Applications of the generalized norm solver

Mandy Wong, Nader Moussa, and Mohammad Maysami

ABSTRACT

The application of a L1/L2 regression solver, termed the generalized norm solver, to two test cases, shows that it is potentially an efficient method for L1 inversion and is easy to parameterize. The generalized norm solver iterates with conjugate direction. Our first test case, the line fitting problem, shows that the generalized solver is capable of removing outliers in data. Our second test case, the 1D Galilee problem, shows that the generalized solver can produce a satisfactory "blocky" solution. In terms of parameters, a low threshold value, if giving convergent solution, gives the best result. Experience shows the optimal number of inner loop iterations is one.

INTRODUCTION

Currently, many geophysics problems are solved with least-squares (L2) model fitting because of its fast convergence, simple parametrization, and easy to understand numerical analysis. However, L2 minimization places disproportionate emphasis on large residual values. Therefore, an L1-type norm inversion technique is more appropiate for solving geophysical problems that have a "blocky" model space. One example is to do adaptive subtraction of multiples Guitton (2005) using IRLS (iterative re-weighted least squares). Other possible applications are tomography (Bube and Langan (1997)) and deconvolution of noisy data (Chapman and Barrodale 1983).

In geophysics, a popular way to run L1-type inversions is with IRLS (Gersztenkorn et al. (1986)) . Running with IRLS often improves the results. A drawback is that its computational time is considerably higher because of repeated application of the costly forward and adjoint data-fitting operator within two iterative loops. To overcome this computational deficiency while retaining the benefit of L1-type inversion, we came up with another solver that does a better job than IRLS. Claerbout (2009) developed an algorithm of that we call the **generalized norm solver**, which steps with conjugate-direction and using the Taylor series expansion. Such a solver allows us to perform inversion using the L2 or the mixed L1/L2 norm Maysami and Moussa (2009). For convenience, we will use the term 'norm' to refers to all kind of measures and norms. It is understood that a norm has a strict definition in mathematics. For the theoretical description of the solver, please refer to the report by Claerbout (2009).

Maysami and Moussa (2009) have implemented such a solver, which allows us to test its robustness in this paper. Three norms will be used in our study: the least-squares (L2), Hybrid, and Huber norms. It is worth mentioning that the theory for using the least-squares norm with our solver is exactly the same as the theory for solving the least-squares problem with conventional conjugate-direction algorithm. For the rest of this paper, we will refer the generalized norm solver with the Hybrid norm as the **hybrid solver** and the generalized norm solver with the Huber norm as the **Huber solver**.

We have applied the generalized norm solver to two test cases. The first test case recovers the equation of a straight line given data that are corrupted with Gaussian noise and spikes. We find that the generalized norm solver recovers the equation of a straight line when using either the hybrid solver or the Huber solver. The second test case is called the 1-D Galilee problem. This problem is a simplified version of a real depth sounding experiment of the Sea of Galilee and a standard test problem in SEP textbooks, (Claerbout (2008); Claerbout and Fomel (2008)).The synthetic data for this case are measurements between the water surface and the bottom of the lake. The first part of the test simply aims to recover the true depth of a 1-D lake given that the data contain occasional large spikes. The second part of the test has data corresponding to a water level that change in a step-like manner. We will refer to these kind of jumps as "drift." We aim to recover the true depth and sudden drift in data. We find that the hybrid solver always gives the best result as compared to the Huber solver and the least-squares solver.

FIRST TEST CASE: LINEAR FITTING PROBLEM

Basic formulation for the linear fitting problem

To verify the L_1-norm solver and establish its utility across several geophysical optimization problems, we tested on a series of one-dimensional fitting problems. We began with the simplest fitting problem, a line-fit estimation that would be suitably solved by least-squares line fitting in most cases.

To show the advantages of an L_1-based methodology, we injected both Gaussian and non-Gaussian noise. In general, while L_2 fitters are well suited to wide-spectrum noise, they are particularly prone to misleading or unphysical results if the noise has many spike- or burst-like. We experimented with a linear fit of a set of data plus several spikes.

The problem setup of the line-fit is similar, but simpler, than the other examples. It serves as a good explanatory case-study of the set-up of a solver. We begin with a data space, $\mathbf{d}$, representing the noisy sampling of a straight line. We intend to model this line with a very simple, two-element line model – namely, slope and intercept according to the conventional $y = \alpha x + \beta$ formula.

