Least Squares Imaging using ℓ_1 and Student residuals.

Alan A. V. B. Souza and Jörg Schleicher, UNICAMP

This paper was prepared for presentation during the 15th International Congress of the Brazilian Geophysical Society held in Rio de Janeiro, Brazil, 31 July to 3 August, 2017. Contents of this paper were reviewed by the Technical Committee of the 15th International Congress of the Brazilian Geophysical Society and do not necessarily represent any position of the SBGf, its officers or members. Electronic reproduction or storage of any part of this paper for commercial purposes without the written consent of the Brazilian Geophysical Society is prohibited.

Abstract

This paper studies the Least Squares Migration (LSM) procedure as an optimization problem. Particularly, we study the behavior of this imaging procedure upon the use of robust loss functions. Our results demonstrate that LSM can work even under heavy noise if a suitable loss function is employed. Both the ℓ_1 norm and the Student's t-norm showed themselves more stable with regard to outliers than the ℓ_2 norm conventionally used in LSM. In the optimization step, we employ the hybrid deterministic-stochastic L-BFGS algorithm that exploits the structure of the objective function to reduce the computational burden of this imaging procedure. The quality of the achieved results show no deterioration over the conventional L-BFGS algorithm.

Introduction

Least squares migration (LSM) is a powerful technique used to improve the results of subsurface imaging, because it is capable of improving the resolution (Valenciano et al., 2009), balancing the amplitude (Farias et al., 2015), and in general to attenuate the acquisition footprint (Nemeth et al., 1999) of the imaged subsurface sections.

However, the ℓ_2 norm, usually applied in the objetive function to quantify the data residual, is strongly dependent on the underlying statistical assumptions (Li et al., 2013). Therefore, massive outliers caused by strong noise in the data can significantly reduce the quality of the resulting images. In this paper, we compare the behavior of LSM with two similar procedures using two other objective functions. We find that both the ℓ_1 norm and the Student's t-norm are more stable with regard to outliers than the ℓ_2 norm conventionally used in LSM.

Another drawback of LSM is its high computational demand. In this respect, we compare the performance of the more economic hybrid deterministic-stochastic L-BFGS algorithm of Friedlander and Schmidt (2012) to that of the conventional L-BFGS algorithm. In our tests, both procedures lead to migrated seismic sections of comparable quality.

In summary, we propose a combined LSM-like procedure using an improved loss function and a more economic algorithm. This procedure increases the method's tolerance to noise while at the same time reducing the computational expense of the procedure.

Least Squares Migration

The usual form of Least Squares Migration (LSM) is based on the Born approximation (Tarantola, 1984), which is used to derive a linear relationship between the medium parameters and the so called scattered wavefield. It is based on the introduction of a background medium, with the wavefields in the involved media satisfying

$$U_s\left(\mathbf{x}_s,\mathbf{x}_r,\mathbf{x},\omega\right) = U\left(\mathbf{x}_s,\mathbf{x}_r,\mathbf{x},\omega\right) - U_0\left(\mathbf{x}_s,\mathbf{x}_r,\mathbf{x},\omega\right), \tag{1}$$

with U being the total, U_0 the background and U_s the scattered wavefield, respectively. The decomposition of U into U_0 and U_s is usually assumed to follow from a decomposition of the physical parameters of interest as $m(\mathbf{x}) = m_0(\mathbf{x}) + \delta m(\mathbf{x})$, with m_0 being the smooth or background part of the parameter function (supposed to be known) and δm a perturbation (to be recovered).

This decomposition of the model parameters, when used jointly with the Born approximation, provides an approximate way to calculate the scattered wavefield for every source and receiver according to

$$\begin{aligned} U_s\left(\mathbf{x}_s,\mathbf{x}_r,\omega\right) &= -\,\omega^2 \int_\Omega s(\omega) G\left(\mathbf{x}_s,\mathbf{x}\right) \delta m\left(\mathbf{x}\right) G\left(\mathbf{x},\mathbf{x}_r\right) d\mathbf{x} \\ &= \int_\Omega K\left(\mathbf{x}_d,\mathbf{x}\right) \delta m\left(\mathbf{x}\right) d\mathbf{x}. \end{aligned} \tag{2}$$

In this equation, $s(\omega)$ is the source signature and the two Green's functions link the source to a generic scattering point $\mathbf{x}$ in the depth domain Ω and this point to the receiver. At the scattering point, the source wavefield interacts with the model perturbation function δm (here assumed to be the squared slowness, also called sloth). After this interaction the second Green's function connects the scattered field to the receiver. These Green's functions and the wavelet signature can be grouped under the *kernel* $K(\mathbf{x}_d,\mathbf{x})$, with $\mathbf{x}_d = (\mathbf{x}_s,\mathbf{x}_r,\omega)$ representing the data domain variables.

