumps from one polarity to another. But even then, individuals far out in the l_1 area do not individually put a large force on the solution. If there are not too many such residuals, we may expect reasonable behavior, and this is what we have experienced. None of the three mainstream examples considered after the theory section required us to work near that limit. Each example had some residuals far out in the l_1 limit, but none had very many in the l_1 limit. Even if we have to work near the l_1 limit, we can always degrade back to the steepest decent update scheme which guarantees to decrease the cost function although not most efficiently. Should we find ourselves in trouble with our method, it would most certainly be at early iterations. This alerts us to giving attention to the initial solution guess — an issue safely be ignored in linear problems, but not for us.

VELOCITY ANALYSIS WITH STRONG NOISE

Velocity analysis is one the most critical and problemetic procedures in seismic exploration industry. In data with noise bursts, velocity analysis is prone to error and even unrealistic results. Therefore, to handle this problem robustly, we formulate velocity analysis as an inversion problem as follows:

$$\min_{\mathbf{m}} H_d(\mathbf{Fm} - \mathbf{d}), \tag{5}$$

where $\mathbf{F}$ is the modeling operator, whose adjoint operator is the slowness scan operator; $\mathbf{m}$ is the slowness field, $\mathbf{d}$ is the data we collect after one shot, and H_d denotes a hyperbolic penalty function with a threshold R_d.

Figure 2a shows a shot gather with t^2 gain from Yilmaz's data set. There are two distinct types of noise in these data: First is the linear noise caused by all kinds of surface waves, which can be attenuated by taking advantage of their physical properties; second is the abnormally high-amplitude bursty noise at the near offsets, which is difficult to fit into a physical model.

Figure 2 shows the inversion results of different methods. Because of the existence of the high-amplitude noise at the near-offset, a velocity scan without inversion yields no meaningful result (Figure 2b and 2c). In the result of the l_2 inversion (Figure 2d), the horizontal stripes contain the dominant energy, making it difficult to identify the velocity trend for the early time. In the reconstructed data from l_2 inversion (Figure 2e), large noise on the near-offset trace has spread to neighboring traces.

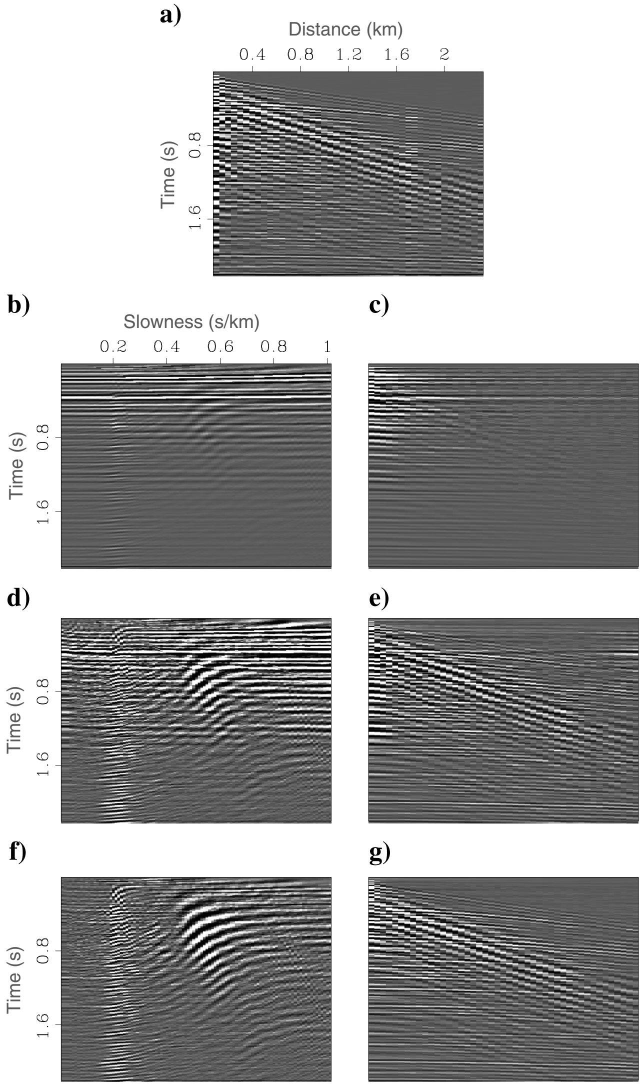

Figure 2. Inversion results of different methods. Panels in the left column are the results of velocity scans, and panels on the right are the corresponding reconstructed data. First row: input data; second row: no inversion is applied (adjoint only); third row: inversion results of l_2; bottom row: inversion results of HYCD.