We construct a forward operator, $\mathbf{L}$, which implements the mapping of this model space onto our recorded data. I n the presence of interfering noise, there will be a deviation between predicted data (**Lm**) and recorded data (**d**). We have tested our solver to minimize this deviation according to L_2, Huber, and hybrid L_1/L_2 definitions. This simplified example shows the important ability of the L_1-style norms to reject large data outliers.

In this case, we formally define the model space,

$$\mathbf{m} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \tag{1}$$

where α represents thes slope and β represents the y-intercept of a line. And the line operator,

$$\mathbf{L} = \begin{bmatrix} x & 1 \end{bmatrix}, \tag{2}$$

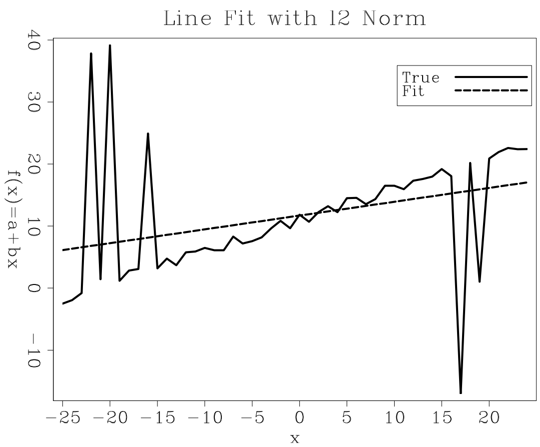
We attempt to minimize the objective functions for each defined norm:

$$\begin{aligned} r_{L_2} &= ||\mathbf{Lm} - \mathbf{d}||_{L_2}, \\ r_{L_1} &= ||\mathbf{Lm} - \mathbf{d}||_{L_1}, \\ r_{Huber} &= ||\mathbf{Lm} - \mathbf{d}||_{Huber}, \\ r_{Hybrid} &= ||\mathbf{Lm} - \mathbf{d}||_{Hybrid}, \end{aligned}$$

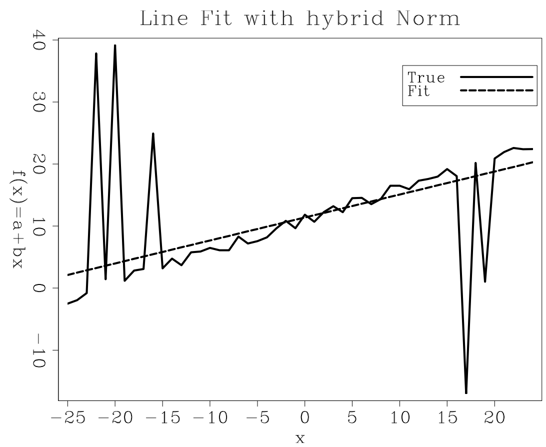
Because each norm represents a different method for computing the residual, the corresponding minimization has different behaviors with regard to the optimal modeled data. As we will show in the following sections, this impacts the ability of each optimization criterion to produce geophysically useful results in the presence of different types of interference, and for the different characteriestics of the desired function (e.g. sparseness or block-like intervals).

Results of the line-fitting problem

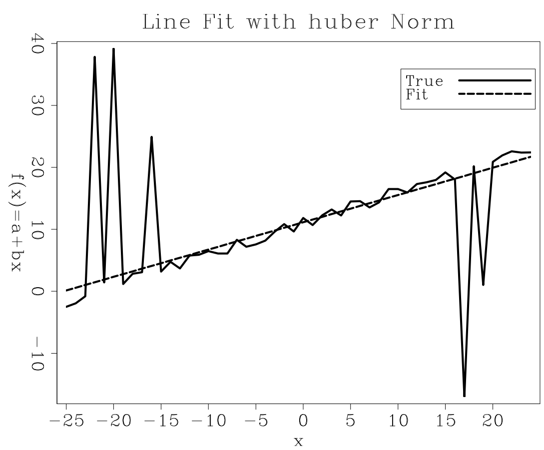
The results of solving this problem with the least-squares, hybrid, and Huber norms are shown in Figure 1. We can see that the fitted line in the L2 norm deviates from the true line due to the presence of spiked data, whereas for the Huber solver and the hybrid solver, the fitted line correctly overlaps the true line. We conclude that our trivial line-fitting example functions properly when using the L_1-type hybrid and Huber norms.