This grouping reveals the role of the *kernel* K as a mapping from the model space to the data space. A suitable discretization of both sides of equation (2) can be expressed in matrix form as $\mathbf{d} = \mathbf{L}\mathbf{m}$. Its approximate inverse $\mathbf{m} \approx \mathbf{L}^\mathsf{T}\mathbf{d}_s$ is a mapping from the data space to the model space. correspondingly in terms of the continuous equation (2), it represents an integral over the data domain variables (Schuster, 2010).

In general, the matrix $\mathbf{L}$ generated from the discretization of the *kernel* K is rectangular. Thus, instead of the above approximate inverse, a better estimate of the model $\mathbf{m}$ can be obtained by means of a least-squares solution, i.e.,

$$\mathbf{m} = \left[\mathbf{L}^\mathsf{T}\mathbf{L}\right]^\dagger \mathbf{L}^\mathsf{T}\mathbf{d}, \tag{3}$$

where the superscript † denotes the pseudo inverse matrix. Equation (3) can be solved using a great variety of linear-system algorithms. In the next section, we discuss the solution of this problem as one of optimization instead of one of linear least squares.

LSM as optimization problem

We study the relationship expressed in equation (1) under application to the reduced data set $\mathbf{d} = \tilde{\mathbf{d}} - \mathbf{d}_{0_i}$, where $\tilde{\mathbf{d}}$ represents the complete (measured) data and $\mathbf{d}_0$ is the synthetic data estimated in the smooth model $\mathbf{m}_0$. Then, the optimization goal becomes

$$J(\mathbf{m}) = \frac{1}{N_s}\sum_{i=0}^{N_s} R(\mathbf{r}_i) = \frac{1}{N_s}\sum_{i=0}^{N_s} R([\mathbf{L}_i\mathbf{m}] - \mathbf{d}_i) \tag{4}$$

In equation (4), $\mathbf{r}_i = [\mathbf{L}_i\mathbf{m}] - \mathbf{d}_i$ is the residual between the dataset $\mathbf{d}_i$ for source number i and the result from the corresponding Born modeling procedure represented by the action of $\mathbf{L}_i$ on the vector $\mathbf{m}$. Matrix $\mathbf{L}_i$ encapsulates the Born modeling (demigration) and migration procedure ($\mathbf{L}^\mathsf{T}$) for one experiment (shot) and all frequencies used. The summation in (4) is over the source index i. Finally, R is a scalar loss function, the explicit form of which will be discussed below. The most frequent choice, being the ℓ_2 norm, has given least-squares migration its name.

Equation (4) has a gradient of the form (O'Leary, 1990)

$$\nabla J = \frac{\partial J}{\partial \mathbf{m}} = \frac{1}{N_S}\sum_{i=0}^{N_s} \nabla_{\mathbf{m}} R(\mathbf{r}_i) = \frac{1}{N_S}\sum_{i=0}^{N_s} \mathbf{L}_i^\mathsf{T}\left[R'(\mathbf{r}_i)\right]. \tag{5}$$

In equation (5), the R' denotes the derivative of function R with respect to its argument, which is to be understood in a pointwise manner without summation. Knowing how to evaluate both function and gradient makes this optimization problem amenable to solution using a variety of methods as described, for instance, in Bonnans et al. (2006).

However, equation (5) has a special structure (sum of functions) that can be exploited to reduce the computational cost to solve problem (4). Using the hybrid deterministic-stochastic method proposed by Friedlander and Schmidt (2012), it is possible to reduce the number of evaluations of both the objective and gradient functions. To this end, we use Algorithm A.1.

In this algorithm, $\widehat{\mathbf{H}}$ is the L-BFGS approximation to the inverse Hessian, which is initialized using the Hessian diagonal as described in the third strategy of Farias et al. (2015). We use at most 10 pairs of models and gradient differences to build this approximation.