For comparison, we clip Figure 2f and 2g to the same value as Figure 2d and 2e, respectively. Obviously, the velocity scan in Figure 2f shows clear velocity trends, and the near-offset bursty noise in the reconstructed data (Figure 2g) is reduced because the inversion has given more attention to the rest of the data.

ALIASED DATA MIGRATION

Kirchhoff migration was widely used before the era of wave-equation migration for marine data, and is still the principal migration method for land data. It always involves summing over or spreading along certain traveltime surfaces in 3D, which reduce to curves in 2D. For the purpose of testing our solver, we define the forward operator to be the Kirchhoff modeling operator, whose adjoint is the traditional Kirchhoff migration operator.

We formulate the inversion problem as follows:

$$\min_{\mathbf{m}} \|\mathbf{Fm}-\mathbf{d}\|_2 + \epsilon H_m(\mathbf{m}), \tag{6}$$

where $\mathbf{F}$ is the forward Kirchhoff modeling operator, $\mathbf{m}$ is the subsurface reflectivity model, $\mathbf{d}$ is the seismic response recorded at the surface and H_m denotes the hyperbolic penalty function with a threshold R_m. The second term in equation 6 is a damping term, where the hyperbolic measure is applied to retrieve the sparse model. Trad (2003) had a similar fomulation, using a model-space diagonal weighting matrix to impose the sparsity constraint.

In field acquisition, data are often irregular and aliased in space. This problem is more severe in the crossline direction. To illustrate the problem, Figure 3a shows an example of highly aliased and irregularly sampled hyperbolas. The aliasing makes the inversion problem an underdetermined problem; therefore, the result of the inversion relies heavily on the regularization. With the model space sampling being 128×128, the sampling of data space is only 128×16. Same as the previous example, we experiment with l_2 and HYCD method to compare their results.

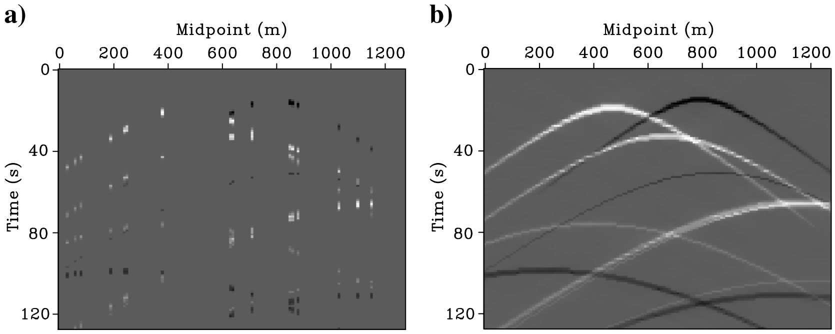

Figure 3. (a) Highly aliased and nonuniformly sampled hyperbola. Input data for the Kirchhoff inversion. (b) Reconstructed data from the HYCD inversion.

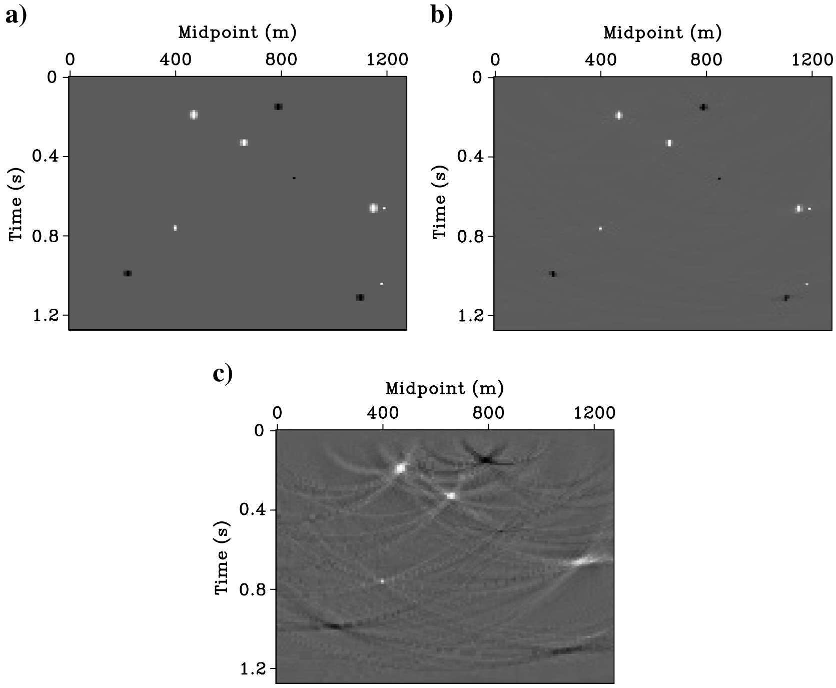

Figure 4. (a) True sparse reflectivity model, (b) inversion result of HYCD, and (c) inversion result of l_2.

