Balancing amplitude and phase in TFWI

Ali Almomin

ABSTRACT

Tomographic Full Waveform Inversion (TFWI) provides a robust and accurate method to invert the seismic data by simultaneously inverting all scales of the model using both amplitude and phase information. However, one shortcoming of TFWI is the large number of iteration required to achieve accurate results due to its slow convergence. In this report, I analyze the source of its slow convergence and propose two modifications to mitigate the problem. First by modifying the formulation of the regularization term to focus more on the phase information, and second by using an alternative enhancing operator that is less sensitive to the amplitudes in the extended model. Then, I test the modified TFWI on the marmousi and BP synthetic models. The results show a significant improvement in the convergence rate.

INTRODUCTION

TFWI, similar to other data-space inversion method, produces highly accurate results due to matching both the amplitude and phase of the data. This is achieved in two steps: first, extending the wave equation and adding an additional axis to the velocity model, and second, adding a regularization term that drives the solution towards a non-extended model. However, one limitation to TFWI in its earlier formation, as shown in Almomin and Biondi (2013), is the large number of iterations required to achieve accurate results. This large number of iterations causes TFWI to be very expensive, especially for large-scale 3D datasets. In Almomin and Biondi (2014a), I proposed a wavelength continuation scheme that preconditioned the gradient and smoothed the updates in the early iterations. While preconditioning did improve the convergence rate, the number of iterations was still in the hundreds.

In this report, I analyze the source of its slow convergence and find that it is due to the unbalanced effects of amplitudes and phase both in the formulation of the regularization term and the enhancing operator. This imbalance resulted in a strong dependence of the kinematic updates on the amplitude fitting, causing it to take more iterations. To mitigate the problem and speed up convergence, I propose two modifications to TFWI. First by modifying the formulation of the regularization term to focus more on the phase information, and second by using an alternative enhancing operator that is less sensitive to the amplitudes in the extended model. Finally, I test the modified TFWI on the marmousi and BP synthetic models.

THEORY

The conventional L2 objective function for data matching can be written as follows:

$$J_{\text{FWI}}(\mathbf{m}) = \frac{1}{2}\|\mathbf{f}(\mathbf{m}) - \mathbf{d}_{\text{obs}}\|_2^2, \tag{1}$$

where $\mathbf{m}$ is the model, $\mathbf{f}$ is the forward modeling operator and $\mathbf{d}_{\text{obs}}$ is the observed surface data. I modify the conventional L2 objective function by adding a nonlinear weighting function as follows:

$$J_{\text{W-TFWI}}(\widetilde{\mathbf{m}}) = \frac{1}{2}\|\mathbf{W}(\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}))\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}) - \mathbf{W}(\mathbf{d}_{\text{obs}})\mathbf{d}_{\text{obs}}\|_2^2, \tag{2}$$

where $\mathbf{W}$ is a diagonal weighting matrix. The elements on the weighting matrix can be described as follows:

$$\mathbf{W}(it, ir, is; \widetilde{\mathbf{f}}) = \frac{1}{\|\mathbf{A}(it, ir, is)\widetilde{\mathbf{f}}\|_2}, \tag{3}$$

where it, ir, is are the time, receiver and source coordinates and the matrix $\mathbf{A}$ represents a convolution by an abstract non-stationary filter. Notice that the norm used in the previous equation is only along the lag axis of the convolution. In other words, the weighting matrix divides each sample by the norm of a filtered version of the data. The purpose of the filter is to window and taper around each sample of the data.

The gradient $\mathbf{g}$ can be calculated as:

$$\widetilde{\mathbf{g}}(\widetilde{\mathbf{m}}) = \left(\frac{\partial \mathbf{r}(\widetilde{\mathbf{m}})}{\partial \widetilde{\mathbf{m}}}\right)^* \mathbf{r}(\widetilde{\mathbf{m}}). \tag{4}$$

Using the definition of the matrix $\mathbf{W}$ and the chain-rule, I can calculate the derivative of the residual with respect to the model as:

$$\frac{\partial \mathbf{r}(\widetilde{\mathbf{m}})}{\partial \widetilde{\mathbf{m}}} = \left(\mathbf{W}(\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})) + \frac{\partial \mathbf{W}(\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}))}{\partial \widetilde{\mathbf{f}}}\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})\right)\frac{\partial \widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})}{\partial \widetilde{\mathbf{m}}}. \tag{5}$$

The derivative of the weighting matrix with respect to the modeled data can be expressed as:

$$\frac{\partial \mathbf{W}(\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}))}{\partial \widetilde{\mathbf{f}}} = -\mathbf{W}(\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}))\frac{\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})^*\mathbf{A}^*\mathbf{A}}{\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})^*\mathbf{A}^*\mathbf{A}\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})}. \tag{6}$$