Specifics of Algorithm A.1

Algorithm A.1 is very similar to the traditional L-BFGS algorithm (Liu and Nocedal, 1989). The most important difference is the use of the growing-batch strategy. This strategy has two important consequences:

- **Reduced cost per iteration**: Usually the initial batch size ($\mathfrak{B}_0$) is very small when compared to the nominal size N_s, i.e., $\mathfrak{B}_0 \ll N_s$. This reduces the cost per iteration, enabling more iterations for a given computational budget.
- **Increasing batch sizes**: The cardinality (number of elements) of the batches is increased at every iteration. The reason is both the gradient and objective function estimates obtained in A.1 are only expected to be equal to the true values of these two quantities when the full objective function is used. The effect of the error from a smaller batch is reduced with increasing batch size. Here, we used an initial batch size of $\mathfrak{B}_0 = 5$, increasing by $incr = 1$ at each iteration.

The denomination of this algorithm as hybrid deterministic-stochastic (HDS) is justified by its behavior. At the beginning of the optimization, it behaves more in line with stochastic methods, like the SGD (Stochastic Gradient Descent) method using a mini-batch (Li et al., 2014). With increasing batch size, it tends to behaves more similarly to deterministic methods.

Specifying the function $R(\mathbf{r})$

The problem posed in equation 4 is now almost completely specified, except for the function R. This function controls how the misfit between calculated and observed data will be measured. Here, we understand R to represent a sum over frequency and receiver coordinates, i.e.,

$$R(\mathbf{r}) = \sum_{\omega,\mathbf{x}_r} \rho\left(r_{\omega,\mathbf{x}_r}\right), \tag{6}$$

where $r_{\omega,\mathbf{x}_r}$ represents a component of the data residual and ρ is a scalar loss function.

For the loss function ρ, usually the least-squares expression is used (see Table 1), because this function is computationally convenient to manipulate and also possesses a useful statistical interpretation. However, in practical problems the data residual does not necessarily present the statistical properties underlying the ℓ_2 function. Multiple factors can contribute to this. For the seismic inversion problem, two causes are notorious: The presence of high amplitude noise and systematic errors due to an incorrect physical model used in the inversion procedure (Li et al., 2013). When this happens, the ℓ_2 function loses its efficiency.

Algorithm A.1: Hybrid deterministic-stochastic L-BGFS algorithm, based on Friedlander and Schmidt (2012).

```
Input: $N_{iter}$, $\mathbf{m}_0$, $|\mathfrak{B}_0|$, incr
Result: $\mathbf{m}$
for $k \leftarrow 1$to $N_{iter}$ do
    Draw a batch with size $|\mathfrak{B}_k|$: $\mathfrak{B}_k \in \{1,\ldots,N\}$
    $\bar{J}_k = \frac{1}{|\mathfrak{B}_k|}\sum_{i\in\mathfrak{B}_k} R(\mathbf{r}_i)$
    $\nabla\bar{J}_k = \frac{1}{|\mathfrak{B}_k|}\sum_{i\in\mathfrak{B}_k} \nabla_{\mathbf{m}} R(\mathbf{r}_i)$
    $\mathbf{d}_k = -\widehat{\mathbf{H}}_k\nabla\bar{J}_k$
    $\mathbf{m}_{k+1}, \nabla\bar{J}_{k+1} = \texttt{LineSearch}\,(\nabla\hat{J}_k, \hat{J}_k, \mathbf{d}_k)$
    $\widehat{\mathbf{H}}_{k+1} = \texttt{L-BFGS\_Update}(\widehat{\mathbf{H}}_k, \mathbf{m}_{k+1}, \mathbf{m}_k, \nabla\bar{J}_{k+1}, \nabla\bar{J}_k)$
    $|\mathfrak{B}_{k+1}| = |\mathfrak{B}_k| + $incr
end
```

Table 1: Loss and influence functions. Student's t defined as van Leeuwen et al. (2013) and ℓ_1 as Brossier et al. (2010).

Name	$\rho(r)$	$\rho'(r)$
ℓ_2	$\frac{1}{2}r^2$	r
ℓ_1	$\lvert r\rvert$	$\frac{r}{\lvert r\rvert}$
student's t	$\log\left(1+\frac{r^2}{\sigma^2 k}\right)$	$\left(\frac{2}{k\sigma^2}\right)\frac{r}{1+\frac{r^2}{k\sigma^2}}$

One possible way to reduce these effects is to use different loss functions. For this reason, besides the ℓ_2 norm, we study in this work the behavior of the ℓ_1 norm and the Student's t-norm (Aravkin et al., 2012). All loss functions compared in this work are compiled in Table 1.