Figure 4 shows the original model and the inversion results with both schemes. The results show that HYCD is superior for retrieving the spiky result that resembles the original model the most. Although severely aliased, the inversion result recovers the exact position and most of the amplitude. Notice how close the two spikes sit next to each other at 0.7 s and 1150 m in Figure 4a. Check the distinct result of HYCD and the smeared result of l_2 in Figure 4b and Figure 4c, respectively. Figure 3b shows the reconstructed data from the HYCD solver. The original data is accurately recovered. This superior result given by HYCD suggests that by properly choosing the model regularization, we can overcome the aliasing problem in the presence of a sparse model.

BLOCKY INTERVAL VELOCITY ESTIMATION

The Dix equation (Dix, 1952) finds interval velocities from root-mean-square (rms) velocity, which is picked during velocity scanning in prestack seismic data (Example 1). The equation can be written as

$$v^2_{\mathrm{int}(k)} = kV^2_k - (k-1)V^2_{k-1}, \tag{7}$$

or

$$\sum_{i=1}^{k} v^2_{\mathrm{int}(k)} = kV^2_k, \tag{8}$$

where v_{int} is interval velocity, V is rms velocity, and k is the sample number, which can be regarded as traveltime depth. Direct calculation of the interval velocity from equation 7 can easily yield wildly unreasonable results because of the error in the picked rms velocity. Therefore, it is necessary to solve this problem as a regularized inversion. The problem is linear if we chose the unknown to be interval velocity squared (v_{int}^2), instead of the interval velocity itself (v_{int}).

Thus we can formulate the Dix inversion problem as follows

$$\min_{\mathbf{m}} H_d(\mathbf{W_d}(\mathbf{Fu}-\mathbf{d})) + \epsilon H_m(\mathbf{D_z u}), \qquad (9)$$

where H_d and H_m denote the hyperbolic measure with different thresholds for data residual and model residual, respectively. The first term in equation 9 represents the data-fitting goal, where $\mathbf{u}$ is the squared interval velocity we are inverting for, $\mathbf{d}$ is the known data computed from the rms velocity, $\mathbf{F}$ is the causal integration operator and $\mathbf{W}_d$ is a data residual weighting function, which is proportional to our confidence in the rms velocity. The second term in equation 9 is the model-styling goal, where $\mathbf{D}_z$ is the vertical derivative of the velocity model and ϵ is the weight controlling the strength of the regularization.

The input rms velocity with 1000 samples is shown in Figure 5. It is obvious that the violent variation at the end of the trace is not realistic. Thus, we use the hyperbolic norm to ignore the large residuals in the data fitting, which are considered to be noise. At the same time, to obtain a blocky interval velocity model, the large residual in the derivative of the interval velocity should be "invisible" to the measure. Therefore, the hyperbolic norm on the model styling is the best choice. For illustration, we have chosen a model with homogeneous blocks. Should one prefer zones of linear trend, it is a matter of changing the model threshold and ϵ.

To compare the inversion results, we also use l_2 solver on the same data with comparable parameters. The inversion results are shown in Figure 6. The left column shows the inverted interval velocity, while the right column shows the corresponding reconstructed rms velocity. The result shows that compared with the l_2 results, the HYCD successfully retrieves the blocky velocity model, and the corresponding reconstructed rms velocity contains less noise while keeping the trend of the original data.

Parameter tuning and sensitivity analysis

In this subsection, we discuss the parameter tuning and the sensitivity analysis for HYCD method. In this example of interval velocity estimation, we need to choose thresholds for model fitting and data fitting. In general, the thresholds are subjective choices. However, the model and data statistics can guide us to a small range of the parameters.

In the input rms velocity, we notice that the picking noise causes fluctuations in all scale. Therefore, we decide to treat at most 30% of the data points with less importance. Then the data threshold quantile is set to be between 0.70 and 0.99. For the model threshold, smaller model quantile yields a blockier model. Therefore, we test the model threshold quantile from 0.35 to 0.99. The tested results are shown in Figures 7 and 8. Notice that in Figure 7, the model blockiness keeps constant due to the fixed model quantile, but the variation at the end of the time series is more significant when more data residual is appreciated by increased data quantile. Notice the inversion results in Figure 8 converge to the l_2 solution with increasing model quantile. The inversion result by HYCD in Figure 6a is produced using data quantile at 0.83 and model quantile at 0.53.

From the inversion tests, we can conclude that the HYCD method is not sensitive to the parameters, and the inversion results will evolve stably and smoothly as the user adjust the parameters. The subjectively "best" result is then picked by users according to their geological assumptions.

