Stanford Exploration Project, Report 125, January 16, 2007, pages 105–121

Least-squares migration of incomplete data sets with regularization in the subsurface-offset domain

Yaxun Tang

ABSTRACT

I present a method to address the migration artifacts caused by insufficient offset coverage. I pose the migration as a least-squares inversion problem regularized with the differential semblance operator, followed by forcing a sparseness constraint in the subsurface-offset domain. I demonstrate that adding these regularization terms suppresses the amplitude smearing in the subsurface-offset domain and improves the resolution of the migrated image. I test my methodology both on a synthetic two-layer data set and the Marmousi data set.

INTRODUCTION

With correct migration velocity and infinite survey length, wave-equation migration will focus the energy perfectly at the zero-offset locations. In reality, however, since the survey length can never be infinite, the offset domain suffers from the truncation effect. The situation gets even worse when the data sets are incomplete or irregular; severe amplitude smearing and aliasing artifacts may appear in the offset gathers and migrated images. This problem is more pronounced in 3-D, because of the irregular nature of 3-D seismic data.

One way to deal with this problem is to interpolate before migration, as is done in Radon-based interpolation schemes. Though Radon-based methods are acknowledged to be effective for data interpolation, they have severe theoretical limitations that require the events in CMP gathers to have the shape of hyperbolas or parabolas. These limitations prevent them from accurately interpolating in situations with very complex velocity structures.

Another approach is to pose the migration problem as a regularized inversion process. A reasonable regularization term, assuming lateral continuity along the reflection-angle axis, would be to smooth across offset-ray parameters in the Angle-Domain Common Image Gathers (ADCIGs). As shown in Prucha et al. (2000) and Kuehl and Sacchi (2001), by smoothing along the angle gathers, the illumination gaps can be successfully filled in and the migration artifacts caused by the insufficient survey length and lack of illumination can be attenuated to some extent.

In this paper, I describe another method based on least-squares migration with regularization in the Subsurface-Offset-Domain Common Image Gathers (SODCIGs). Regularizing in the SODCIGs instead of in the ADCIGs has the advantage of being computationally cheaper,

since it saves the computational cost of transforming from SODCIGs into ADCIGs. The basic idea is to add a Differential Semblence Operator (DSO) in the SODCIGs along the offset dimension to penalize energy far from zero-offset locations (Shen et al., 2003; Valenciano, 2006), followed by applying a sparseness constraint which minimizes the model residuals in the L_1 norm or Cauchy norm to enhance the resolution of the image cube.

I start with reviewing the theory of Bayes inversion and derive the weighting function for the sparseness constraint, then derive the objective function with DSO regularization and sparseness constraints based on the propagation of wavefields. I demonstrate that regularizing with DSO in the SODCIGs is equivalent to regularizing with a roughener along the offset-ray axes in the ADCIGs. To further reduce the computational cost, I approximate the Hessian with a diagonal matrix, which eliminates the need to propagate wavefields upward and downward within each iteration; however, the trade-off is a loss of accuracy. My approximated inversion scheme is tested on a simple two-layer model as well as the complex Marmousi model.

BAYES INVERSION

If we let $\mathbf{m}$ be the model vector and $\mathbf{d}$ be the data vector, Bayes's theorem would state

$$p(\mathbf{m},\mathbf{d}) = \frac{p(\mathbf{m})p(\mathbf{d},\mathbf{m})}{p(\mathbf{d})}, \tag{1}$$

where $p(\mathbf{m},\mathbf{d})$ is the distribution of the model parameters posterior to the data $\mathbf{d}$, expressing the likelihood of the model $\mathbf{m}$ for a given data $\mathbf{d}$; $p(\mathbf{m})$ is the probability distribution of the model $\mathbf{m}$, representing the prior information about the model; and $p(\mathbf{d},\mathbf{m})$ describes how knowledge of the data modifies the prior knowledge. The quantity $p(\mathbf{d})$ is the probability distribution of the data $\mathbf{d}$, which is a constant for a given data $\mathbf{d}$; thus $p(\mathbf{d})$ can be seen as a scaling factor (Ulrych et al., 2001). Therefore, equation (1) can be simplified as

$$p(\mathbf{m},\mathbf{d}) \propto p(\mathbf{m})p(\mathbf{d},\mathbf{m}). \tag{2}$$

In the presence of noise, the recorded data $\mathbf{d}$ can be expressed as follows:

$$\mathbf{d} = \mathbf{L}\mathbf{m} + \mathbf{n}, \tag{3}$$

where $\mathbf{n}$ is the noise vector. The likelihood function $p(\mathbf{d},\mathbf{m})$ is constructed by taking the difference between the observed data and the modeled data, thus

$$p(\mathbf{d},\mathbf{m}) = p(\mathbf{n}). \tag{4}$$

If we assume the noise has a Gaussian distribution with a zero mean and variance σ^2, its probability function can be written as follows:

$$p(n_i) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{n_i^2}{2\sigma^2}} \tag{5}$$

If each component of the noise vector is independent and the variance of each noise sample remains constant, the total probability of the noise vector is

$$\begin{aligned} p(\mathbf{n}) &= p(n_1)p(n_2)\cdots p(n_N) && (6)\\ &= \frac{1}{(2\pi\sigma^2)^{N/2}} e^{\frac{-1}{2\sigma^2}\sum_{i=1}^{N} n_i^2} && (7)\\ &= \alpha_1 e^{\frac{-1}{2\sigma^2}\mathbf{n}'\mathbf{n}}, && (8) \end{aligned}$$

where $\mathbf{n}'$ is the adjoint of $\mathbf{n}$. If we further assume that the model parameters $m_i, i = 1,2,\ldots,M$ have a Gaussian distribution with a zero mean and a variance σ_m^2 and are independent, the probability of $\mathbf{m}$ then is

$$\begin{aligned} p(\mathbf{m}) &= \frac{1}{(2\pi\sigma_m^2)^{M/2}} e^{\frac{-1}{2\sigma_m^2}\mathbf{m}'\mathbf{m}} && (9)\\ &= \alpha_2 e^{\frac{-1}{2\sigma_m^2}\mathbf{m}'\mathbf{m}} && (10) \end{aligned}$$

Plugging equations (8) and (10) into equation (2) yields the following posterior distribution:

$$\begin{aligned} p(\mathbf{m},\mathbf{d}) &\propto \alpha_1\alpha_2 e^{-\frac{1}{2\sigma_m^2}\mathbf{m}'\mathbf{m}-\frac{1}{2\sigma^2}\mathbf{n}'\mathbf{n}} && (11)\\ &\propto \alpha_1\alpha_2 e^{-\frac{1}{2\sigma_m^2}\mathbf{m}'\mathbf{m}-\frac{1}{2\sigma^2}(\mathbf{L}\mathbf{m}-\mathbf{d})'(\mathbf{L}\mathbf{m}-\mathbf{d})}. && (12) \end{aligned}$$

Since the posterior probability $p(\mathbf{m},\mathbf{d})$ is a quantity describing the probability that the model is correct given a certain set of observations, we would like it to be maximized. Maximizing the posterior function is equivalent to finding the minimum of the following objective function:

$$J(\mathbf{m}) = (\mathbf{L}\mathbf{m}-\mathbf{d})'(\mathbf{L}\mathbf{m}-\mathbf{d}) + \epsilon\mathbf{m}'\mathbf{m}, \tag{13}$$

where $\epsilon = \frac{\sigma^2}{\sigma_m^2}$. We can see that if the model parameters are assumed to have a Gaussian distribution, the solution of maximum posterior probability under Bayes's theorem is equivalent to the damped least-squares solution. Thus, Bayesian inversion gives another perspective on the same problem, where minimizing the model residuals in the L_2 norm corresponds to a Gaussian prior probability distribution. As the Gaussian distribution is a short-tailed function that is tightly centered around the mean, it will result in a smooth solution. Therefore, when a sparse solution is required, the L_2 norm is no longer appropriate, and long-tailed distribution functions such as the exponential and Cauchy distributions should be chosen for the prior probability distribution of $\mathbf{m}$.