(a)



(b)



(c)

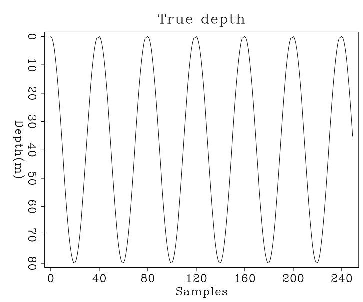
Figure 1: Line fitting using the generalized norm solver: **(a)**L_2 fitting, **(b)** hybrid norm fitting, **(c)** Huber norm fitting. Notice that the L_2 fit-line does not match the actual data trend – this illustrates the susceptibility of least-squares minimization to strong outliers (spikes), while the other norms are totally unaffected by these data points. [**ER**]

TEST CASE TWO: THE ONE DIMENSIONAL GALILEE PROBLEM

Formulation of the 1D Galilee Problem

Our next test case is the removal of spikes and drift from a 1-dimensional representation of the Galilee depth data. We constructed this experiment to be more rigorous than the earlier line-fitting and noise-removal problem. The 1D Galilee problem is a synthetic problem that originates from a depth sounding experiment on the Sea of Galilee. Imagine performing a depth-sounding experiment along a fixed track in the lake. The lake has a sinusoidal depth with blocky drift and large spikes in time. As a boat goes back and forth across this 1D lake, it is measuring the sum of the true lake depth plus the drift in time. Figure 2 shows the true depth of our 1D Galilee lake and figure 3 shows the recorded data, which is the sum of the drift function and the true depth. We have added 2 spikes as outliers in the data. These 2 spikes can be viewed as equipment failure. Note that the data covers the lake back and forth roughly 6.25 times.

Figure 2: The true depth of the Sea of Galilee along a fixed track. [**ER**]



To formulate the problem for inversion, we have set our unknown model space to be the lake depth, $\mathbf{m}$, and the drift function, $\mathbf{u}$. Data space $\mathbf{d}$ is the recorded depth as shown in Figure 3. Our data fitting goal can be defined as

$$0 \approx \mathbf{Lm} + \mathbf{u} - \mathbf{d}, \tag{3}$$

where $\mathbf{L}$ is the binning operator that matches the data acqusition in time to its corresponding location in space. For a lake with 4 grid points and 6 data points, equation 3 would look like this:

$$0 \approx \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{bmatrix} + \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} - \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_6 \end{bmatrix}.$$