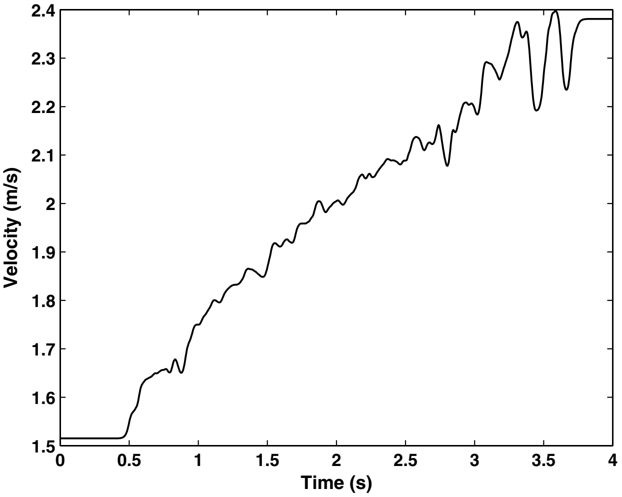

Figure 5. Input 1D rms velocity.

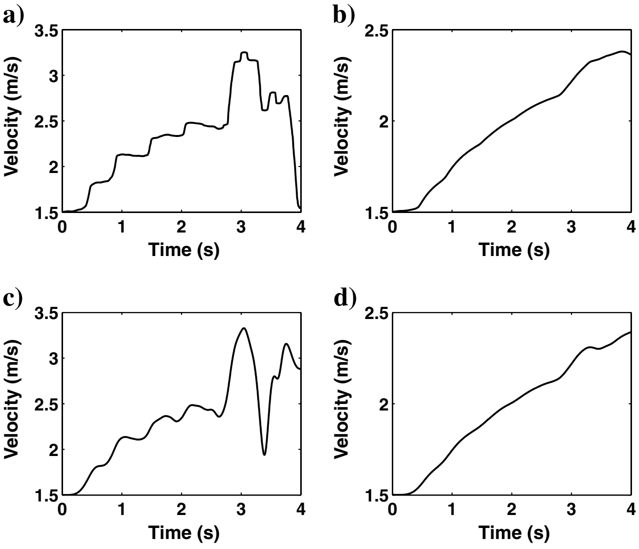

Figure 6. Comparison of the inversion results. Panels on the left column are the estimated interval velocity, and panels on the right are the corresponding reconstructed rms velocity. Top panels: inversion results of HYCD; bottom panels: inversion results of l_2. Notice that although the reconstructed rms velocity from these two methods (b and d) are very similar, the interval velocity from HYCD (c) is more blocky than the l_2 result (a).

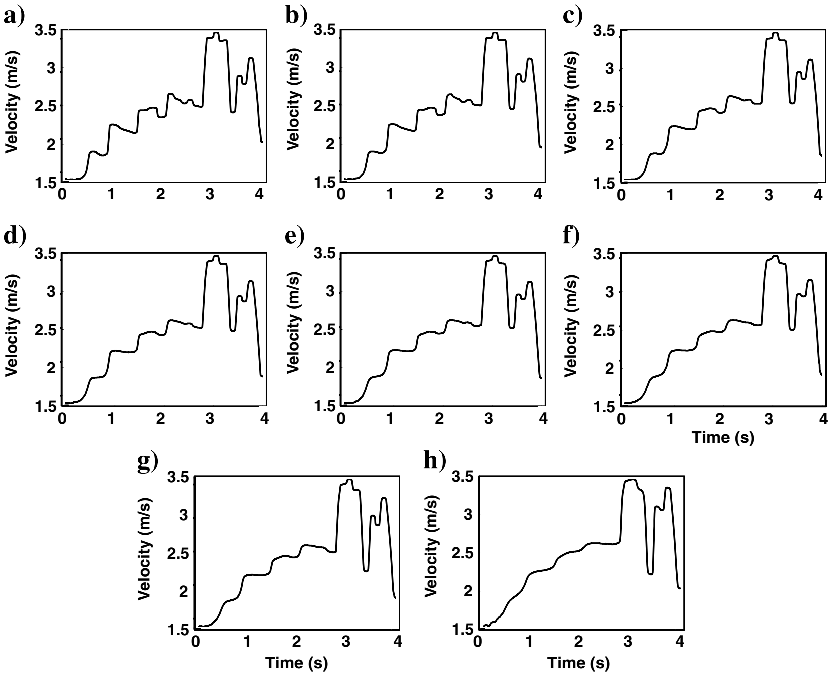

Figure 7. Inversion result using different data quantile (qnt_d). (a) $qnt_d = 0.70$; (b) $qnt_d = 0.75$; (c) $qnt_d = 0.80$; (d) $qnt_d = 0.83$; (e) $qnt_d = 0.85$; (f) $qnt_d = 0.88$; (g) $qnt_d = 0.90$; (h) $qnt_d = 0.99$. Model quantile is fixed at 0.53.