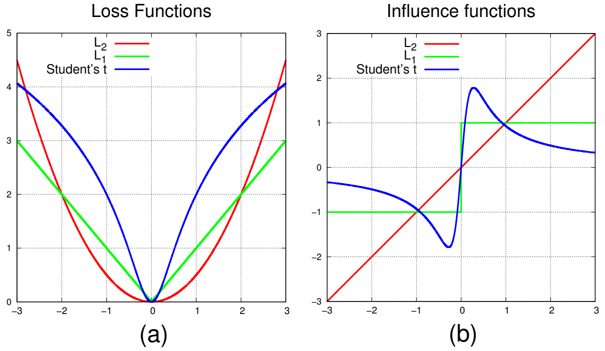


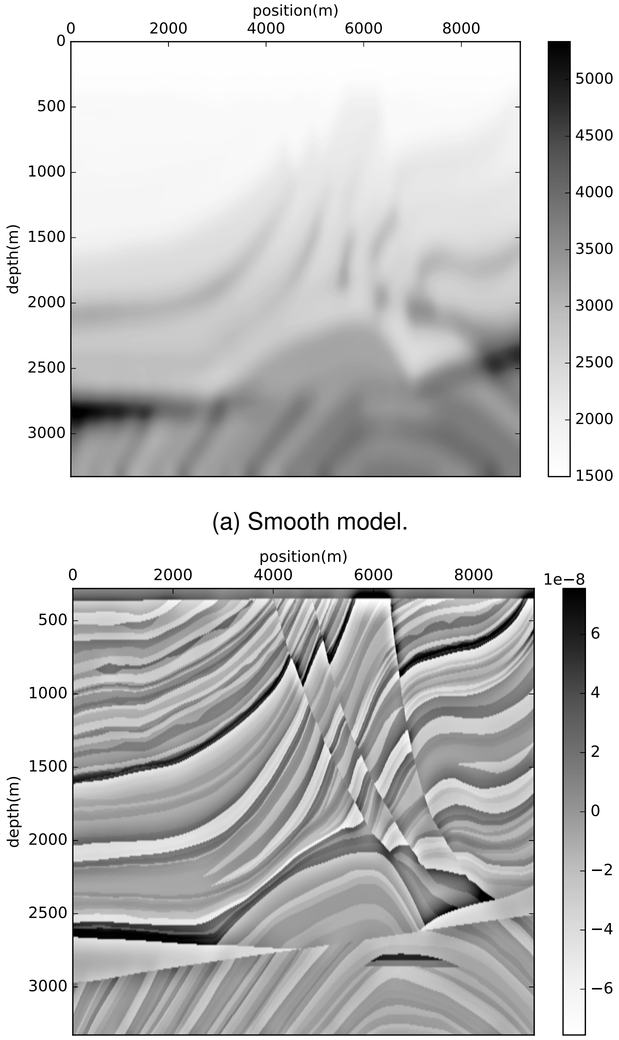
Figure 1: (a) Loss and (b) influence functions used.

The behavior of these functions can be understood and compared observing both their values and derivatives (see graphs in Figure 1). All functions have their global minimum in the origin, but they differ in their behavior in its neighborhood (Figure 1a). The ℓ_1 norm is the only function that is not differentiable there. Both the ℓ_1 and Student's t-norms assign a smaller weight to larger residuals than the ℓ_2 function, in this way reducing the importance of outliers.

The derivatives of these functions, also known as influence functions (Figure 1b), actually show that the ℓ_2 norm progressively attributes larger weights to larger residuals. In contrast, the ℓ_1 norm, except at the origin, attributes the same weights to residuals of all sizes, and the Student's t-norm even reduces the weights with increasing residual. This behavior explains the tolerance of these functions to high amplitude noise (outliers) when compared to the ℓ_2 function.