I can now substitute the previous terms and the gradient as the backprojection of a virtual source $\mathbf{v}(\widetilde{\mathbf{m}})$ described as:

$$\mathbf{v}(\widetilde{\mathbf{m}}) = \mathbf{W}(\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}))\left(\mathbf{r}(\widetilde{\mathbf{m}}) - \frac{\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})^*\mathbf{A}^*\mathbf{A}\mathbf{r}(\widetilde{\mathbf{m}})}{\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})^*\mathbf{A}^*\mathbf{A}\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})}\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})\right). \tag{7}$$

In order to interpret the gradient expression, I redefine the weighting matrix as

$$\mathbf{W}(it, ir, is; \widetilde{\mathbf{f_1}}, \widetilde{\mathbf{f_2}}) = \frac{1}{\sqrt{\widetilde{\mathbf{f_1}}^* \mathbf{A}^*(it, ir, is)\mathbf{A}(it, ir, is)\widetilde{\mathbf{f_2}}}}, \tag{8}$$

where the variable $\mathbf{f}_1$ and $\mathbf{f}_2$ are two data sets. This new definition gives the weighting matrix the freedom to divide by the product of two datasets instead of restricting the calculating to one. I can now write the virtual source as:

$$\mathbf{v}(\widetilde{\mathbf{m}}) = \mathbf{W}(\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}), \widetilde{\mathbf{f}}(\widetilde{\mathbf{m}})) \left(\mathbf{r}(\widetilde{\mathbf{m}}) - \frac{\mathbf{W}^2(\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}), \widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}))}{\mathbf{W}^2(\widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}), \widetilde{\mathbf{r}}(\widetilde{\mathbf{m}}))} \widetilde{\mathbf{f}}(\widetilde{\mathbf{m}}) \right). \tag{9}$$

We can see that the gradient is computed by the following steps. First, the modeled data is divided by the norm of the filtered modeled data. This can be seen as a way to "remove" the amplitudes from the modeled data. Then, the data is scaled by weights which are the result of the product of the modeled data and the residual, which scales the amplitudes of the normalized modeled data to those of the residual. At this point, the modeled data has a similar amplitude scale to the residual, which makes differencing them less prone to scale differences. After calculating the difference, the resultant is divided by the amplitude of the modeled data again. This last division removes the amplitude imprint of the modeled data that will be imposed by the step to follow, which is the back projection of the data into the model space. It is worth noting that this normalization will automatically incorporate the preconditioning described in Almomin and Biondi (2014b)

SYNTHETIC EXAMPLES

I now test the new algorithm on the Marmousi model. I create the observed data with variable density acoustic wave-equation. The first shot gather from the observed data is shown in figure 1. In order to create amplitude discrepancies between the observed and modeled data, I ignore the density variations in the modeled data.

Using the correct kinematics, the first shot of the modeled data is shown in figure 2. Notice that the amplitude behavior between the two figures 1 and 2 is variable both in offset and time. Next, I compare the first TFWI gradient using four objective functions: conventional L2 difference, gather-normalization, trace-normalization (Shen, 2014) and a running-window normalization using a Gaussian window as formulated in this report. Figure 3 shows the first gradient using the L2 difference. This gradient is dominated by large differences in amplitudes in the direct arrivals. Figures 4 and 5 shows the first gradient using the shot-gather-normalization and trace-normalization difference. This gradient has largely improved compared to the L2 gradient, but we can still notice the diminishing amplitudes of the deeper reflectors. Figure 6 shows the first gradient using the running window-normalization difference. This gradient has significantly improved compared to the other two gradients, especially when examining the deeper reflectors.

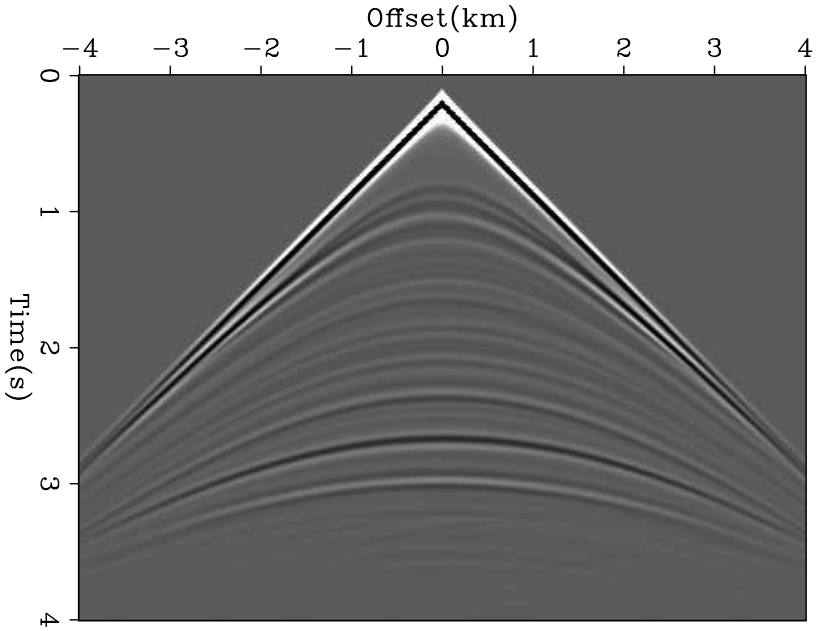

Figure 1: First shot gather from the observed data. [**ER**]