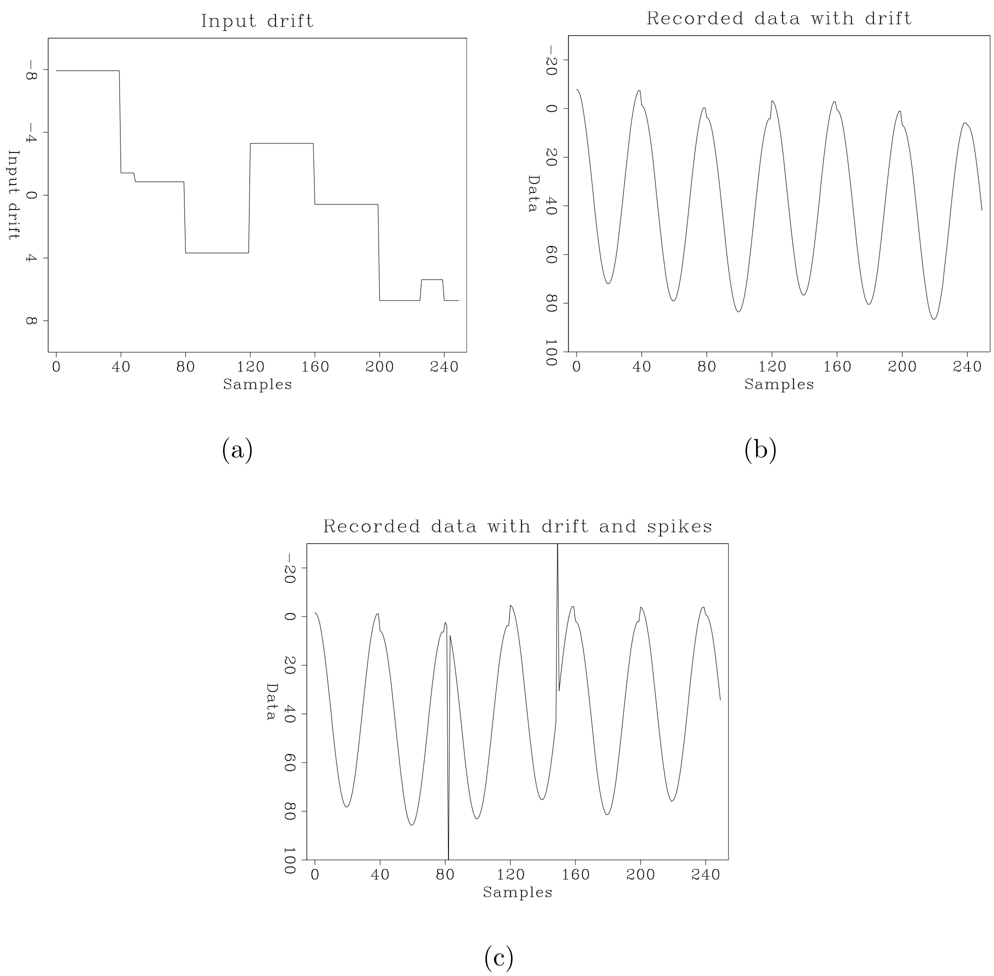


Figure 3: **(a)** The drift as a function of aquisition time. **(b)** The recorded data, which is the sum of the true lake depth and drift. **(c)** The recorded data with two outliers. The two spikes are added to the data to account for equipment failure.[**ER**]

Equation 3 by itself is an under-determined problem, because there are more unknowns than the recorded data points. If there are n_d data points and the lake has n_m grid points, then the model space has a dimension of $n_m + n_d$, because we are solving for both lake depth and drift in time. The data space has a dimension of n_d. To introduce more constraints, we can add a regularization by requiring the drift function $\mathbf{u}$ to be smooth,

$$0 \approx \frac{d}{dt}\mathbf{u}. \tag{4}$$

It is worth pointing we only expect good results when we run this regularization with $L1$ or $L1$-type norms, as $L2$ smoothing will wipe out the "blockiness" in the drift function, which is part of the model space. To illustrate the limitation of least-squares fitting, I will first show the result of applying inversion to the 1D Galilee problem.

Result of least-squares inversion of the 1D Galilee Problem

Least-squares inversion looks a for solution in the model space that minimizes the square of the residual. I first run un-regularized inversion on spike-free but drifted data. That means using the data-fitting goal in equation 3 to fit the data shown from Figure 3 (b). After L2 fitting, the estimated depth is shown in figure 4 (a), and the estimated drift is shown in figure 4 (b). An interesting observation from the un-regularized L2 inversion of the non-spike data is that it gives very good estimate of the lake depth and drift. Although the problem is still under-determined, the amount of data collected is sufficient enough to determine the relative jumps from each pass in the lake. As mentioned before, the data cover the lake back and forth roughly 6.25 times.

At this point, further attempts to solve this problem seem redundant, as we are getting a nearly perfect result. However, the result is different when I re-run the un-regularized inversion (equation 3) on the spiked and drifted data (Figure 3 (c)). After L2 fitting, the estimated depth is shown in figure 4 (c),and while the estimated drift is shown in Figure 4 (d). This time, the fitted depth deviates from the true depth, with segments that are clearly affected by spikes. The fitted drift function shows erratic jumps. This is because by minimizing the square of the residual, L2 inversion emphasizes large spikes in data. In a situation like this, L1 or L1-type inversion like the hybrid and Huber solvers should give better results. This is because minimizing the absolute value of the residual puts less emphasis on large spikes in data. This assertion will be verified when I apply the hybrid and the Huber solvers in the next section.

In addition to the two un-regularized least-squares examples above, I also ran a regularized inversion on the spike-free but drifted data. That means using the fitting goals in equation 3 and equation 4 on the data shown in Figure 3 (b). After L2 fitting, the estimated depth is shown in Figure 4 (e), and the estimated drift is shown in Figure 4 (f). The regularized drift in this case, shown in Figure 4 (f) , is a smoothed version of the un-regularized drift as shown in Figure 4 (b). The smoothing is as expected because of the type of regularization used. One conclusion is that when the data is spike-free, the L1-type solver is unnecessary, because Figure 4 (b) shows that we are obtaining a satisfactory result with L2 unregularized fitting. However, Figure 4 (d) demonstrates that L1-type inversion is needed for data containing large spikes. The next step is to see how the generalized L1 solver handle this problem using the hybrid and the Huber norms.