Numerical Experiments

We tested the above algorithm and loss functions on a modified (we added a thicker water layer) and decimated version of the Marmousi model (Brougois et al., 1990). The lateral and vertical grid spacing used is equal to 8 meters. The wavefield propagation was carried out in the frequency domain, using frequencies from 3 Hz up to 30 Hz with increments of 0.4 Hz. The data set consist of 171 shots (shot spacing of 60 meters) in a fixed-spread configuration, with both sources and receivers at the surface. The first/last shot and receiver are approximately at a distance of 300 and 200 meters, respectively, from the edges of the domain. Figures 2a and 2b show, respectively, the smooth background velocity model used as migration velocity and the exact sloth perturbation model obtained by subtracting the smoothed model (obtained by box filtering of 25×25 samples) from the unsmoothed one.



(a) Smooth model.

(b) Sloth (squared slowness) perturbation in s^2/m^2.

Figure 2: (a) Background model and (b) perturbation.

This data set was corrupted with a mix of Gaussian noise ($SNR = 6$ for all frequencies) and high amplitude noise. These outliers were simulated by adding high amplitude Gaussian noise randomly to 2% and 5% of all shots and receivers, respectively. Figure 3 depicts one frequency slice of the data used in this numerical experiment after the addition of noise.

The inversion procedure was realized using the HDS L-BFGS algorithm as schematized in Algorithm A.1. We tested all three loss functions detailed in Table 1. For comparability, we chose a fixed number of 50 iterations, which is roughly equivalent to 9 passes over the full data set for the batch-size parameters used. A pass corresponds to the evaluation of both the gradient and objective functions for all shots that make part of the data

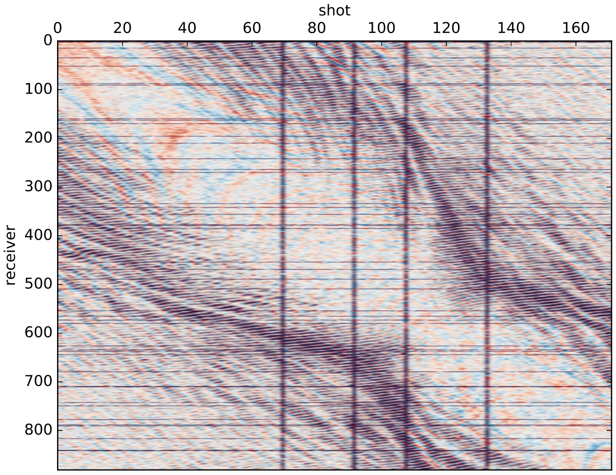


Figure 3: Frequency slice (at 14 Hz) of the corrupted data set used in the tested LSM procedure.

set.

For comparison, we also applied the LSM procedure using the conventional L-BFGS algorithm with the same number of iterations and using the Student's t loss function. The relative decrease of the objective function for both the HDS and conventional L-BFGS methods is plotted in Figure 4. This graph shows that after 50 iterations, both algorithms attain practically the same reduction of the objective function. However, the HDS L-BFGS algorithm attains this result using five times less passes over the data set. We see that for this particular problem, the HDS L-BFGS procedure with its computationally cheaper iterations is as useful as the conventional one with more expensive iterations.

In this example, the use of the HDS L-BFGS algorithm significanlty reduced the computational requirements of the LSM procedure. The number of function and gradient evaluations were reduced by a factor of five. Since the computing of both the function and gradient values is the most demanding part of the optimisation procedure, these reductions translated into a total runtime reduction of more than 80%.

Results

The results of the inversion after 50 iterations are shown in Figure 5. We recognize that the migrated image using the ℓ_2 norm (Figure 5a) is compromised due to the noise that created strong artifacts in various parts of the image. Even the relative amplitudes of the reflectors are affected. The results for both ℓ_1 and Student's t-norm are cleaner and shown better subsurface images. The ℓ_1-norm result is somewhat cleaner ihn terms of artifacts than the Student's t-norm results. The images 5c and 5d are interesting to compare, since the former was calculated using the HD L-BFGS method while the latter used the conventional L-BFGS method. Both images have almost the same quality level and differences are very hard to spot. On the one hand, the artifacts in image 5c seem to be a little weaker than in the conventional result of Figure 5d. On the other hand, the latter image shows a small amplitude boost throughout the section in comparison to Figure 5c.

