Fast 3D Least-Squares RTM by Preconditioning with Non-Stationary Matching Filters
Antoine Guitton, Colorado School of Mines, Golden, Colorado, USA

SUMMARY
I approximate the inverse Hessian \((L'L)^{-1}\), where \(L\) is a Born modeling operator and \(L'\) is the corresponding adjoint RTM operator, with non-stationary matching filters. These non-stationary filters can be seen as a low-rank approximation of the true inverse Hessian. For 3D least-squares imaging, I use these matching filters to precondition the inversion of prestack seismic data and observe a significant convergence speed-up.

INTRODUCTION
Assuming that the background velocity model is accurate enough, least-squares imaging improves the resolution of seismic images (deconvolution effect), the illumination of deep reflectors, and the amplitude fidelity (Kuhl and Sacchi, 2003; Clapp, 2005). In addition, migration artifacts due to the acquisition geometry are reduced (Nemeth et al., 1999). With two-way operators, back-scattering noise is also attenuated. However, these important features come at an increased cost, especially in 3D, where many iterations involving computer-intensive migration and demigration steps are needed to best fit the recorded prestack data.

Because least-squares imaging is expensive, many cost-saving strategies are possible if we recognize that the benefits of least-squares inversion comes from the effects of the inverse Hessian operator on the migrated images. Acknowledging this fact, we can either try to approximate the inverse Hessian without running any inversion (Rickett, 2003; Guitton, 2004), speed-up the inversion by designing preconditioning operators (Aoki and Schuster, 2009; Hou and Symes, 2015, 2016), decrease the size of the problem using a target-oriented solution (Tang, 2009), or design pseudo-unitary migration operators (Zhang et al., 2014).

My goal in this paper is to speed-up the inversion by designing a preconditioning operator based on the computation of matching filters. These filters operate in the model space, are non-stationary (vary in x,y and z directions), and approximate the effects of the inverse Hessian. Estimating these filters is straightforward and is explained in details in Guitton (2004). The novelty here is that I am using these filters to precondition the inversion for 3D least-squares reverse-time migration (LSRTM). On a synthetic 3D data example, I am showing a significant speed-up where three to four iterations of the preconditioned inversion yields significantly better results than fifteen iterations of a non-preconditioned one. On a 3D field data example, the cost saving is about a factor two: with 3D data, these are important cost reduction numbers. Compared to other preconditioning techniques such as those using de-blurring filters (Aoki and Schuster, 2009), or approximate inverses (Hou and Symes, 2015), matching filters are simple to implement, robust and easily generalizable to more complicated imaging operators (e.g., elastic, anisotropic, etc...).

THEORY
This section describes the estimation of the non-stationary filters first, and then how these filters can be used to precondition LSRTM. In this paper, a migrated image corresponds to an acoustic velocity perturbation \(m\) according to the following decomposition of the velocity model \(m\):\[
m_1 = m_b + m \quad (1)
\]
where \(m_b\) is the background velocity (i.e., migration velocity). Throughout this paper, I call \(m\) a reflectivity model as well.

Filter estimation
In a first step, we need to estimate the non-stationary matching filters. Let’s define \(L\) a Born modeling operator using a two-way scalar wave equation (velocity only) and \(L'\) its adjoint (i.e., reverse time migration operator). First, I estimate a migrated image \(m_1\) with RTM using the input data \(d\):\[
m_1 = L'd \quad (2)
\]
From \(m_1\), I compute a second image \(m_2\) going through a full loop of demigration/migration steps as follows:

\[
m_2 = L'Lm_1 \quad (3)
\]
Therefore, \(m_2\) is the result of the migration of the remodeled data. Having the two migration results \(m_1\) and \(m_2\), I estimate a bank of non-stationary filters \(a\) minimizing

\[
f(a) = \|M_2a - m_1\|^2_2 \quad (4)
\]
where \(M_2\) is the operator representation of the non-stationary convolution with the remigrated image \(m_2\). The least-squares solution \(\hat{a}\) is given by the normal equations

\[
\hat{a} = (M_2^*M_2)^{-1}M_2^*m_1 \quad (5)
\]
showing that the filters are obtained by the correlation of the two migrated images divided by the autocorrelation of \(m_2\). In practice, a conjugate-direction solver (Claerbout and Fomel, 2014) is used to estimate the filter coefficients. Now we have

\[
(M_2^*M_2)^{-1}M_2^*M_1 \approx (L'L)^{-1} \quad (6)
\]
with \(M_1\) the operator representation of the non-stationary convolution with the migrated image \(m_1\). These filters...
can be seen as low rank approximations of the Hessian. Notice also that \( m_1 \) is “encoded” within equation 6 and affects the accuracy of this method. I illustrated in Guitton (2004) how these filters can efficiently approximate the inverse Hessian for a 2D, one-way source-receiver migration operator.