If we keep equation (8) unchanged and assume $p(\mathbf{m})$ satisfies the exponential probability distribution, with a mean of zero:

$$\begin{aligned} p(\mathbf{m}) &= \frac{1}{2\sigma_m} e^{\frac{-1}{\sigma_m}\sum_{i=1}^{M}|m_i|} && (14)\\ &= \alpha_2 e^{\frac{-1}{\sigma_m}\sum_{i=1}^{M}|m_i|}. && (15) \end{aligned}$$

Then the *a posteriori* probability becomes

$$p(\mathbf{m},\mathbf{d}) \propto \alpha_1\alpha_2 e^{-\frac{1}{2\sigma^2}(\mathbf{L}\mathbf{m}-\mathbf{d})'(\mathbf{L}\mathbf{m}-\mathbf{d})-\frac{1}{\sigma_m}\sum_{i=1}^{M}|m_i|}. \tag{16}$$

Finding the maximum of the above function is equivalent to finding the minimum of the following objective function:

$$J(\mathbf{m}) = (\mathbf{Lm} - \mathbf{d})'(\mathbf{Lm} - \mathbf{d}) + \epsilon f(\mathbf{m}), \tag{17}$$

where $\epsilon = \frac{2\sigma^2}{\sigma_m}$, and the regularization term, $f(\mathbf{m})$, is defined by

$$f(\mathbf{m}) = \sum_{i=1}^{M} |m_i|. \tag{18}$$

Therefore, by adding an L_1 norm regularization term to the least-squares problem, we get a sparse solution of the model parameters.

Though minimizing the objective function (17) results in a non-linear problem, it can be solved efficiently by Iteratively Re-weighted Least-Squares (IRLS). The gradient of the regularization term is

$$\bigtriangledown f(\mathbf{m}) = \frac{\partial}{\partial \mathbf{m}} f(\mathbf{m}) = \begin{pmatrix} \frac{\partial f(\mathbf{m})}{\partial m_1} \\ \frac{\partial f(\mathbf{m})}{\partial m_2} \\ \vdots \\ \frac{\partial f(\mathbf{m})}{\partial m_M} \end{pmatrix} = \begin{pmatrix} \frac{m_1}{|m_1|} \\ \frac{m_2}{|m_2|} \\ \vdots \\ \frac{m_M}{|m_M|} \end{pmatrix}. \tag{19}$$

Therefore the regularization term $f(\mathbf{m})$, which minimizes the residual in the L_1 norm, can be solved in the L_2 norm by introducing a diagonal weighting function $\mathbf{W_m}$:

$$\mathbf{W_m} = \mathbf{diag}(\sqrt{\frac{1}{|m_1|}}, \sqrt{\frac{1}{|m_2|}}, \cdots, \sqrt{\frac{1}{|m_M|}}). \tag{20}$$

Then the objective function (17) can be changed to

$$\begin{aligned} J(\mathbf{m}) &= (\mathbf{Lm} - \mathbf{d})'(\mathbf{Lm} - \mathbf{d}) + \epsilon(\mathbf{W_m m})'(\mathbf{W_m m}) & (21) \\ &= \|\mathbf{Lm} - \mathbf{d}\|_{\mathbf{2}} + \epsilon \|\mathbf{W_m m}\|_{\mathbf{2}}. & (22) \end{aligned}$$

The derivation of the objective function for the *a priori* case with a Cauchy distribution is similar, except the diagonal weighting function changes to the following:

$$\mathbf{W_m} = \mathbf{diag}(\sqrt{\frac{1}{1+(m_1/\sigma_m)^2}}, \sqrt{\frac{1}{1+(m_2/\sigma_m)^2}}, \cdots, \sqrt{\frac{1}{1+(m_M/\sigma_m)^2}}). \tag{23}$$

Since $\mathbf{W_m}$ is a function of the model $\mathbf{m}$, if we use the gradient-based IRLS method to solve the objective function (22), we have to recompute the weight $\mathbf{W_m}$ at each iteration and the algorithm can be summarized as follows:

1. At the first iteration, $\mathbf{W_m}$ is set to be the identity matrix:

$$\mathbf{W_m^0} = \mathbf{I} \tag{24}$$

2. At the kth iteration, we solve the following fitting goals:

$$0 \approx \mathbf{Lm^k - d} \tag{25}$$
$$0 \approx \epsilon \mathbf{W_m^{k-1} m^k}, \tag{26}$$

where $W_{mi}^{k-1} = (|m_i^{k-1}|)^{-1/2}$ in the case of the L_1 norm, or $W_{mi}^{k-1} = \left[1 + \left(\frac{m_i^k}{\sigma_m}\right)^2\right]^{-1/2}$ in the case of the Cauchy norm.

REGULARIZED LEAST-SQUARES INVERSION

To force the energy in the SODCIGs to concentrate at the zero-offset location, we can pose the problem as a regularized inversion process, and the objective function is defined as follows:

$$J(\mathbf{m}) = \|\mathbf{W_d}(\mathbf{Lm - d})\|_2 + f\left(D(\mathbf{m})\right), \tag{27}$$

where $\mathbf{d}$ is the recorded data. $\mathbf{L}$ is a 2-D/3-D wave-equation modeling operator that transforms the model to prestack data; here I use the adjoint of the Double Square Root (DSR) migration operator. $\mathbf{W_d}$ is a mask weight, which enables us to minimize the data residuals only at known locations, and $\mathbf{m}$ is the model space in terms of the SODCIGs, a 3-D image cube in the 2-D case and a 5-D image cube in the 3-D case. Operator $D(\cdot)$ is defined as follows:

$$D(\mathbf{m}) = \mathbf{diag}(|\mathbf{h}|)\mathbf{m} \tag{28}$$
$$\mathbf{diag}(|\mathbf{h}|) = \mathbf{diag}(|h_1|, |h_2|, \cdots, |h_M|) \tag{29}$$

which is the DSO operator acting along the offset dimension to penalize energy far from zero-offset locations. Near-offset energy, especially that around zero-offset locations where $|\mathbf{h}| \approx 0$, will not be affected. After applying DSO, the model-dependent sparseness transform operator $f(\cdot)$, which minimizes model residuals in the L_1 norm or Cauchy norm, is performed. The sparseness constraint is applied depth-by-depth and CMP-by-CMP along the offset dimension. The purpose of adding such a sparseness constraint is to penalize noise which is typically incoherent and weak, and consequently enhance the resolution of the final inverted result.