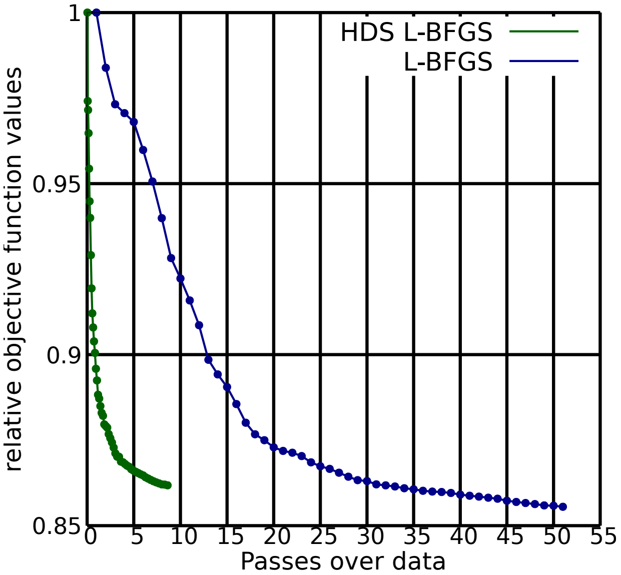


Figure 4: Relative objective function progression for the Student's loss function during the optimization procedure for both Hybrid Deterministic-Stochastic L-BFGS (HDS L-BFGS) and conventional L-BFGS. The relative objective functions values shown here were calculated for both algorithms using a full evaluation of the objective function.

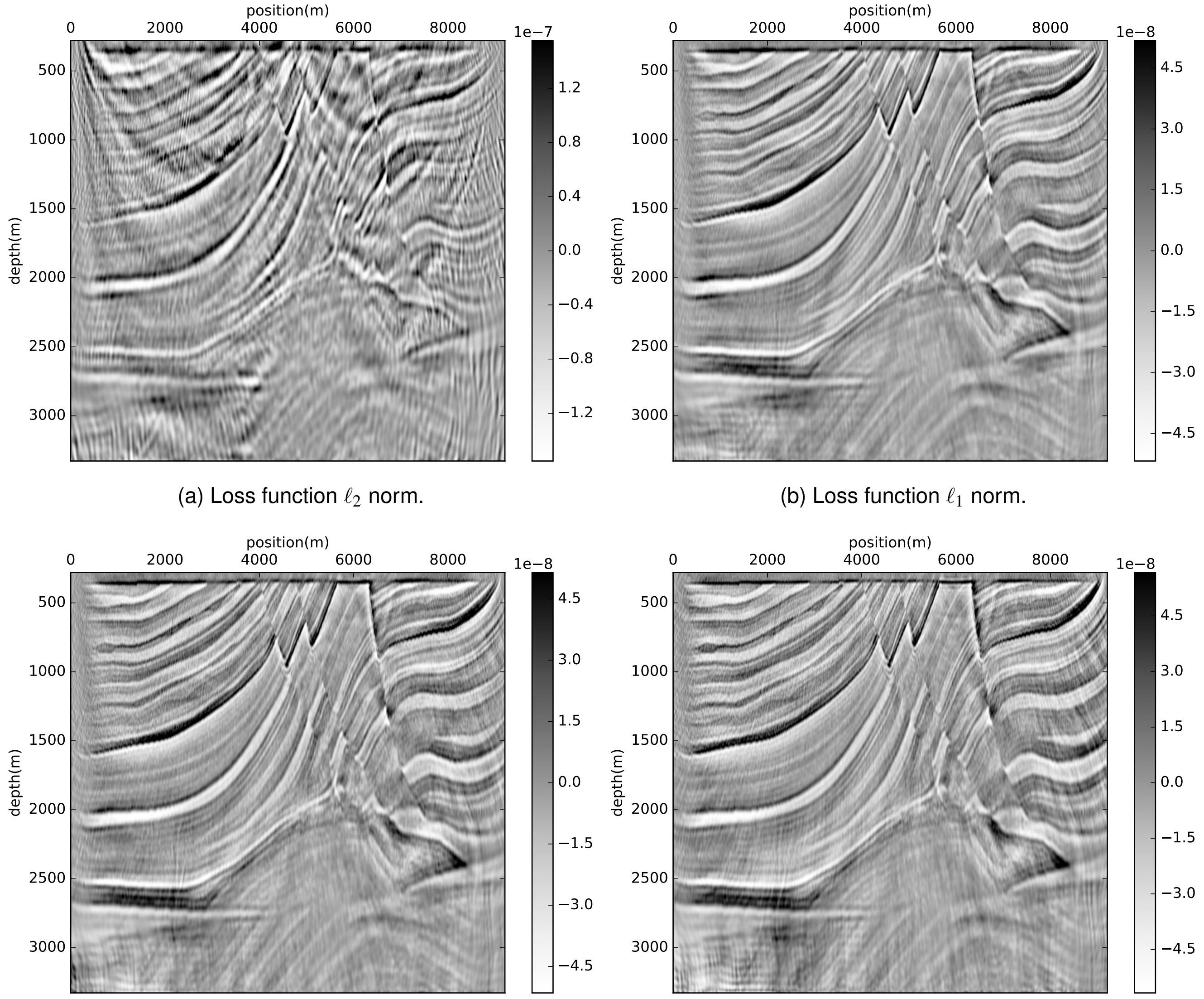
Finally, to allow for a more detailed analysis, the graph in Figure 6 shows the resulting depth profile at the horizontal position $x = 6608$ m as obtained by the four inversions. This region is one of the most complex parts of this model. The graph shows a good agreement of the inversion results using the more robust loss functions with the true model (black line). As expected, the ℓ_2-norm result (red line) fails to adequately recover the sloth perturbations. The results usint the Student's t-norm with both L-BFGS algorithms are very similar to each other. It is to be noted that at some places of the section, particularly in the bottom part, all algorithms had difficulties in fitting the higher contrasts present in the model.