Now that I have estimated the non-stationary filters \( \hat{a} \), I can use them to precondition the least-squares inversion: using a conjugate-direction solver, we end up with an algorithm similar to the weighted CG algorithm of Hou and Symes (2016) where the non-stationary convolution is used in the definition of the model-space inner product.

**A 3D SYNTHETIC DATA EXAMPLE**

For this 3D synthetic example, I use a subset of the 3D SEAM model for the reflectivity and velocity models and opt for an OBC-style acquisition geometry. The shot/receiver geometry is similar to the one used in the 3D field OBC example that I will show in the next section. In this experiment, I have twelve parallel receiver lines every 400 m. at a depth of around 90 m. I model and migrate 288 3D receiver gathers, with around 4200 traces (shot positions) per gather. I generate Born-modeled data using the velocity and perturbation models of Figures 1a and 1b, respectively. There is a salt-body on the right side of the model. The maximum frequency of the data is 20 Hz. Therefore the goal of LSRTM is to recover an image as close as possible to Figure 1b. Now, I describe the different stages of the approximate inverse Hessian computation.

**Computing \( m_1 \) and \( m_2 \)**

The first step consists in computing a migrated image \( m_1 \) shown in Figure 1c and a re-migrated image \( m_2 = L^t d \) shown in Figure 1d. Because the Born-modeling operator \( L \) is not unitary, the re-migrated image \( m_2 \) is significantly different from \( m_1 \) with much lower amplitudes in the deepest parts of the model. Also, note that the RTM images were processed to remove the back-scattering noise resulting from the sharp velocity contrast where salt is present. This processing step is necessary to focus the filter estimation on the reflectors only.

**Computing the 3D matching filters**

The second step consists in estimating 3D matching filters to minimize the objective function in equation (4). I estimate 3D filters of dimensions 15x15x15 and keep them constant within a patch of 5x5x5 samples. The number of patches is 40 in the \( z \) direction, 100 in \( x \), and 50 in \( y \). Having the filters, it is easy to compute an improved RTM image that will look closer to the true reflectivity function in Figure 1b. I show in Figure 2a such a result. Comparing with Figure 1c, we see the amplitude-balancing effect of the filters. Comparing with Figure 1b, i.e. the true reflectivity model, we notice that the filters indeed help recovering it very well.

Figure 1: (a) velocity and (b) reflectivity models used to generate the Born data. In (b), clipped values are showed in red. (c) \( m_1 = L^t d \) and (d) \( m_2 = L^t L m_1 \) needed to estimate the filters. Note the amplitude decrease in the deepest parts of \( m_2 \) compared with \( m_1 \).
3D LSRTM Inversion

Now, I am using these filters to improve the convergence of LSRTM. But first, I run fifteen iterations of LSRTM without any preconditioning. The final image, using the same clip values as in Figure 1b, is shown in Figure 2b. For the inversion, I scale the gradient at each iteration by the receiver illumination map. In addition, backscattering noise is attenuated applying a mild high-pass filter. The amplitude fidelity is improved after inversion. Comparing with the filtering result of Figure 2a, I notice that the deepest parts of the model \((z \geq 2\, \text{km})\) are still lacking strength: more iterations and/or a better illumination-compensation scheme (e.g., incorporating the source illumination as well) would help. However, LSRTM is able to extend the image on the edges of the model better than the simple filtering approach. Then, I use the filters estimated in the previous section to precondition the inversion. I show the inversion results after 5 iterations in Figure 2c. The estimated reflectivity looks very close to the true model in Figure 1b.

A 3D OBC FIELD DATA EXAMPLE

Now, I apply 3D LSRTM with and without preconditioning to a field dataset from the North Sea shot over the Volve field. The acquisition geometry is similar to the one used in the synthetic example. The maximum frequency of the data is again 20 Hz. The velocity model in Figure 3a shows the top and base chalk layer very well between \(z=2\, \text{km}\) and \(z=3\, \text{km}\).