In fact, adding the DSO regularization term in the SODCIGs is similar to adding a roughener to smooth along the offset-ray parameters in the ADCIGs. As offset-ray parameters are connected to the offset wavenumbers via the following equation:

$$\mathbf{p_h} = \frac{\mathbf{k_h}}{\omega}, \tag{30}$$

for a single frequency,

$$\frac{\partial}{\partial \mathbf{p_h}} = \omega \frac{\partial}{\partial \mathbf{k_h}}, \tag{31}$$

therefore, for a specific CMP location $\mathbf{m_0}$, roughening along ray-parameters can be expressed as follows:

$$\begin{aligned} 0 &\approx \frac{\partial}{\partial \mathbf{p_h}} P(\tau, \mathbf{m_0}, \mathbf{p_h}, z) && (32)\\ &= \omega \frac{\partial}{\partial \mathbf{k_h}} S\mathcal{F}_\omega^{-1} P(\omega, \mathbf{m_0}, \mathbf{k_h}, z) && (33)\\ &= \omega S\mathcal{F}_\omega^{-1} \frac{\partial}{\partial \mathbf{k_h}} P(\omega, \mathbf{m_0}, \mathbf{k_h}, z), && (34)\end{aligned}$$

where S is the slant-stack operator and $\mathcal{F}_\omega^{-1}$ is the inverse Fourier transform over frequencies. Considering the Fourier duality, convolving the wavefield $P(\omega, \mathbf{m_0}, \mathbf{k_h}, z)$ with the differential operator in the offset-wavenumber domain has the same effect as multiplying the wavefield $P(\omega, \mathbf{m_0}, \mathbf{h}, z)$ with $\mathbf{diag(h)}$ in the offset-space domain:

$$\frac{\partial}{\partial \mathbf{k_h}} P(\omega, \mathbf{m_0}, \mathbf{k_h}, z) \Leftrightarrow \mathbf{diag(h)} P(\omega, \mathbf{m_0}, \mathbf{h}, z). \quad (35)$$

So smoothing along the offset-ray parameters acts like the DSO regularization term.

Following the discussion in the previous section, the objective function (27) can be rewritten as follows:

$$J(\mathbf{m}) = \|\mathbf{W_d}(\mathbf{Lm} - \mathbf{d})\|_2 + \|\mathbf{W_s} D(\mathbf{m})\|_2, \quad (36)$$

or more concisely in terms of fitting goals,

$$\begin{aligned} 0 &\approx \mathbf{W_d}(\mathbf{Lm} - \mathbf{d}) && (37)\\ 0 &\approx \epsilon \mathbf{W_s} D(\mathbf{m}), && (38)\end{aligned}$$

where $\mathbf{W_s}$ is the diagonal weighting matrix that forces the sparseness constraints. Here I use the Cauchy norm, thus

$$W_{si} = \frac{1}{\sqrt{1 + (D(m_i)/\sigma)^2}}. \quad (39)$$

DIAGONAL APPROXIMATION OF HESSIAN MATRIX

Solving fitting goals (37) and (38) is expensive, since we have to propagate wavefields downward and upward within each iteration, with the cost for each iteration equal to the cost of two migrations. For small-scale problems, it is solvable; for large-scale problems, however, the computational cost might be prohibitive. What's more, currently there are no universal criteria for choosing the hyperparameters: ϵ, which balances the data-fitting goal and the model-styling goal, and σ, which controls the sparseness of the model space. They can be decided only by trial and error, which obviously is not practical for very large-scale problems.

Instead of propagating wavefields at each iteration, however, we can precompute the Hessian or approximate it with a diagonal matrix and then solve the modified fitting goals iteratively. The solution of fitting goal (37) in the least-squares sense is

$$\mathbf{m} \approx \left(\mathbf{L}'\mathbf{W}_\mathbf{d}'\mathbf{W}_\mathbf{d}\mathbf{L}\right)^{-1}(\mathbf{W}_\mathbf{d}\mathbf{L})'\mathbf{W}_\mathbf{d}\mathbf{d}. \tag{40}$$

The weighted Hessian matrix $\mathbf{H} = \mathbf{L}'\mathbf{W}_\mathbf{d}'\mathbf{W}_\mathbf{d}\mathbf{L}$ can be either fully computed (Valenciano and Biondi, 2004) or approximated with a diagonal matrix; here I do the latter, approximating the weighted Hessian with its diagonals as follows (Rickett, 2003):

$$\mathbf{H} \approx \mathbf{W_H} = \frac{\mathbf{diag}(\mathbf{L}'\mathbf{W}_\mathbf{d}'\mathbf{W}_\mathbf{d}\mathbf{L}\mathbf{m_{ref}})}{\mathbf{diag}(\mathbf{m_{ref}})}, \tag{41}$$

and I choose the migrated image cube as the reference image cube:

$$\mathbf{m_{ref}} = (\mathbf{W_d}\mathbf{L})'\mathbf{W_d}\mathbf{d}. \tag{42}$$

Therefore, fitting goals (37) and (38) can be modified as follows:

$$0 \approx \mathbf{W_H}\mathbf{m} - \mathbf{m_{mig}} \tag{43}$$
$$0 \approx \epsilon\mathbf{W_s}D(\mathbf{m}), \tag{44}$$

where $\mathbf{m_{mig}} = (\mathbf{W_d}\mathbf{L})'\mathbf{W_d}\mathbf{d}$, which is obtained by migrating the recorded data. To avoid the right-hand side of equation (41) being divided by zeros, I multiply $\mathbf{diag}(\mathbf{m_{ref}})$ on both sides of equation (43), resulting in

$$0 \approx \mathbf{W_{refm}}\mathbf{m} - \mathbf{W_{refd}}\mathbf{m_{mig}} \tag{45}$$
$$0 \approx \epsilon\mathbf{W_s}D(\mathbf{m}), \tag{46}$$

where $\mathbf{W_{refm}} = \mathbf{diag}(\mathbf{L}'\mathbf{W}_\mathbf{d}'\mathbf{W}_\mathbf{d}\mathbf{L}\mathbf{m_{ref}})$ and $\mathbf{W_{remd}} = \mathbf{diag}(\mathbf{m_{ref}})$. Fitting goals (45) and (46) can be solved by using the IRLS algorithm described in the previous section.

SYNTHETIC DATA EXAMPLES

I test my methodology on two synthetic 2-D data sets. One shown in Figure 2(a) is a two-layer model with one reflector being horizontal and the other dipping at 15°. The velocity increases with depth: $v(z) = 2000 + 0.3z$, which is shown in Figure 1. To make the synthetic data set more realistic, some random noise has also been added. Then I replace approximately 70% of the traces in the offset dimension with zeros. The incomplete and sparse data set is shown in Figure 2(b). Then I perform DSR migration on both data sets to generate the SODCIGs; the corresponding migrated image cubes are shown in Figure 3. Comparing Figure 3(a) with Figure 3(b), we can see that even with the complete data set (Figure 2(a)), the SODCIGs suffer from the amplitude smearing effects caused by the offset truncation. The situation gets worse as the offset coverage is further reduced; there are severe amplitude smearing and aliasing artifacts in the SODCIGs as shown in Figure 3(b), and because of the interference of these artifacts in the offset domain, the resolution of the migrated image (i.e. offset=0) is also