Discussion

There are some imaging problems in the bottom left corner of all four sections in Figure 5. These problems can be attributed to the use of the Hessian diagonal as preconditioner. The reason is this region presents a high velocity contrast, even in the migration velocity model after the smoothing procedure, and it is also positioned at the far left corner of the model where data coverage is poor. These conditions generate a high amplitude on the inverse diagonal used during the application of the L-BFGS matrix. A possible solution to this problem includes adding a small term to regularize the division or using this preconditioner for a limited number of iterations.

Conclusion

In this work, we have investigated the ability of the LSM procedure to recover the perturbation model even under heavy noise. Our results demonstrate that LSM can work under these conditions if a suitable loss function is



(a) Loss function ℓ_2 norm.

(b) Loss function ℓ_1 norm.

(c) Loss function Student's t-norm.

(d) Loss function Student's t-norm, conventional L-BFGS.

Figure 5: Inversion results using the HDS L-BFGS algorithm for different loss functions (5a, 5b, 5c) and conventional L-BFGS algorithm (5d).

employed. Both the ℓ_1 norm and the Student's t-norm have shown themselves more stable with regard to outliers than the ℓ_2 norm conventionally used in LSM.

In the algorithmic part of the present investigation, we have seen that the HDS L-BFGS algorithm is very interesting in this application since it can reduce the computational requirements considerably while, at the same time, providing the same quality level as the more expensive conventional L-BFGS algorithm. In our example, the overall runtime economy amounted to more than 80%. It is important to remark that the convergence for this method is guaranteed only for smooth functions. Despite of that, the ℓ_1 optimization result with the HDS L-BFGS algorithm is adequate. This behavior has been previously observed in other settings, as shown by Brossier et al. (2010).

Acknowledgments

We would like to thank PETROBRAS for permission to publish this work.

References

Aravkin, A., M. P. Friedlander, F. Herrmann, and T. van Leeuwen, 2012, Robust inversion, dimensionality reduction, and randomized sampling: Mathematical Programming, **134**, 101–125.

Bonnans, J., J. Gilbert, C. Lemaréchal, and C. Sagastizábal, 2006, Numerical optimization – theoretical and practical aspects: Springer Verlag, Berlin. Universitext.

Brossier, R., S. Operto, and J. Virieux, 2010, Which data residual norm for robust elastic frequency-domain full waveform inversion?: Geophysics, **75**, R37–R46.