Result of the 1D Galilee Problem using the Generalized norm solver

We begin with the simplest test for any L1-type solver, which is the ability to remove outliers in the data. When there is no water-level drift in the Galilee sounding experiment, the data is affected only by random spikes (non-Gaussian noise). The ideal fitted output would be to recover the true depth. The result is shown in figure 5. When we apply the generalized solver to the problem, L2 gives the worst result, as expected, because it cannot isolate outliers from the overall fitting goal. The Huber criteria gives an intermediate result, while the hybrid minimization criteria gives the best result, almost completely recovering the true depth of the lake without distortion.

Next, I apply the solver to the full 1-D Galilee problem, which is the inversion using fitting goals from equation 3 and 4 of data that including drifts and spikes, as shown in Figure 3 (b). The hybrid norm gives the most satisfying result, as summarized in Figure 6. There are two parameters that can be adjusted in this problem: `epsilon`, which describes the level of regularization, and `percentile`, which describes the transition point between the $L2$ and the $L1$ measure (per the defining equations of each norm). Please refer to Claerbout (2009) and Maysami and Moussa (2009) for the definition of each norm. In general, a small `epsilon` means less weight is placed on the regularization goal, meaning a less smooth result. The `percentile` parameter allows us to configure the degree of confidence that any randomly-selected data element is a statistical outlier, placing it within either the $L1$ or the $L2$ fitting goal. If a low percentile is set, it indicates that we believe most of the data should be fitted with L_1. For this problem, we have set the percentile to be as low as possible without having a divergent solution. For the Huber solver, the limit is $percentile \approx 0.3$. We found that the hybrid solver has a better tolerance, and the limit for it is $percentile \approx 0.1$.

In terms of convergence, there are two iterative parameter. The first one is `niter`, which controls then number of applications of the costly forward and adjoint operators. Another one is `psiter` which corresponds to the number of iterations of the inner loop that determines the step sizes, α and β, in our conjugate direction scheme. We found that when $psiter = 1$, we obtain the best result in the full 1-D Galilee problem. For the outer loop parameter, `niter`, the $L2$ solver converges in just 16 it-