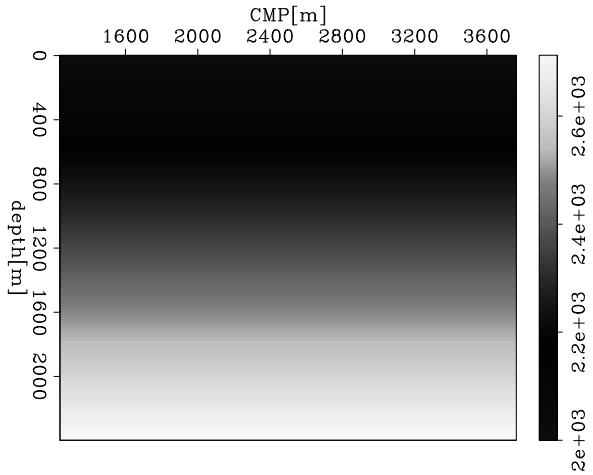

Figure 1: The velocity model for the two-layer model. yaxun1-layer_vel [ER]

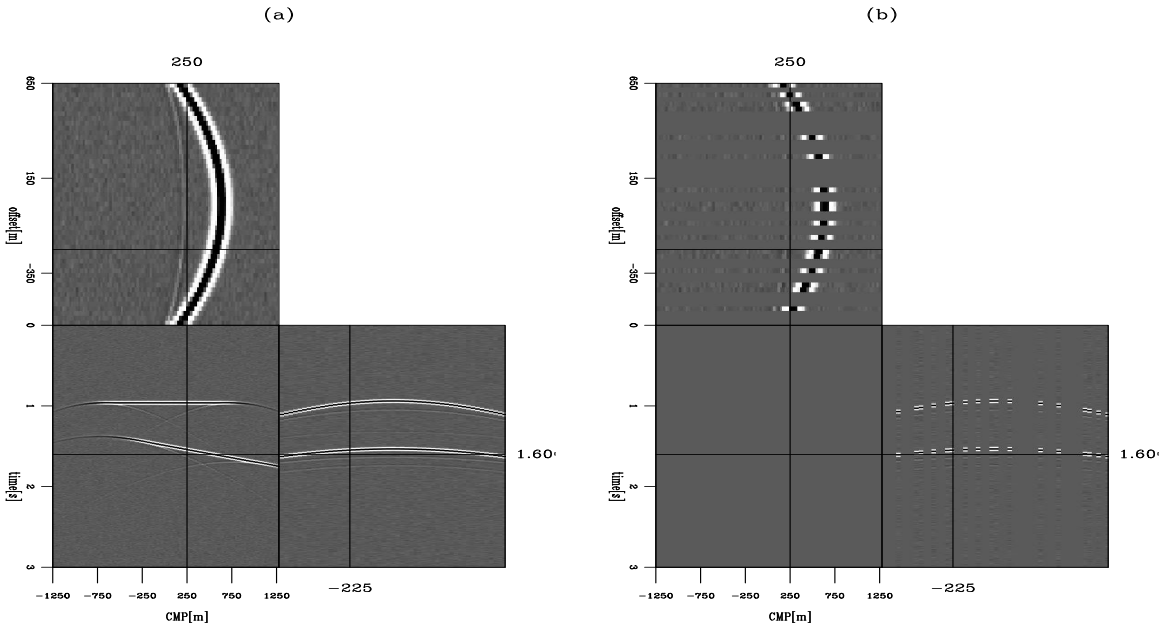

Figure 2: (a) The synthetic data set for the two-layer model, (b) the incomplete data set with about 70% of the traces in the offset dimension replaced by zeros. yaxun1-layer_mod [ER]

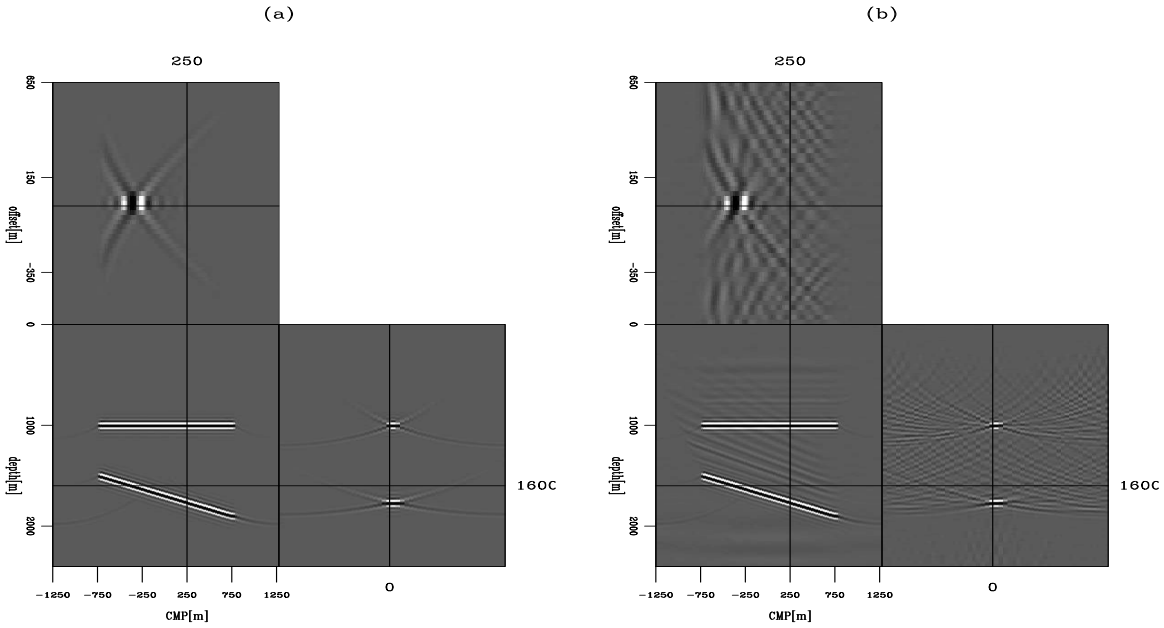

Figure 3: SODCIGs for the two-layer model, (a) obtained by migrating Figure 2(a) and (b) obtained by migrating Figure 2(b). In both plots, the panel in the middle shows the migrated image ($h = 0$), the panel on the right shows the SODCIGs and the panel on the top shows the depth slice. yaxun1-layer_sodcig [ER]

degraded. The effect is more obvious if we transform the SODCIGs into the ADCIGs, which are shown in Figure 4; there are some gaps in the middle of the ADCIGs (Figure 4(b)) obtained by migrating the incomplete data set, indicating that there are some illumination problems.