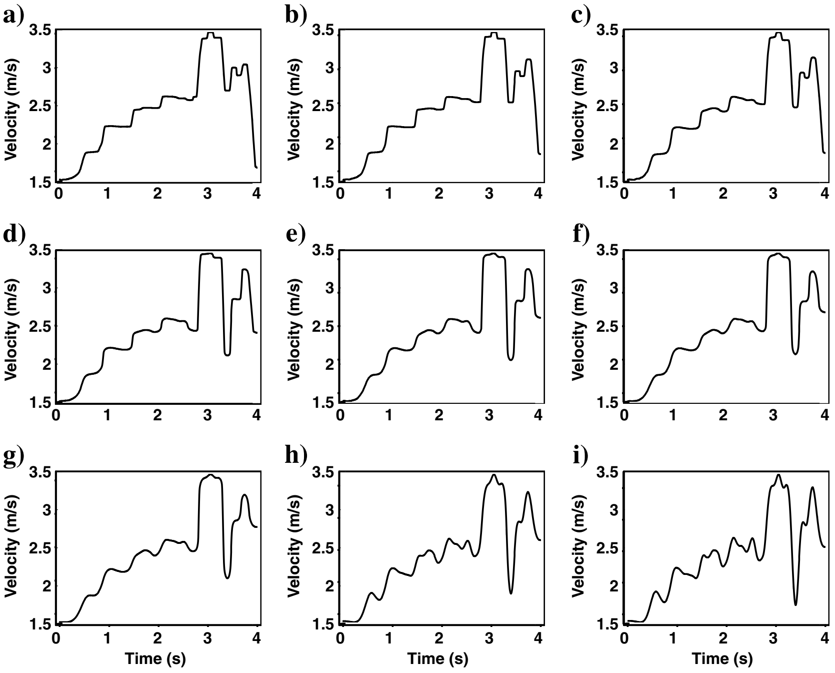

Figure 8. Inversion result using different model quantile (qnt_m). (a) $qnt_m = 0.35$; (b) $qnt_m = 0.45$; (c) $qnt_m = 0.53$; (d) $qnt_m = 0.68$; (e) $qnt_m = 0.76$; (f) $qnt_m = 0.81$; (g) $qnt_m = 0.88$; (h) $qnt_m = 0.95$; (i) $qnt_m = 0.99$. Data quantile is fixed at 0.83.

CONCLUSIONS

We set out to find a fast and reliable mean of dealing with erratic data and blocky models.

We developed a method based on a hyperbolic penalty function (which is a composite of l_1 norm and l_2 norm). When the physics is linear, convergence is guaranteed by the convexity of the penalty function. The speed of the convergence is beyond the scope of this paper, but experience shows that the cost of the HYCD solver is on the same order as a conventional conjugate direction solver. The two new parameters introduced by the penalty function are threshold for data residual and threshold for model residual, which must be determined according to the noise level in the data and the desired sparsity in the model, respectively.

We tested the method with field data on rms velocity estimation and Dix interval velocity estimation, and we tested Kirchhoff migration of aliased synthetic data. All results were excellent. Beside these three examples, three other applications have been done successfully using the new HYCD solver in our research group. We expect this method to become the default method in our laboratory.

In principle, we introduce extra nonlinearity into the optimization by using the hyperbola, but the hyperbolic penalty function is convex, so gradient methods are assured to lead us to a universal minimum penalty when the original problem is linear. The number of required iterations is not known theoretically, but with large image-estimation applications we commonly cease iteration long before the theoretical requirement. When doubts arise, we resolve them by initiating solutions from different locations. Although this paper investigates only the hyperbola as a penalty function, actually, the only property for HYCD to succeed is the convexity of the hyperbolas. Therefore, other convex functions might be tried.

ACKNOWLEDGMENTS

We would like to thank Robert Clapp, Antoine Guitton, Ali Almomin, and Luis Canales for fruitful discussions about the development of this solver and the choice of the test cases.