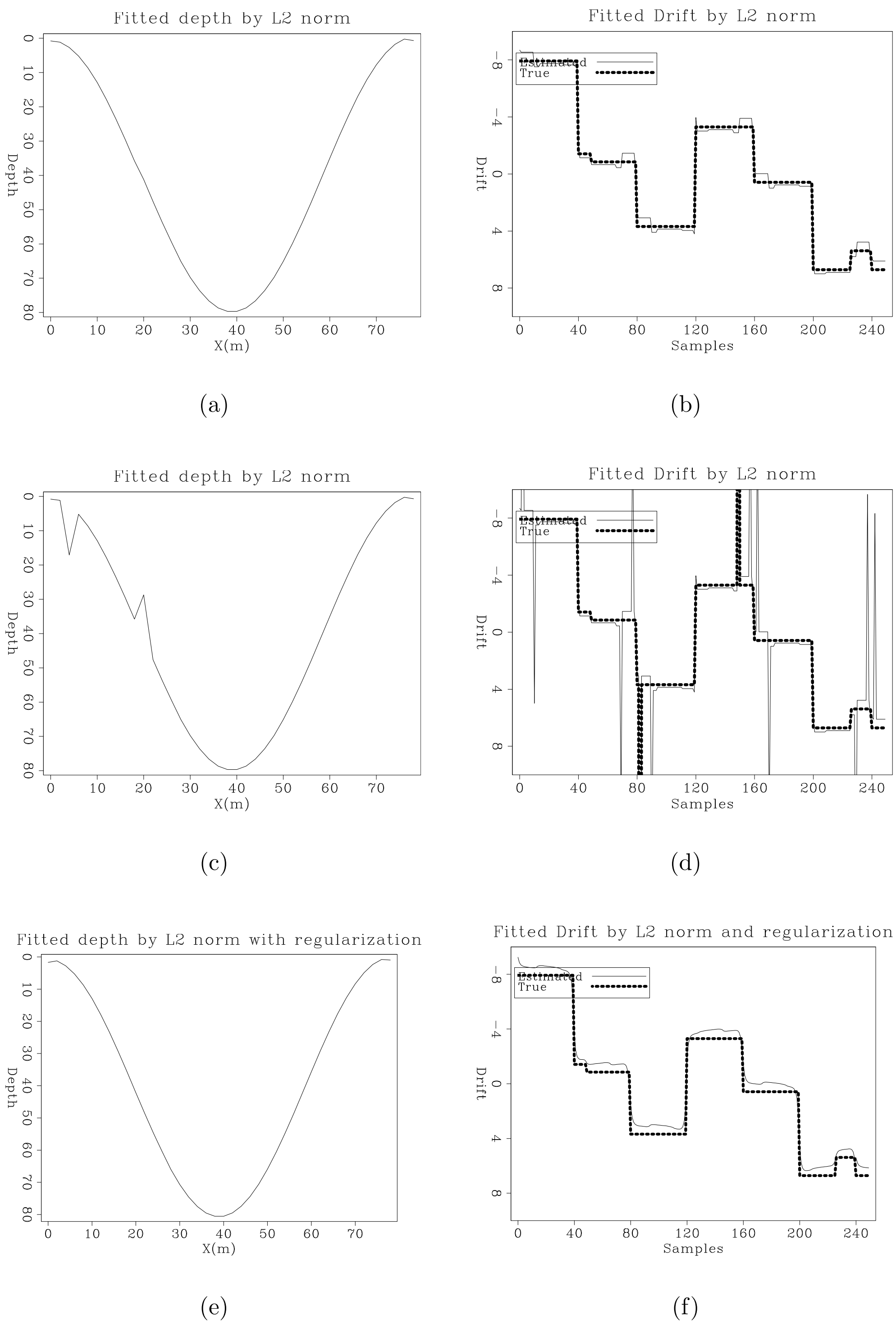
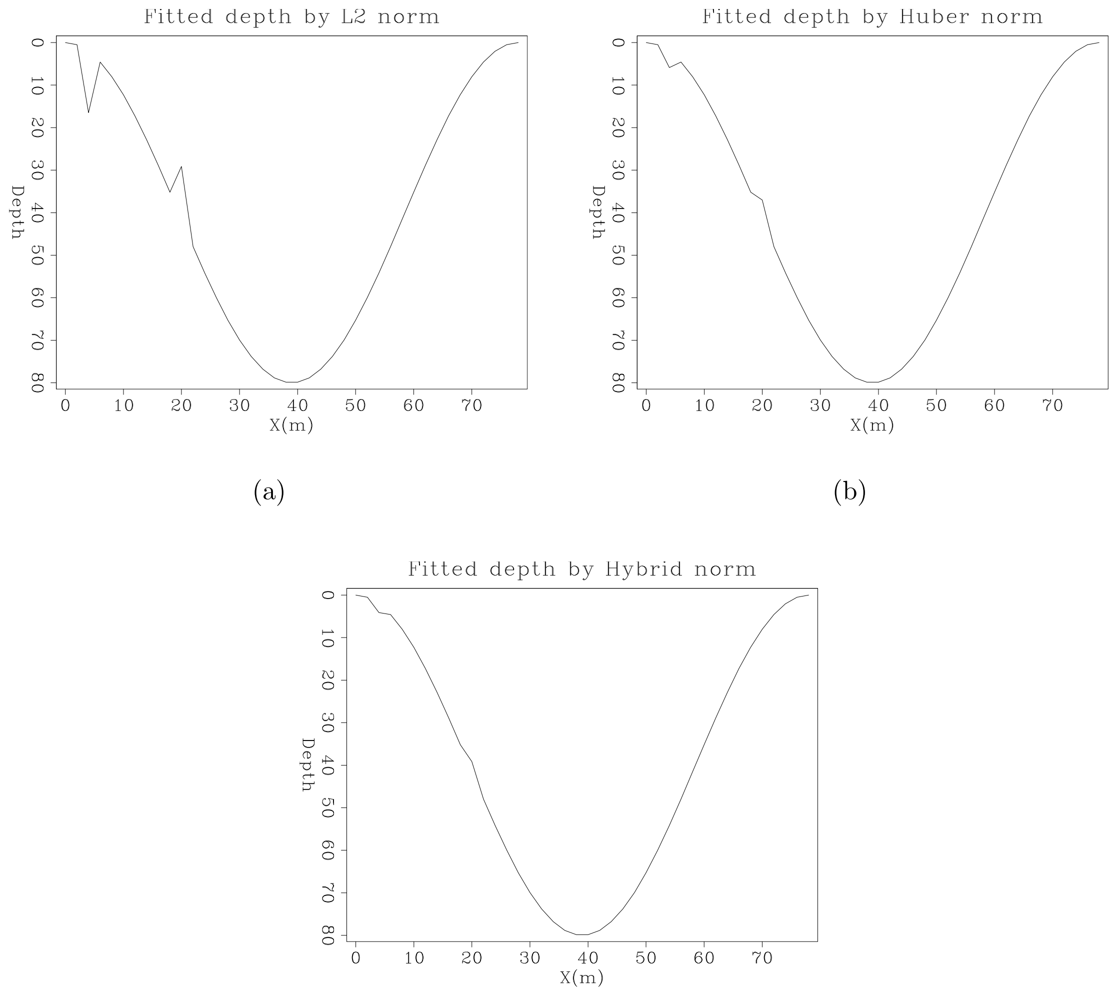