From this simple experiment, we intuitively understand that the amplitude smearing in the SODCIGs is another representation of poor illumination and that the more energy smearing we see in the SODCIGs, the more severe the illumination problem must be. Therefore, if we could make the energy more concentrated at zero-offset and penalize the energy at nonzero-offset, we would compensate for the illumination problem and fill the holes in the ADCIGs. To achieve this purpose, I first approximate the weighted Hessian matrix with equation (41), then solve the inversion problem based on the fitting goals (45) and (46). The reference image $\mathbf{m_{ref}}$ or $\mathbf{W_{refm}}$ is chosen to be the migrated image cube of the incomplete data, which is shown in Figure 2(b). The weight $\mathbf{W_{refd}}$ is created by demigrating $\mathbf{m_{ref}}$ and then migrating the demigrated image again. The mask weight is shown in Figure 5. As I apply the sparseness constraint along the offset dimension depth-by-depth and CMP-by-CMP, it would be inappropriate to use a global parameter σ to control the sparseness; therefore I apply σ locally, choosing for its value the mean value of the current offset vector. The final inversion result is shown in Figure 6(a); for comparison, Figure 6(b) shows the migration result. Figure 7 illustrates one single trace located at CMP=0 meters and offset=0 meters, Figure 7(a) is the result by migration, while Figure 7(b) is the result by inversion, where both (a) and (b) are normalized to compare their relative amplitude ratios. From the results we can clearly see that the DSO regularization term perfectly eliminates the energy at non-zero offset. The sparse-

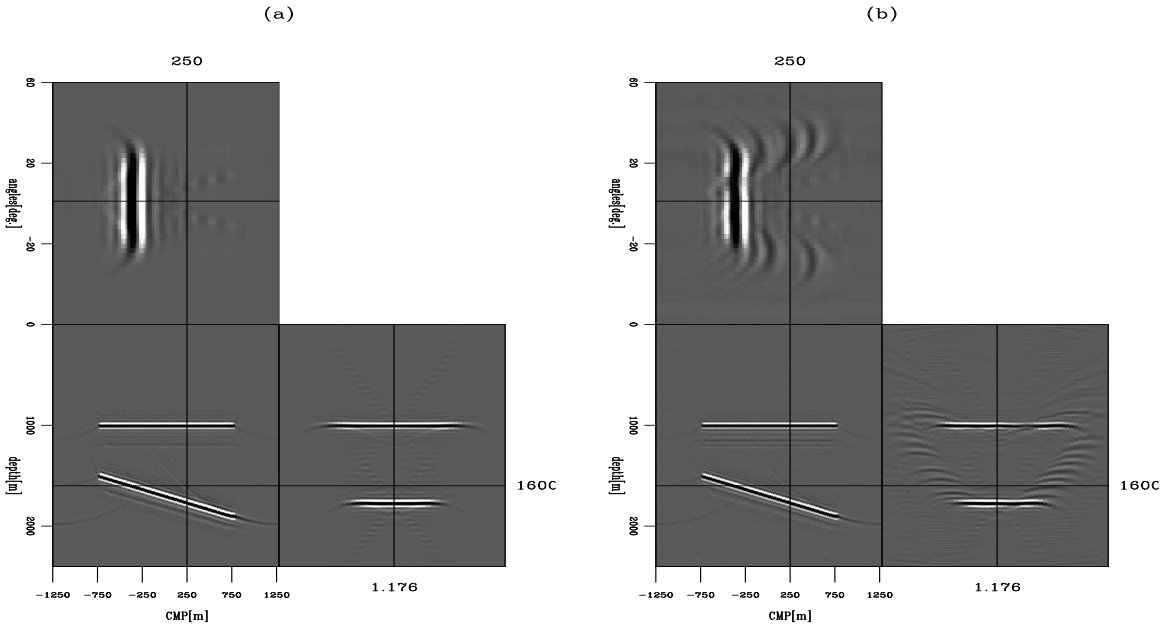

Figure 4: ADCIGs for the two-layer model, (a) computed from Figure 3(a), and (b) computed from Figure 3(b). In both plots, the panel in the middle shows the image for each opening angle, the panel on the right shows the ADCIGs and the panel on the top shows the depth slice. yaxun1-layer_adcig [ER]

ness constraint also successfully penalizes weak amplitudes and consequently improves the resolution of the image. Figure 8 shows the comparison of ADCIGs between migration and inversion, where, as expected, the inversion result in Figure 8(a) fills the illumination gaps presented in Figure 8(b).

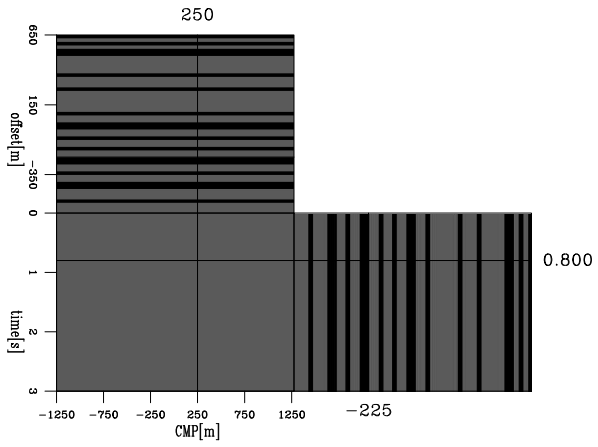

Figure 5: The computed mask weight from Figure 2(b). Black stands for ones, while grey stands for zeros. yaxun1-layer_rn70_mask [ER]

The model with two reflectors in the previous example is simple. To test whether the inversion scheme works for complex models, I apply it to the Marmousi model, which is shown in Figure 9(a), again with about 70% of the traces in the offset dimension replaced with zeros. The computed mask weight is shown in Figure 9(b). As before, I use the migrated

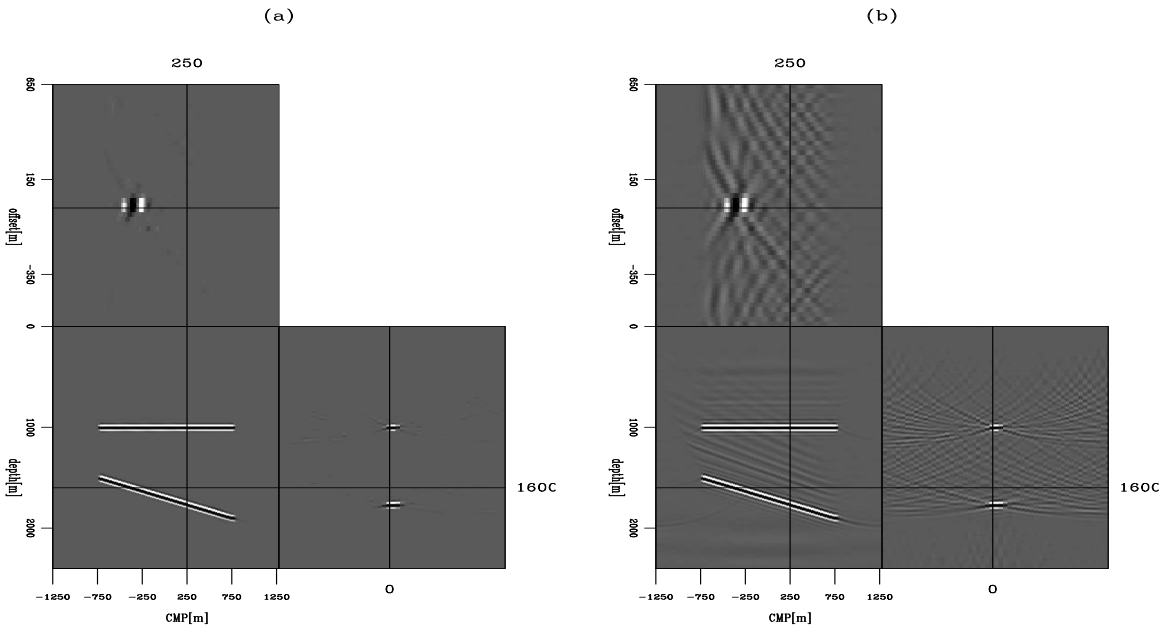

Figure 6: SODCIGs for the two-layer model. (a) The inversion result, and (b) the migration result. Note the inversion result has perfectly penalized the energy far from zero-offset locations and the sidelobes of the amplitudes as well. yaxun1-layer_inv_sodcig [ER]

image cube as the reference image cube for computing the weighting matrices $\mathbf{W_{refd}}$ and $\mathbf{W_{refm}}$. The parameter σ is also chosen to be the mean value of the current offset vector. Because there are no good suggestions for the parameter ϵ, it is chosen by trial and error to get a satisfactory result. Since I use only one reference velocity (the average between the maximum and the minimum velocities at each depth step) for the DSR-SSF algorithm, some steeply dipping faults are not well imaged, and because of the inaccuracy of the reference velocity, some locations are mispositioned, indicating there should be some residual moveout in both SODCIGs and ADCIGs.