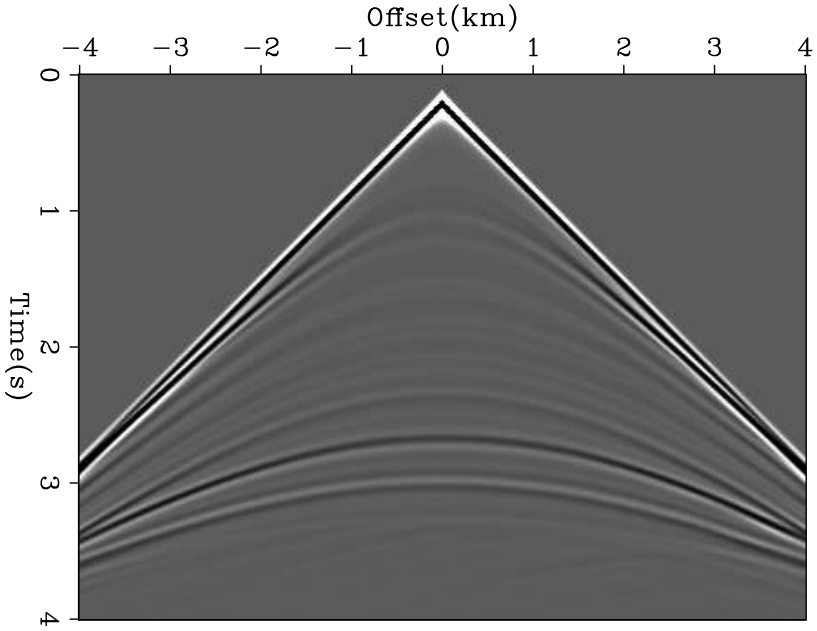

Figure 2: First shot gather from the modeled data using the correct velocity. [**ER**]

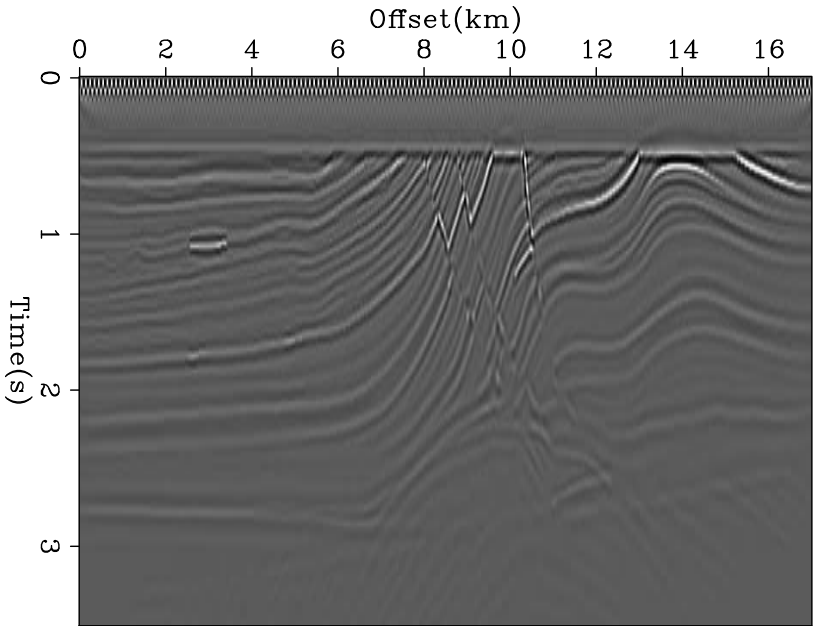

Figure 3: Model gradient using L2 residual and kinematically correct velocity. [**ER**]

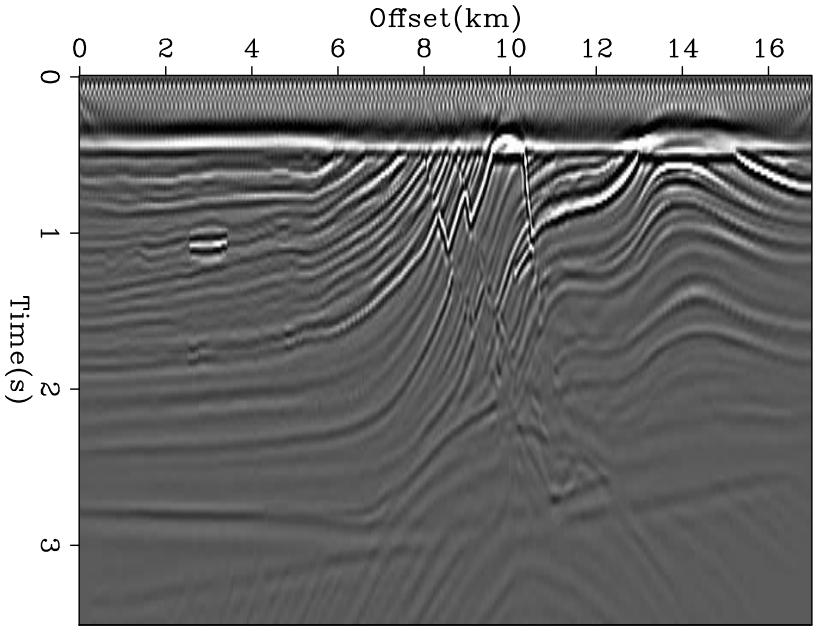

Figure 4: Model gradient using gather-normalized residual and kinematically correct velocity. [**ER**]