APPENDIX A

MODEL DERIVATIVES

Here is the usual definition of residual r_i of theoretical data $\sum_j F_{i,j} m_j$ from observed data d_i

$$r_i = \left(\sum_j F_{i,j} m_j\right) - d_i \qquad or \qquad \mathbf{r} = \mathbf{Fm} - \mathbf{d}. \tag{A-1}$$

Let $H(r_i)$ be a convex function ($H'' \geq 0$) of a residual mismatch r_i that grows with mismatch size and gives a measure of mismatch. The average penalty measure for mismatch measure between theory and data is

$$\bar{H}(\mathbf{m}) = \frac{1}{N} \sum_{i=1}^{N} H(r_i).$$

Let $H'(r_i)$ denote dH/dr evaluated at r_i. Define a vector $H'(\mathbf{r})$ by applying $H'()$ to each component of $\mathbf{r}$

$$H'(\mathbf{r}) = \frac{dH(r_i)}{dr_i}. \tag{A-3}$$

In the steepest-descent method, the model updates in the direction $\Delta\mathbf{m}$, the gradient of the mismatch measure of the residual. The jth element of the gradient is

$$\begin{aligned} \Delta\mathbf{m} &= \frac{\partial \bar{H}}{\partial m_j} = \frac{1}{N} \sum_i \frac{dH(r_i)}{dr_i} \frac{\partial r_i}{\partial m_j} = \frac{1}{N} \sum_i H'(r_i) F_{i,j} \\ &= \frac{1}{N} \mathbf{F}^T H'(\mathbf{r}). \end{aligned} \tag{A-4}$$

The gradient vanishes at the minimum giving "normal equations" $\mathbf{0} = \mathbf{F}^T H'(\mathbf{r})$ like those with the simple least-squares method. In other words, at minimum average mismatch, the fitting functions (rows of $\mathbf{F}^T$) are orthogonal (normal) to the soft clipped residual.

Define a model update direction by the gradient $\Delta\mathbf{m} = \mathbf{F}^T H'(\mathbf{r})$. Because $\mathbf{r} = \mathbf{Fm} - \mathbf{d}$, the residual update direction is $\Delta\mathbf{r} = \mathbf{F}\Delta\mathbf{m}$. To find the distance α to move in those directions

$$\mathbf{m} \leftarrow \mathbf{m} + \alpha\Delta\mathbf{m}, \tag{A-5}$$

$$\mathbf{r} \leftarrow \mathbf{r} + \alpha\Delta\mathbf{r}, \tag{A-6}$$

choose the scalar α to minimize the average penalty

$$\bar{H} = \frac{1}{N} \sum_i H(r_i + \alpha\Delta r_i). \tag{A-7}$$

The sum in equation A-7 is a sum of "dishes," shapes between l_2 parabolas and l_1 V-shaped curves. The ith dish is centered on $\alpha = -r_i/\Delta r_i$. It is steep and narrow if Δr_i is large, and low and flat where Δr_i is small. The sum of convex functions is convex. There are no local minima. equation A-7 now is a 1D function of α. The minimum is found by the Newton method.

Express $H_i = H(r_i + \alpha\Delta r_i)$ in a Taylor expansion keeping only the first three terms. Let H_i' and H_i'' be first and second derivatives of $H(r_i)$ at r_i. Then equation A-7 becomes a familiar least-squares problem

$$\bar{H} = \frac{1}{N} \sum_i (H_i + \alpha\Delta r_i H_i' + (\alpha\Delta r_i)^2 H_i''/2). \tag{A-8}$$

To find α, set $d\bar{H}/d\alpha = 0$

$$0 = \frac{d\bar{H}}{d\alpha} = \frac{1}{N} \sum_i (\Delta r_i H_i' + \alpha(\Delta r_i)^2 H_i''), \tag{A-9}$$

and solve for α

$$\alpha = -\frac{\sum_i \Delta r_i H_i'}{\sum_i (\Delta r_i)^2 H_i''}, \tag{A-10}$$

which resembles the familiar least-squares case $H = r^2/2$, $H_i' = r_i$, and $H'' = 1$, where α comes out $\alpha = -\sum_i \Delta r_i r_i / \sum_i (\Delta r_i)^2$.

Now move the solution $\mathbf{m}$ to $\mathbf{m} + \alpha\Delta\mathbf{m}$ and likewise update the residuals. At the new location the convex function and its derivatives (H_i, H_i', H_i'') take new values. Thus, we can find another α to update a second time, or more. This is line search. This is cheap. If the residual grows instead of shrinking, then $\alpha \leftarrow \alpha/2$, etc. Eventually, we get to the bottom of the line we are scanning and are ready for a new line, so we pay the money to compute a new $\Delta\mathbf{m} = \mathbf{F}^T H'(\mathbf{r})$ and $\Delta\mathbf{r} = \mathbf{F}\Delta\mathbf{m}$. Finally, geophysical applications sometimes involve costly operators (e.g., migration), sometimes cheap ones (e.g., gradient). For the costly ones we do more Newton iterations; for the cheap ones fewer.

APPENDIX B

PLANE SEARCH

The most universally used method of solving immense linear regressions such as imaging applications is the conjugate gradient (CG) method. It has the remarkable property that in the presence of exact arithmetic, the exact solution is found in a finite number of iterations. A simpler method with the same property is the CD method. It says not to move along the gradient direction line, but somewhere in the plane of the gradient and the previous step taken.