The final inversion result is shown in Figure10 (b); for comparison, Figure10(a) is the migration result. By using the approximated inversion scheme, we suppress the weak and incoherent noise and obtain a much cleaner result, while also improving the resulotion to some extent. This is more obvious if we extract a single trace from the migration result and the inversion result to compare their relative amplitudes. Figure 11 shows the extracted trace located at CMP=4 km, offset=0 km, while Figure 12 shows the extracted trace located at CMP=7.5 km, offset=0 km. In both figures, (a) is obtained from the migration result, while (b) is obtained from the inversion result. From Figure 11 and Figure 12, we can see that small amplitudes and the sidelobes of the wavelets are penalized by the inversion scheme and the inversion result yields an image with higher resolution. But also notice that some weak reflections which are presented in the migration result are attenuated in the inversion result.

Figure 13 illustrates the SODCIGs for two different locations; (a) and (c) are the SODCIGs at CMP=4 km and CMP=7.5 km respectively obtained from the migration result, while (b) and

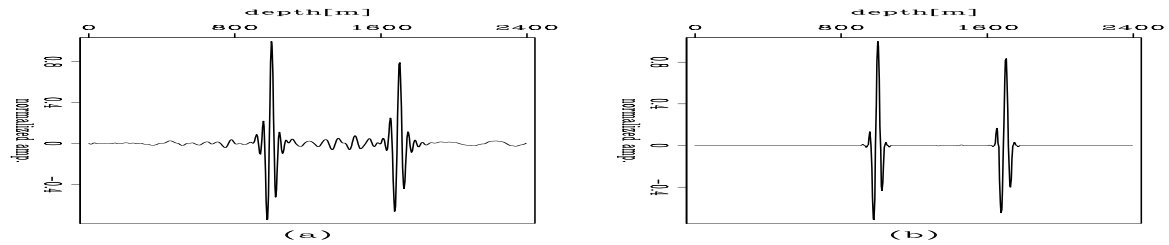

Figure 7: Comparison of a single trace located at CMP=0 meter and offset=0 meter. (a) The migration result, (b) the inversion result. The amplitudes in both (a) and (b) are normalized to compare their relative ratios. yaxun1-layer_wavelet [ER]

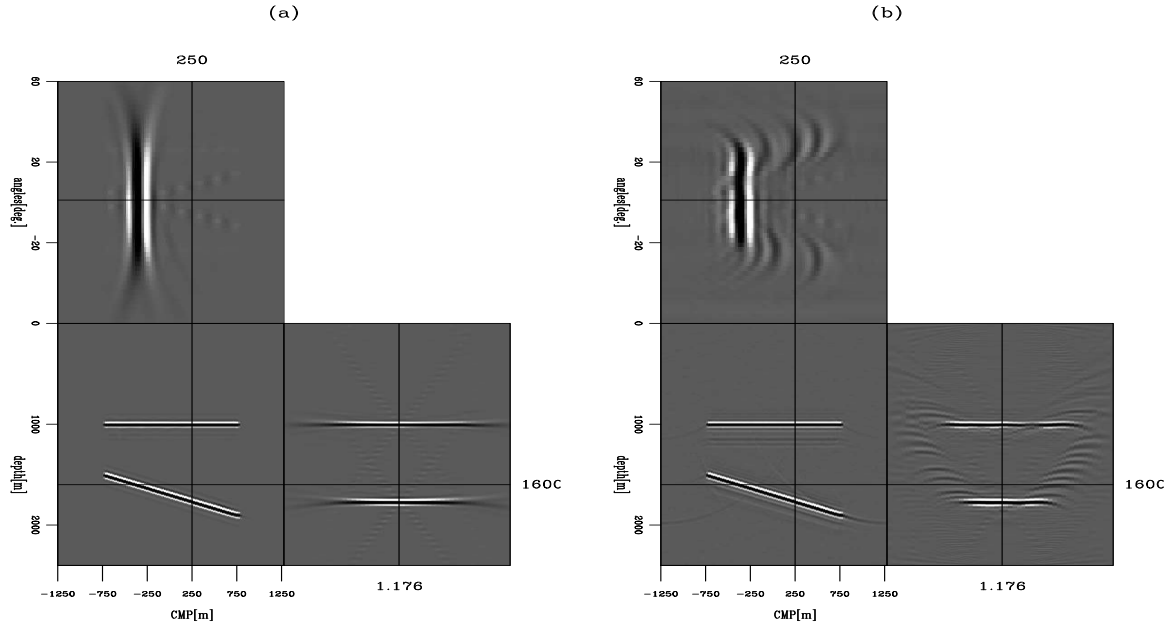

Figure 8: ADCIGs for the two-layer model. (a) The inversion result, (b) the migration result. Note the inversion has filled in the illumination holes. yaxun1-layer_inv_adcig [ER]

(d) show the SODCIGs at the same CMP locations obtained from the inversion result. Because of the DSO regularization term in the inversion scheme, events that are far from zero-offset locations are penalized, making the energy more concentrated at zero-offset. The ADCIGs at the corresponding locations shown in Figure 14 explain this further, with the ADCIGs (Figure 14(b) and (d)) from the inversion result smoothed across angles and the illumination holes present in (a) and (c) filled in to some degree.

As mentioned above, because of the inaccuracy of the reference velocity, there are still some residual moveouts at some locations in both SODCIGs and ADCIGs, as seen in Figure 13(a) and Figure 14(a). One nice thing to see is by choosing a proper trade-off parameter ϵ, the proposed inversion scheme can successfully preserve the residual moveouts both in SODCIGs and ADCIGs, as shown in Figure 13(b) and Figure 14(b). The angle gathers even get cleaner, which makes it much easier to estimate the residual moveouts. Therefore, this