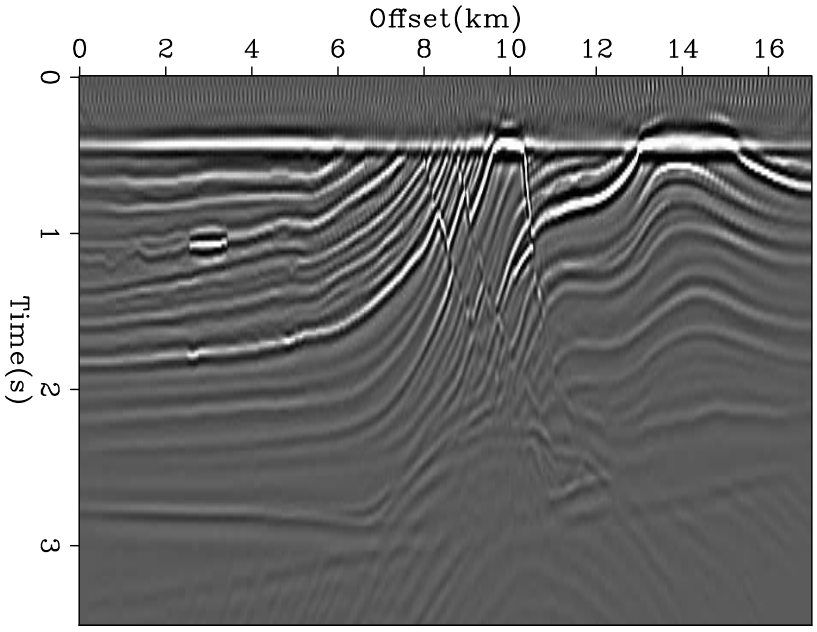

Figure 5: Model gradient using trace-normalized residual and kinematically correct velocity. [**ER**]

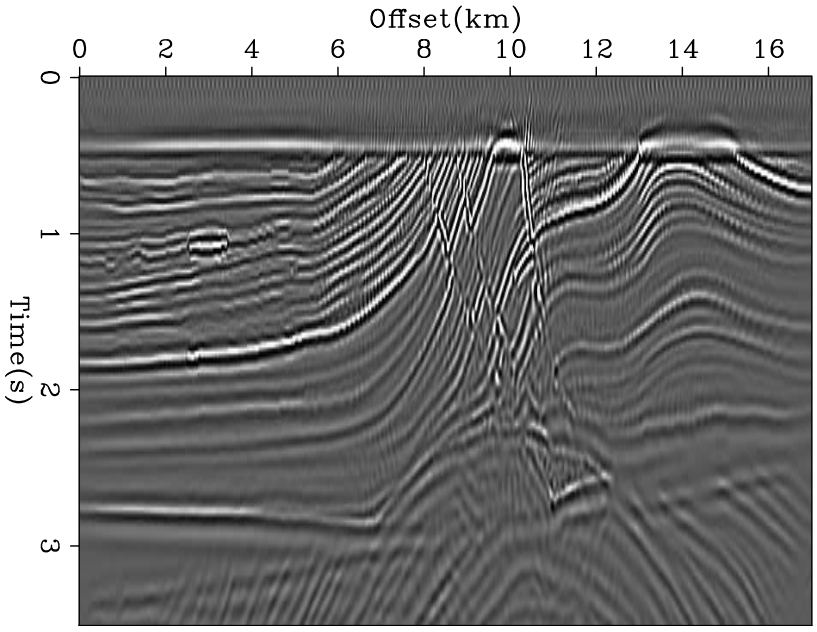

Figure 6: Model gradient using running window-normalized residual and kinematically correct velocity. [**ER**]

Next, I repeat the same experiment but use horizontal average of the velocity as the background, i.e., using the wrong kinematics. Figure 7 shows the first gradient using the L2 difference. Similar to the gradient calculated with the wrong kinematics, this gradient is also dominated by large differences in amplitudes in the direct arrivals. Figures 8 and 9 shows the first gradient using the shot-gather-normalization and trace-normalization difference. While some improvements can be seen compared to the L2 gradient, this gradient is dominated by the first reflector. Figure 10 shows the first gradient using the running-window normalization difference. Also, the low amplitude behavior for the deeper reflectors seems to be compensated for using the running-window normalization.