Brougois, A., M. Bourget, P. Lailly, M. Poulet, P. Ricarte,

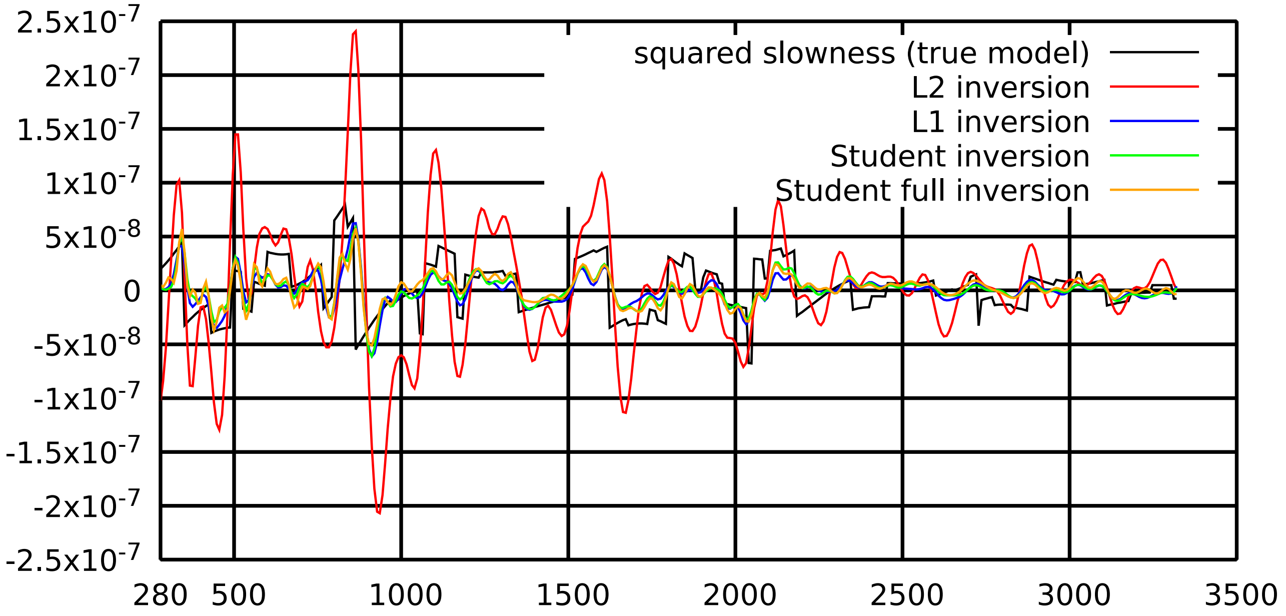


Figure 6: Vertical profiles at $x = 6608$ m of the inversion results using the HDS L-BFGS algorithm for different loss functions.

and R. Versteeg, 1990, *in* Marmousi, model and data: 5–16.

Farias, F. F., A. A. V. B. Souza, A. Bulcão, B. P. Dias, D. M. S. Filho, L. A. Santos, and W. J. Mansur, 2015, *in* Reverse time migration with amplitude weighted by seismic illumination: 1106–1108.

Friedlander, M. P., and M. Schmidt, 2012, Hybrid deterministic-stochastic methods for data fitting: SIAM Journal on Scientific Computing, **34**, A1380–A1405.

Li, M., T. Zhang, Y. Chen, and A. J. Smola, 2014, Efficient mini-batch training for stochastic optimization: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 661–670.

Li, X., A. Tamalet, T. van Leeuwen, and F. J. Herrmann, 2013, *in* Optimization driven model-space versus data-space approaches to invert elastic data with the acoustic wave equation: 986–990.

Liu, D., and J. Nocedal, 1989, On the limited memory bfgs method for large scale optimization: Mathematical Programming, **45**, 503–528.

Nemeth, T., C. Wu, and G. T. Schuster, 1999, Least-squares migration of incomplete reflection data: GEOPHYSICS, **64**, 208–221.

O'Leary, D. P., 1990, Robust regression computation using iteratively reweighted least squares: SIAM Journal on Matrix Analysis and Applications, **11**, 466–480.

Schuster, G. T., 2010, Seismic Interferometry: Cambridge University Press.

Tarantola, A., 1984, Linearized inversion of seismic reflection data: Geophysical Prospecting, **32**, 998–1015.

Valenciano, A. A., B. L. Biondi, and R. G. Clapp, 2009, Imaging by target-oriented wave-equation inversion: GEOPHYSICS, **74**, WCA109–WCA120.

van Leeuwen, T., A. Y. Aravkin, H. Calandra, and F. J. Herrmann, 2013, In which domain should we measure the misfit for robust full waveform inversion?: Presented at the EAGE Annual Conference Proceedings.