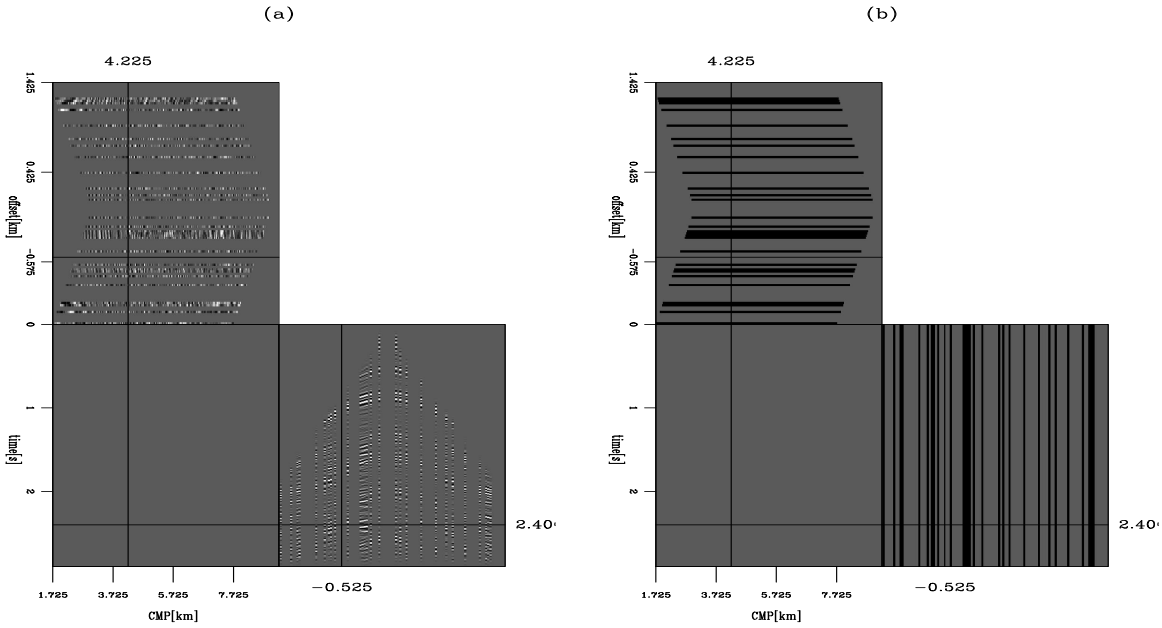

Figure 9: The Marmousi data set (a) and the corresponding mask weight (b), where black stands for ones, while grey stands for zeros. yaxun1-mar_model [ER]

approximated inversion scheme may have the potential to improve the accuracy of residual moveout estimation, and consequently improve velocity estimation results. However, this still needs further investigation.

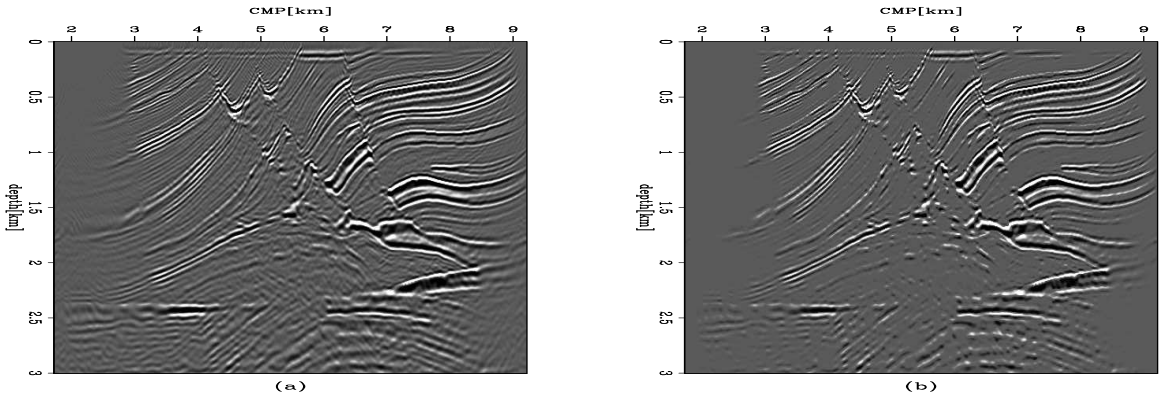

Figure 10: Comparison of the migration result and the inversion result. (a) The image obtained by migration, and (b) the image obtained by inversion. yaxun1-mar_h0 [CR]

DISCUSSION AND CONCLUSION

I have presented a regularized inversion scheme in the SODCIGs to deal with the artifacts caused by insufficient offset coverage. The inversion scheme concentrates the migrated energy

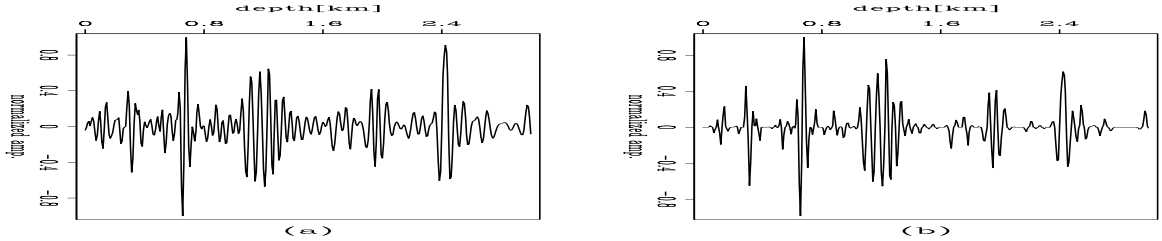

Figure 11: Comparison of a single trace located at CMP=4 km and offset=0 km. (a) The result from migration, and (b) the result from inversion. The amplitudes in both (a) and (b) are normalized to compare their relative ratios. yaxun1-mar_wavelet1 [CR]

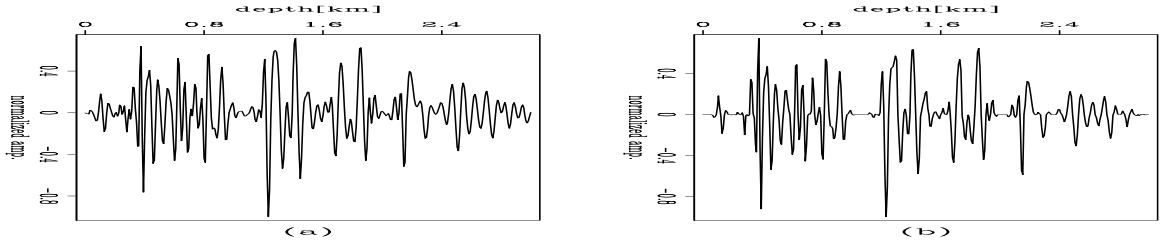

Figure 12: Comparison of a single trace located at CMP=7.5 km and offset=0 km. (a) The result from migration, and (b) the result from inversion. The amplitudes in both (a) and (b) are normalized to compare their relative ratios. yaxun1-mar_wavelet2 [CR]