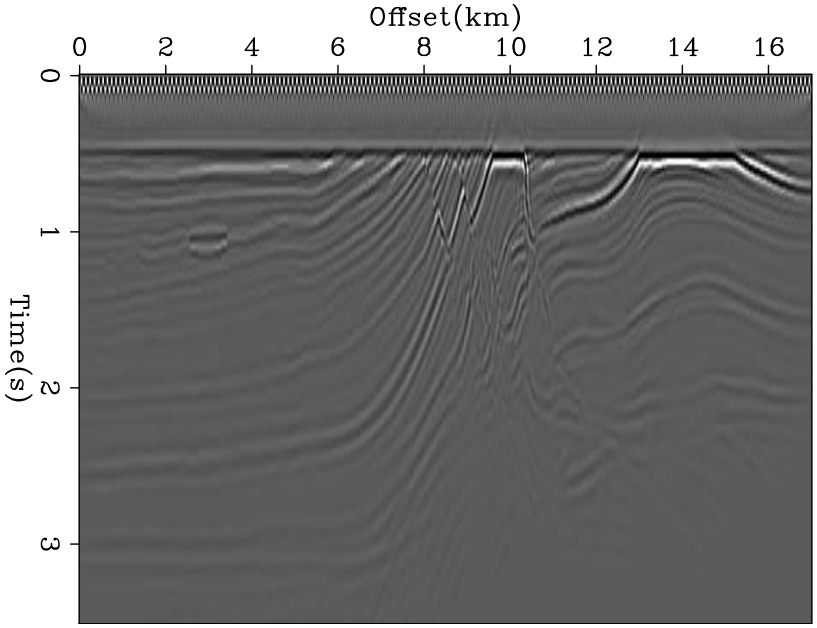

Figure 7: Model gradient using L2 residual and incorrect velocity. [**ER**]

Figures 11 to 14 show the Angle-domain common-image gathers (ADCIGs) at four different locations of the gradients shown in figures 7 to 10. We can see that using gather-normalization or trace-normalization results in a loss of energy at deeper reflectors, especially near the zero angle. The running window-normalization improved the amplitude-versus-angle behavior of the gathers and produced more balanced gathers.

CONCLUSIONS

Matching the amplitudes of the data is important in the framework of TFWI in order to ensure a simultaneous inversion of scales. However, this task can prove challenging when trying to match the amplitudes of field data. An amplitude mismatch can have a large impact on the result, and can dominate the phase component of the data.

To mitigate this issue, I proposed a normalized objective function using an abstract non-linear weighting function. This formulation is very flexible and allows for several

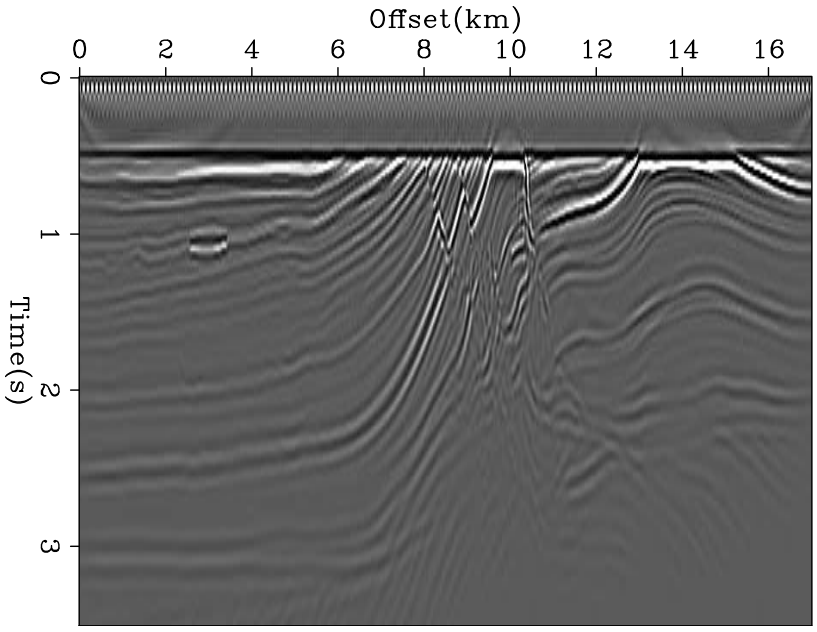

Figure 8: Model gradient using gather-normalized residual and incorrect velocity. [**ER**]