3D OBC FIELD DATA EXAMPLE

Now, I apply 3D LSRTM with and without preconditioning to a field dataset from the North Sea shot over the Volve field. The acquisition geometry is similar to the one used in the synthetic example. The maximum frequency of the data is again 20 Hz. The velocity model in Figure 3a shows the top and base chalk layer very well between \(z=2\, \text{km}\) and \(z=3\, \text{km}\).

First, I compute the RTM image shown in Figure 3b: the top panel shows an interesting depth-slice at 1.225 km where channel structures are present. We notice weak reflections in the deepest parts of the model below the chalk and strong migration artifacts on the edges.
of the image (as seen in the corner of the depth slice). Then, I run 15 iterations of LSRTM and display the final image in Figure 3c. The first thing to notice are the stronger reflectors below the chalk layer. Then we also see the more balanced amplitudes in the depth slice and the attenuation of the migration artifacts on the edges. Although not easily noticeable, the resolution of the reflectors above chalk has increased a lot: a strong black reflector above the chalk layer at $z \approx 2$ km appears.

Figure 4: A comparison between (a) the LSRTM image with preconditioning after two iterations, and (b) the LSRTM image without preconditioning after five iterations using (a) as the starting model.

Next, after estimating 3D matching filters from the migration result of Figure 3b and the re-migrated image, I run only two iterations of preconditioned LSRTM. Note that due to weak reflections in the deepest parts of the model, the size of each filter is now 21x21x21. It turns out that after only two iterations, the residual stops decreasing, not improving the image further. The result is shown in Figure 4a. We notice most of the beneficial aspects of LSRTM with only two iterations (i.e. decreasing artifacts, improved illumination). However, looking in more details, we don’t see a significant improvement in resolution.

Finally, to get all the benefits of LSRTM, I use the result of Figure 4a as the starting model for five more iterations without preconditioning of LSRTM. I show the final model in Figure 4b. Now, this last result is much closer to Figure 3c for half the computing cost. The arrow labeled “1” in Figure 4b shows a reflector that was not visible with preconditioning only in Figure 4a. Arrow “2” also shows in a depth-slice the reduction in artifacts and improved illumination.

DISCUSSION - CONCLUSION

The methodology works very well: non-stationary matching filters provide a cheap and efficient way to approximate the inverse Hessian in 3D. The convergence of LSRTM is improved dramatically with a synthetic test case, and substantially with a field data example. Switching from preconditioned to un-preconditioned LSRTM seems to provide all the benefits of LSRTM for about half the cost.

Many interesting avenues of research are left untouched. First, effects of the wrong velocity on the filter estimation process and preconditioned LSRTM need to be investigated. Second, understanding the influence of $m_1$ on the accuracy of the approximation should be investigated: would spikes for $m_1$ perform better than migrated images? Second, it would be very beneficial to use this technique with extended images where the cost of inversion is quite higher than with a zero subsurface-offset LSRTM (Hou and Symes, 2016).

There are implementation details that need to be worked on as well. The filter estimation process can be expensive. For shallow target, I observe that filters can be quite small. For deeper reflectors and not-so-well illuminated targets (like below the chalk layer in the North Sea dataset), bigger and more expensive filters are needed. A filter which size would change with position would help mitigating the computing costs. One missing feature of the preconditioned LSRTM with matching filters is the increase of resolution. This should come at no surprise to us: because I use a convolution in equation (4) to match a lower resolution image $m_2$ to a higher resolution image $m_1$, we can’t recover the high wavenumbers. I offer three solutions. First, I can stop using the filters, as done with the Volve dataset and iterate a few times to recover the high wavenumbers. Second, I can use different filter sizes for each LSRTM iteration. Finally, I can add a regularization operator to the LSRTM objective function that will boost the high wavenumbers.

These issues set aside, one of the main benefits of the proposed approach is that it works for any imaging operator. Including anisotropic or elastic effects is trivial because the filter estimation process remains the same: all the physical aspects of wave propagation are encapsulated in the two images we are trying to match.

ACKNOWLEDGMENTS

I would like to thank the sponsors of the Center for Wave Phenomena for their financial support. I would also like to thank Marianne Houbiers at Statoil, Statoil ASA and the Volve license partners ExxonMobil E&P Norway AS and Bayerngas Norge AS, for the release of the Volve data. The views expressed in this paper are the views of the author and do not necessarily reflect the views of Statoil ASA and the Volve field license partners.
REFERENCES