Figure 4: **(a,b)**: L2 inversion without regularizaton using the spike-free data. **(c,d)**: L2 inversion without regularization using the spiked data. **(e,f)**: L2 inversion with regularization using the spike-free data. [**ER**]



(a) (b)

(c)

Figure 5: Fitting depth for measurements with spiky noise with the generalized norm solver using **(a)** L2 norm. **(b)** Huber norm with $eps = 0.1$ and $percentile = 0.375$ **(c)** Hybrid norm with $eps = 0.1$ and $percentile = 0.275$. [**ER**]

erations, the Huber solver converges in 36 iterations, and the hybrid solver converges in 66 iterations. Repeated experience indicates that we can always set *psiter* to one when using the generalized norm solver.

Comments on the result

From observating the fitted drift between figure 6 (b) and 6 (d) , the Huber solver is not doing significantly better than the least-squares solver. One possible explanation is that the underlying Taylor series assumption failed while trying to solve for the stepping coeficient α and β. Recall that the formulas for the Huber norm are

$$
\begin{aligned}
C(r) &= \begin{cases} |r| - |r_t|/2 & |r/r_t| \geq 1 \\ r^2/2r_t & |r/r_t| < 1 \end{cases} \\
C'(r) &= \begin{cases} \mathrm{sgn}(r/r_t) & |r/r_t| \geq 1 \\ r/r_t & |r/r_t| < 1 \end{cases} \\
C''(r) &= \begin{cases} 0 & |r/r_t| \geq 1 \\ 1/r_t & |r/r_t| < 1 \end{cases}
\end{aligned}
\tag{5}
$$

Notice that the second derivative vanishes if the residual falls to the threshold value r_t. This could lead to failure of the Huber solver, because the second derivatives are used in the denonimator when solving for the step size α and β in the conjugate-direction scheme (Claerbout (2009)).

We are delighted to see that hybrid solver gives a resonable result for the full Galilee problem as shown in figure 5(c), we can hardly describe the model solution as "blocky." This might be because we have used a small threshold value for the model-fitting goal. For example, a threshold value of 0.30 percentile means we would like to see blocks about 3 to 4 points long. A higher threshold value for the model-fitting goal (equation 4) can increase blockiness; however our present solver has restricted us to use the same threshold value for both the model-fitting and the data-fitting goals (equation 3). One possible improvement for the future is to separate the thresholds for these goals.

CONCLUSION

The applications of the generalized norm solver show promising results in our two sample problems. The line-fitting problem shows that our solver can correctly remove spikes and noise added to the data. The 1-D Galilee problem shows that the solver can properly produce a blocky model space while removing outliers. In terms of convergence, the hybrid solver takes longer to converge than the Huber solver. While