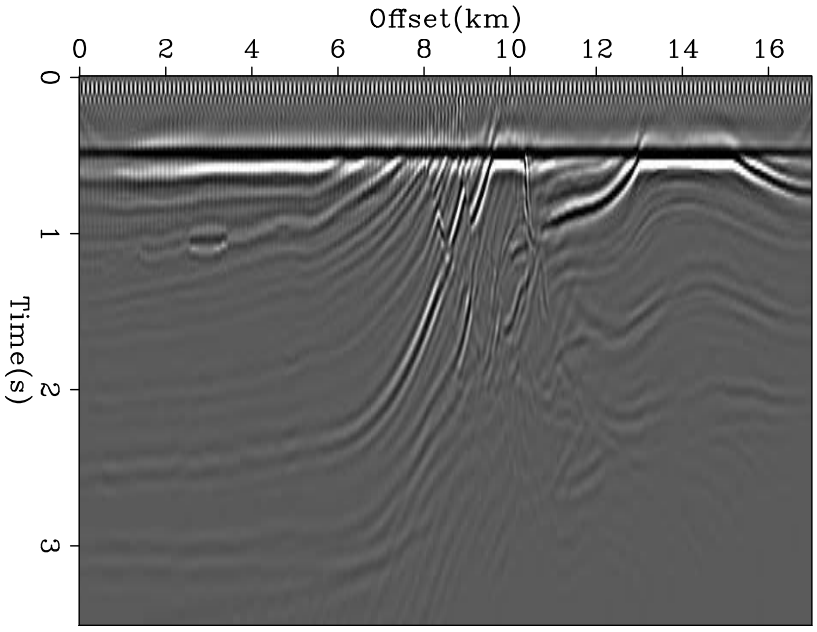

Figure 9: Model gradient using trace-normalized residual and incorrect velocity. [**ER**]

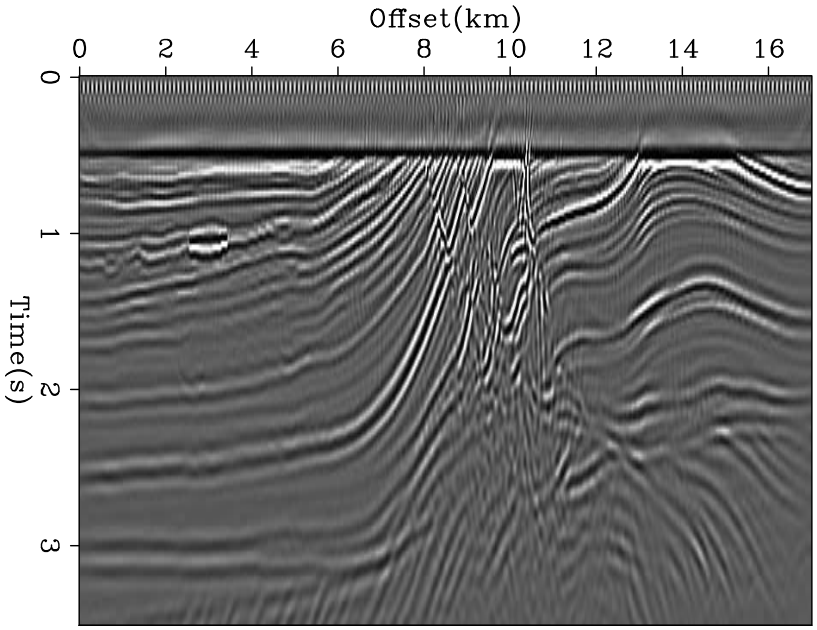

Figure 10: Model gradient using window-normalized residual and incorrect velocity. [**ER**]