Similar to Bube and Nemeth (2007), we determine the scaling factors for the gradient and the previous step iteratively.

With the steepest-descent method, we improve the model $\mathbf{m}$ by adjusting a single scalar parameter α that multiplies $\Delta\mathbf{m} = \mathbf{g} = \mathbf{F}^T\mathbf{r}$. With the hyperbolic penalty function this becomes $\Delta\mathbf{m} = \mathbf{g} = \mathbf{F}^T H'(\mathbf{r})$. Extending to the CD method there are two parameters, α and β, and two vectors. One vector is the gradient vector $\mathbf{g}$. The other vector is the previous step $\mathbf{s}$. These vectors may be viewed in data space or in model space. We are going to take linear combinations of $\mathbf{g}$ and $\mathbf{s}$ in both spaces and we need to choose notation for distinguishing them.

We need some unconventional notation. In matrix analysis, lower-case letters are conventionally vectors and upper-case letters are matrices. But in Fourier analysis, lower-case letters become upper-case upon transformation. By analogy, we handle $\mathbf{g}$ and $\mathbf{s}$ this way: Keep using bold capitals for operators but now use ordinary italic for vectors with model space being lower-case italic and data space being upper-case italic.

At the kth iteration, we update the model m with gradient g and step s where

$$s_{k+1} = \alpha_k g_k + \beta_k s_k, \tag{B-1}$$

and the scalars α and β are yet to be found. The corresponding change of the residual in data space is found by multiplying through with $\mathbf{F}$:

$$\Delta r = S_{k+1} = \mathbf{F}s_{k+1} = \mathbf{F}(\alpha_k g_k + \beta_k s_k) \tag{B-2}$$

$$= \alpha_k \mathbf{F} g_k + \beta_k \mathbf{F} s_k, \tag{B-3}$$

$$\Delta\mathbf{r}(\alpha,\beta) = \alpha_k G_k + \beta_k S_k. \tag{B-4}$$

In standard l_2 optimization, there is a 2×2 matrix to solve for (α,β). We proceed here in the same way with the hyperbolic penalty function. Here, we are embedded in a giant multivariate regression in which we have a bivariate regression (two unknowns). From the multivariate regression, we are given three vectors in data space, $\bar{r}_i$, G_i, and S_i. Our next residual will be this perturbation of the old one:

$$r_i = \bar{r}_i + \alpha G_i + \beta S_i. \tag{B-5}$$

Minimize the average penalty by variation of (α,β):

$$\bar{H}(\alpha,\beta) = \frac{1}{N}\sum_i H(\bar{r}_i + \alpha G_i + \beta S_i). \tag{B-6}$$

Let the coefficients (H_i, H_i', H_i'') refer to a Taylor expansion of $H(r)$ in small values of (α,β) about $\bar{r}_i$. Each residual of each data point has its own Taylor series fitting the hyperbola at its own location. So all the residuals that do not move far have good approximations. Then equation B-6 becomes

$$\bar{H}(\alpha,\beta) = \frac{1}{N}\sum_i H_i + (\alpha G_i + \beta S_i)H_i' + (\alpha G_i + \beta S_i)^2 H_i''/2. \tag{B-7}$$

There are two unknowns, (α,β) in a quadratic form. Set $d\bar{H}/d\alpha = 0$ and $d\bar{H}/d\beta = 0$ getting

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \sum_i H_i' \begin{bmatrix} G_i \\ S_i \end{bmatrix} + H_i'' \left\{ \begin{bmatrix} \frac{\partial}{\partial\alpha} \\ \frac{\partial}{\partial\beta} \end{bmatrix} (\alpha G_i + \beta S_i) \right\} (\alpha G_i + \beta S_i), \tag{B-8}$$

resulting in a 2×2 set of equations to solve for α and β.

$$\left\{ \sum_i H_i'' \left[\begin{pmatrix} G_i \\ S_i \end{pmatrix} \begin{pmatrix} G_i & S_i \end{pmatrix} \right] \right\} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = -\sum_i H_i' \begin{bmatrix} G_i \\ S_i \end{bmatrix}. \tag{B-9}$$

New (to us) in equation B-9 is the presence of H' and H''. (Previously with the l_2 penalty function, we had $H_i' = r_i$ and $H_i'' = 1$.) The solution of any 2×2 set of simultaneous equations is well-behaved with minor exceptions. The determinant would vanish if the gradient was in the same direction as the previous step, but that would imply the previous step did not go the proper distance. Experience shows the determinant does vanish when all the inputs are zero, and it may vanish if we do so many iterations that we should have stopped already, in other words, when the gradient and previous step are both tending to zero.