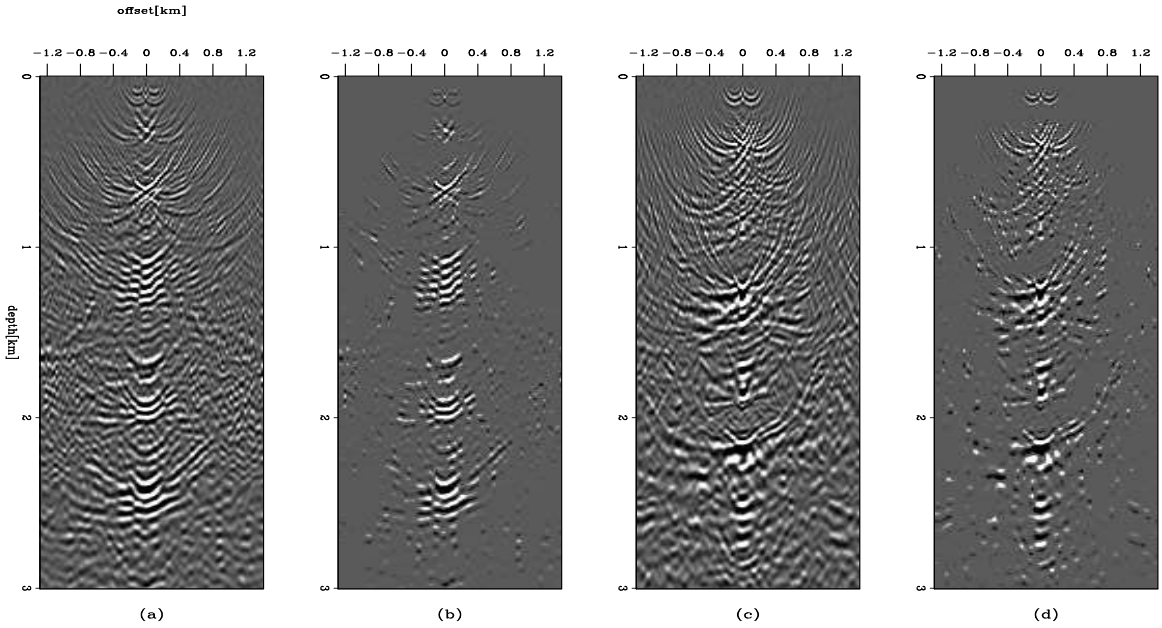

Figure 13: Subsurface-offset-domain common-image gathers for two different surface locations. Panels (a) and (c) are the SODCIGs at CMP=4.0 km and CMP=7.5 km obtained by migration, while (b) and (d) are the corresponding SODCIGs obtained by inversion. yaxun1-mar_sodcig [CR]

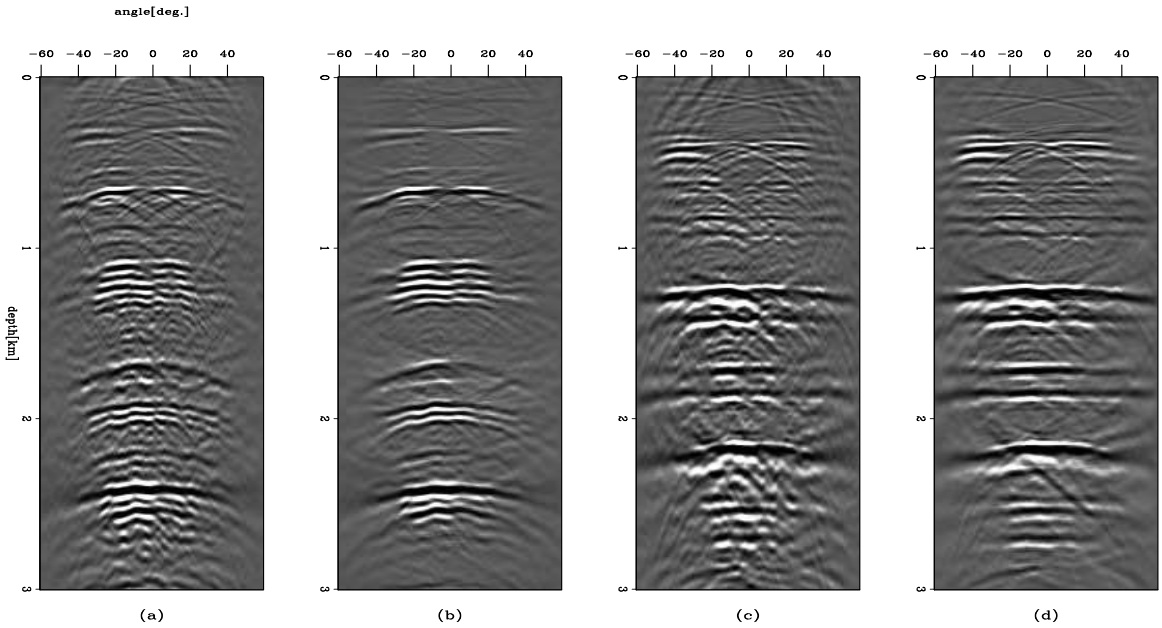

Figure 14: Angle-domain common-image gathers for two different surface locations. Panels (a) and (c) are the ADCIGs at CMP=4.0 km and CMP=7.5 km obtained by migration, while (b) and (d) are the corresponding ADCIGs obtained by inversion. yaxun1-mar_adcig [CR]

at zero-offset locations and removes incoherent and weak noise by constraining the solution with the DSO and sparseness operator. Though I tested my approach only on 2-D data sets, it would be quite easy to extend it to 3-D, since I approximate the Hessian with a diagonal matrix, which reduces the computational expense.

Compared to regularizing in the ADCIGs with a roughening operator acting along the angle axis, regularization in the SODCIGs has the advantage of being computationally cheaper. More importantly, with proper selection of the trade-off parameter ϵ, it can preserve the velocity information correctly when a wrong migration velocity is used. Therefore it may have the potential to update the velocity more accurately, since it can produce much cleaner angle gathers.

The proposed inversion scheme may also be dangerous if we choose inappropriate hyperparameters ϵ and σ. Since by adding the sparseness constraint in the image cube, we run the risk of penalizing true reflections that have very weak energy, over-regularization may lead to too-sparse solutions, forfeiting the ability to image weak reflections. For the example of the Marmousi model, we can clearly see that some weak reflections are greatly attenuated.

The deconvolution effects, i.e. the wavelet-squeezing effects, in the above examples are not obvious; this is because of the approximation of the Hessian with a diagonal matrix. The approximated diagonal deconvolution filter is not sufficient to deconvolve the image accurately, especially for the complex Marmousi model. A more accurate but also more expensive way is to compute the full Hessian with wave equations instead of approximating it with a diagonal matrix.

ACKNOWLEDGMENTS

I would like to thank Bob Clapp, Guojian Shan, Jeff Shragge, Alejandro Valenciano and Bill Curry for useful discussions.

REFERENCES

Kuehl, H. and M. Sacchi, 2001, Generalized least-squares DSR migration using a common angle imaging condtion: Soc. of Expl. Geophys., 71st Ann. Internat. Mtg, 1025–1028.

Prucha, M. L., R. G. Clapp, and B. Biondi, 2000, Seismic image regularization in the reflection angle domain: SEP–**103**, 109–119.

Rickett, J. E., 2003, Illumination-based normalization for wave-equation depth migration: Geophysics, **68**, 1371–1379.

Shen, P., W. Symes, and C. C. Stolk, 2003, Differential semblance velocity analysis by wave-equation migration: 73rd Ann. Internat. Mtg., Soc. of Expl. Geophys., Expanded Abstracts, 2132–2135.

Ulrych, T. J., M. D. Sacchi, and A. Woodbury, 2001, A bayes tour of inversion: A tutorial: Geophysics, **66**, 55–69.

Valenciano, A. A. and B. Biondi, 2004, Target-oriented computation of the wave-equation imaging Hessian: SEP–**117**, 63–76.

Valenciano, A. A., 2006, Target-oriented wave-equation inversion with regularization in the subsurface offset domain: SEP–**124**.