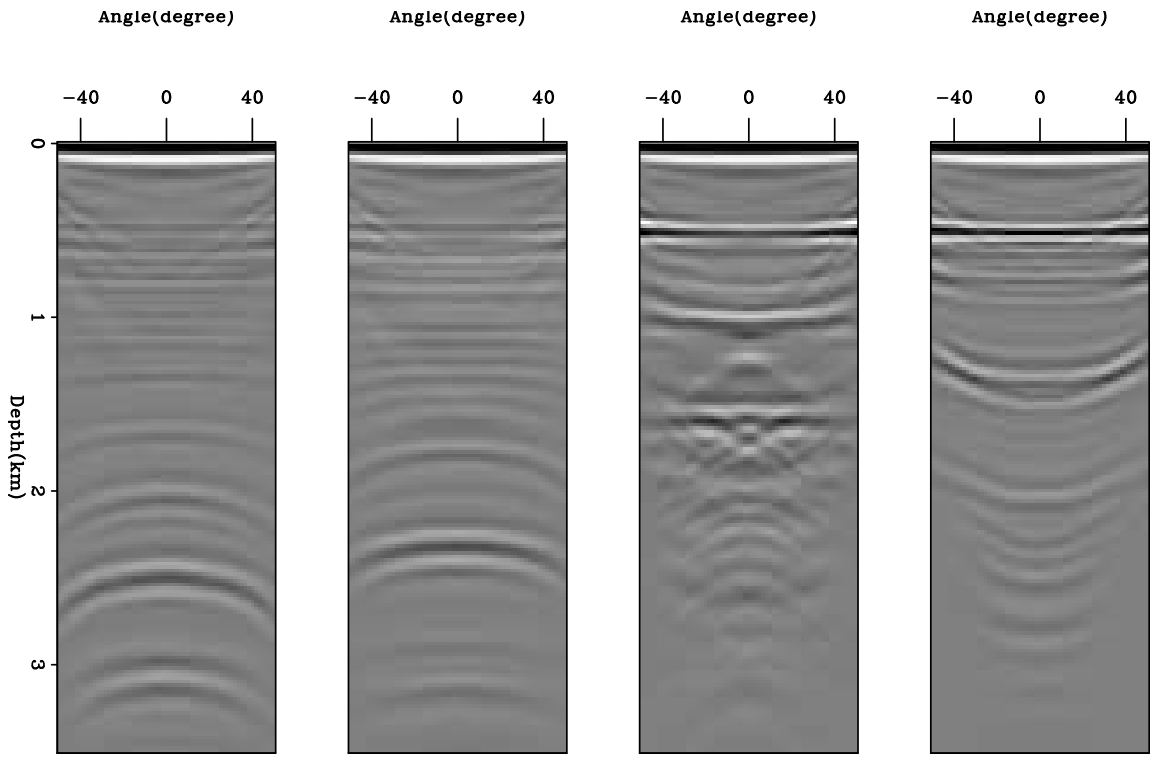

Figure 11: ADCIGs using L2 residual and incorrect velocity. [**ER**]

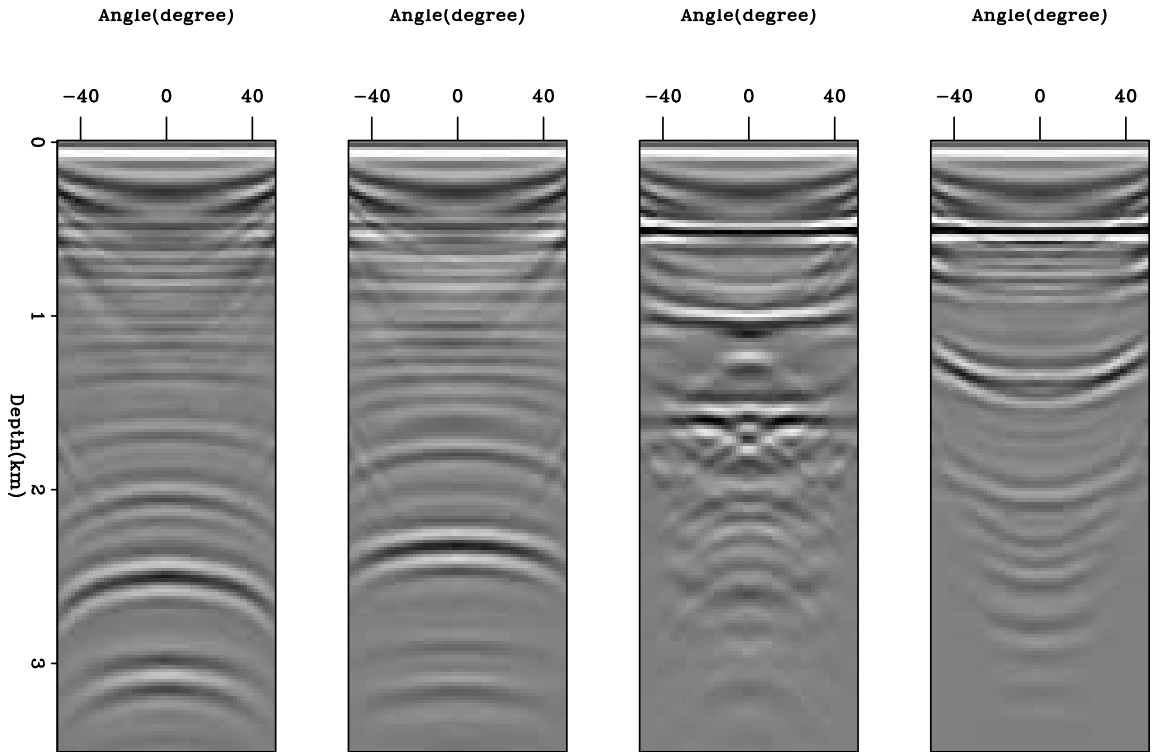

Figure 12: ADCIGs using gather-normalized residual and incorrect velocity. [**ER**]