After updating $\mathbf{m} \leftarrow \mathbf{m} + \alpha\mathbf{g} + \beta\mathbf{s}$ and updating the residuals, at the new residual location, the values of (H_i, H_i', H_i'') have changed. Thus, we repeat updating α and β a second time or more. Do not update $\mathbf{s}$ yet. Eventually, we found the best location in the plane. We have finished the plane search. It is usually cheap. Now it is time to get a new plane. For the new plane, we update $\mathbf{s}$ and we pay the money (run the operator $\mathbf{F}^T$) to compute a new $\mathbf{g} = \mathbf{F}^T H'(\mathbf{r})$. This is the nonlinear CD method. With $H(r)$ being the hyperbola, we call it the HYCD method.

In our experience, the presence of some residuals out in the l_1 region do not greatly increase the number of iterations compared to the usual l_2 parabolic penalty function. Should anyone choose a threshold R so small that it drives many of the residuals into the l_1 region, convergence might be slow. While we do not doubt this might happen, we have not yet found an application that drove us to this difficulty.

REFERENCES

Andrew, G., and J. Gao, 2007, Scalable training of L^1-regularized log-linear models: Proceedings of the 24th International Conference on Machine Learning, 33–40.

Bube, K., and R. Langan, 1997, Hybrid l^1/l^2 minimization with applications to tomography: Geophysics, **62**, 1183–1195, doi: 10.1190/1.1444219.

Bube, K., and T. Nemeth, 2007, Fast line searches for the robust solution of linear systems in the hybrid l^1/l^2 and huber norms: Geophysics, **72**, no. 2, A13–A17, doi: 10.1190/1.2431639.

Candés, E., J. Romberg, and T. Tao, 2006, Stable signal recovery from incomplete and inaccurate measurements: Communications on Pure and Applied Mathematics, **59**, 1207–1223, doi: 10.1002/(ISSN)1097-0312.

Claerbout, J. F., and F. Muir, 1973, Robust modeling with erratic data: Geophysics, **38**, 826–844, doi: 10.1190/1.1440378.

Darche, G., 1989, Iterative l_1 deconvolution: SEP report, **61**, 281–302.

Daubechies, I., R. Devore, M. Fornasier, and C. S. Gntrk, 2010, Iteratively reweighted least squares minimization for sparse recovery: Communications on Pure and Applied Mathematics, **63**, 1–28, doi: 10.1002/cpa.v63:1.

Dix, C. H., 1952, Seismic prospecting for oil, Harper & Brothers.

Donoho, D. L., 2006a, Compressed sensing: IEEE Transactions on Information Theory, **52**, 1289–1306, doi: 10.1109/TIT.2006.871582.

Donoho, D. L., 2006b, For most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution: Communications on Pure and Applied Mathematics, **59**, 797–829, doi: 10.1002/(ISSN)1097-0312.

Gersztenkom, A., J. Bednar, and L. R. Lines, 1986, Robust iterative inversion for the one-dimensional acoustic wave equation: Geophysics, **51**, 357–368, doi: 10.1190/1.1442095.

Guitton, A., 2005, Multidimensional seismic noise attenuation: Ph.D. thesis, Stanford University.

Guitton, A., and D. Verschuur, 2004, Adaptive subtraction of multiples using the l_1-norm: Geophysical Prospecting, **52**, 27–38, doi: 10.1046/j.1365-2478.2004.00401.x.

Hestenes, M. R., and E. Stiefel, 1952, Methods of conjugate gradients for solving linear systems: Journal of Research of the National Bureau of Standards (United States), **49**, no. 6, 409–436.

Huber, P. J., 1964, Robust estimation of a location parameter: Annals of Mathematical Statistics, **35**, 73–101, doi: 10.1214/aoms/1177703732.

Liu, D. C., and J. Nocedal, 1989, On the limited memory BFGS method for large scale optimization: Mathematical Programming, **45**, 503–528, doi: 10.1007/BF01589116.

Nichols, D., 1994, Velocity-stack inversion using l^p norms: SEP report, **82**, 1–16.

Schmidt, M., G. Fung, and R. Rosales, 2007, Fast optimization methods for L1 regularization: A comparative study and two new approaches: European Conference on Machine Learning (ECML), Lecture Notes in Computer Science, **4701**, 286–297, doi: 10.1007/978-3-540-74958-5.

Trad, D., 2003, Interpolation and multiple attenuation with migration operators: Geophysics, **68**, 2043–2054, doi: 10.1190/1.1635058.

Valenciano, A. A., M. Brown, and A. Guitton, 2004, Interval velocity estimation using edge-preserving regularization: 74th Annual International Meeting, SEG, Expanded Abstracts, **23**, 2431–2434.