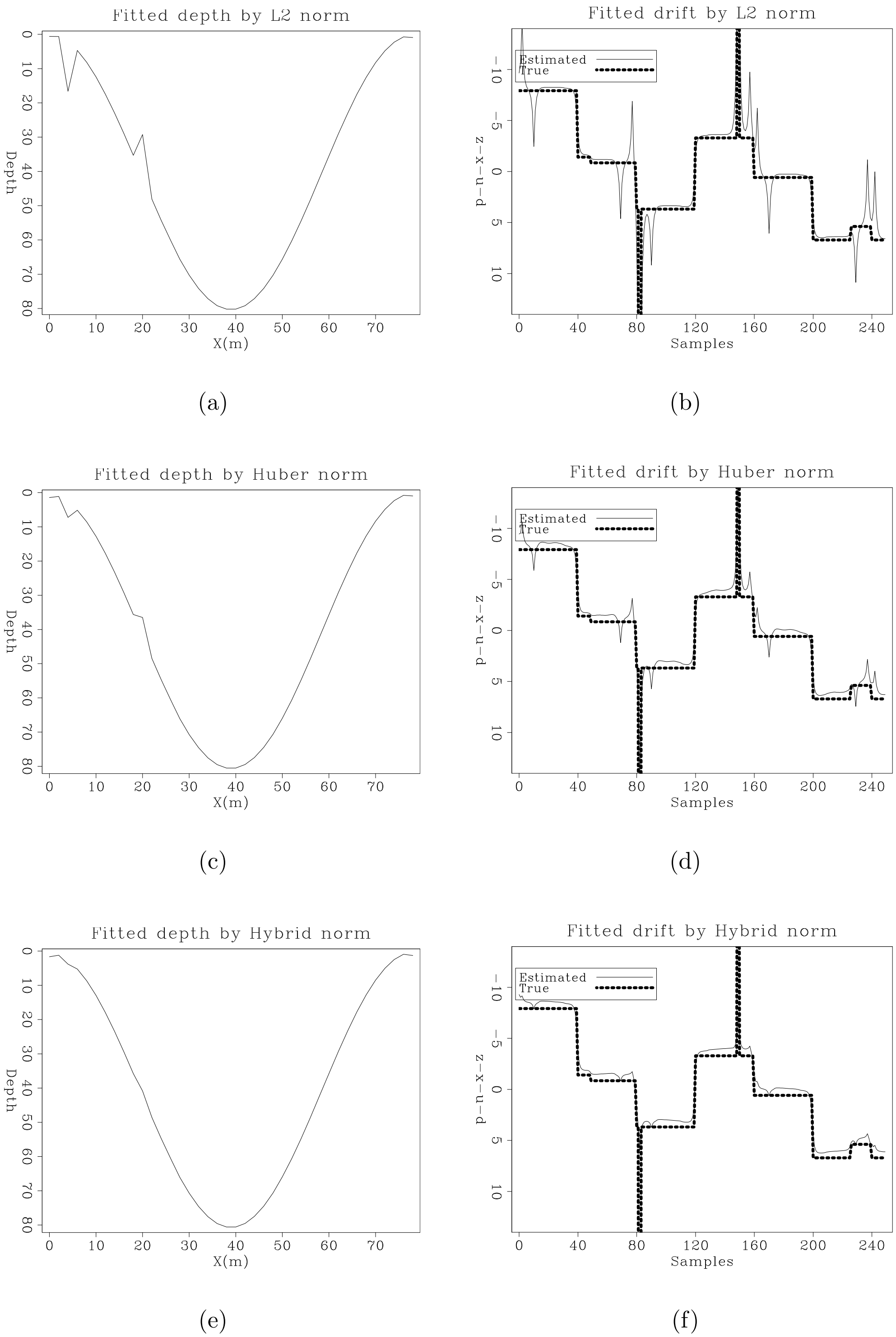


Figure 6: Fitting depth and drift for measurements with both drift and spiky noise with the generalized norm solver (regularized system) using the L2 norm **(a,b)**; the Huber norm with $eps = 0.1$ and $percentile = 0.07$ **(c,d)**; and the hybrid norm with $eps = 0.1$ and $percentile = 0.32$ **(e,f)**. [**ER**]

only the one-dimensional problem is examined with this solver, we plan to further explore this solver with 2-D field data problem and directly compare the result with the IRLS algorithm.

ACKNOWLEDGMENTS

We thank Bob Clapp for providing guidance, ideas, and technical support in this project.

REFERENCES

Bube, K. P. and R. T. Langan, 1997, Hybrid lambda1/lambda2 minimization with application to tomography: Geophysics, **62**, 1183–1195.

Claerbout, J., 2009, Blocky model via the l1/l2 hybrid norm: SEP report, **139**, 1–10.

Claerbout, J. F., ed., 2008, Image estimation by example: Geophysical sounding image construction.

Claerbout, J. F. and S. Fomel, eds., 2008, Basic earth imaging, 3.2 ed.

Gersztenkorn, A., J. B. Bednar, and L. R. Lines, 1986, Robust iterative inversion for the one-dimensional acoustic wave equation: Geophysics, **51**.

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

Maysami, M. and N. Moussa, 2009, Generalized-norm conjugate direction solver: SEP report, **139**, 11–22.