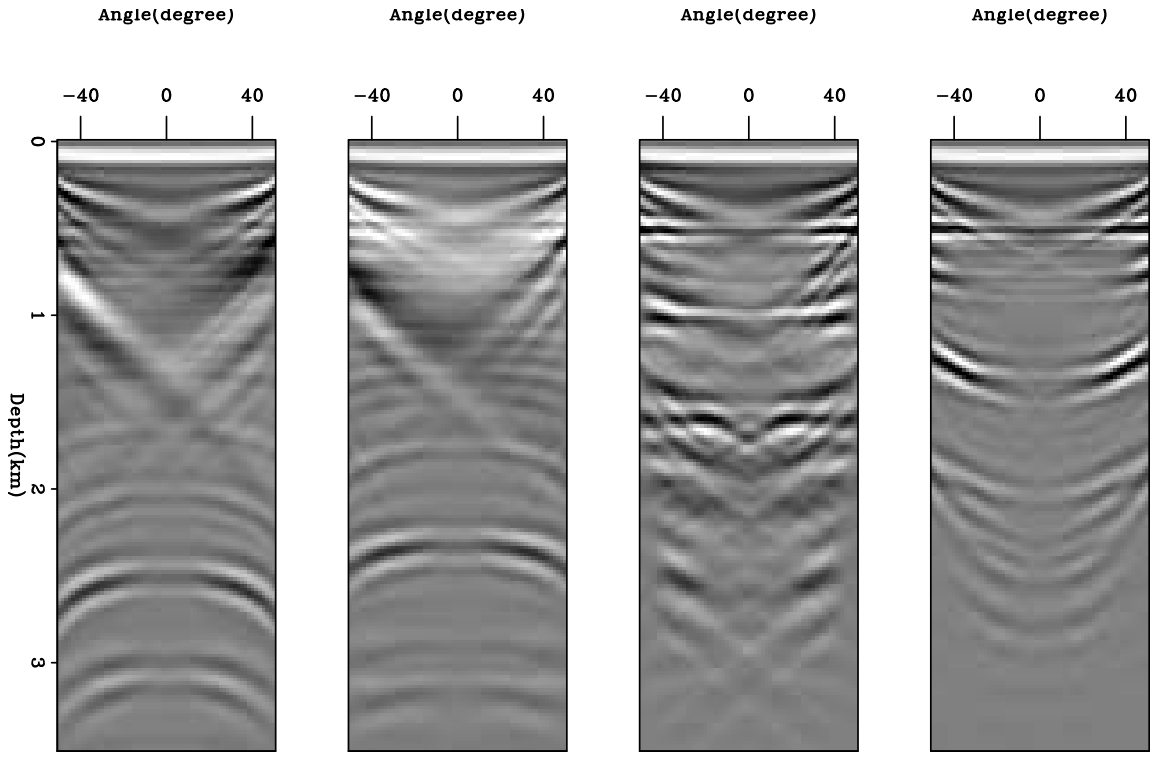

Figure 13: ADCIGs using trace-normalized residual and incorrect velocity. [**ER**]

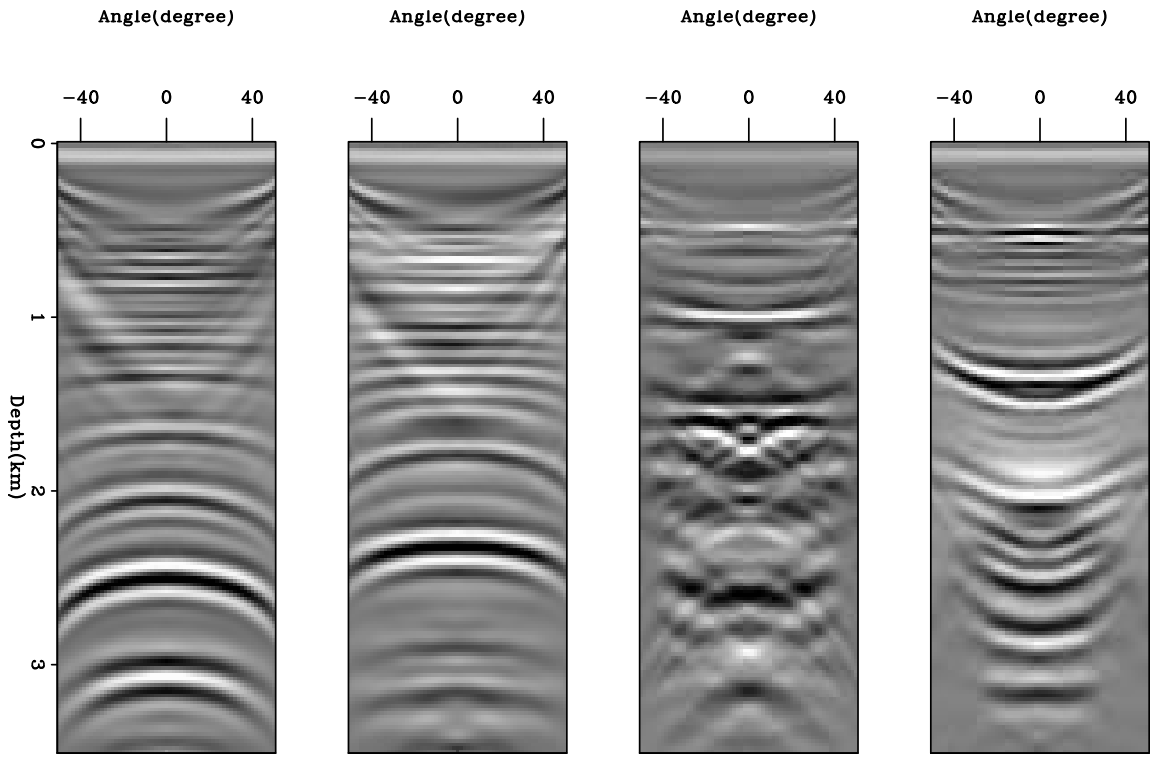

Figure 14: ADCIGs using window-normalized residual and incorrect velocity. [**ER**]

methods of normalization. This flexibility is capable of handing multiple events with varying amplitude-versus-offset behaviors. The preliminary results show a major improvement in the first gradient when compared to other methods of normalizing the objective function.

REFERENCES

Almomin, A. and B. Biondi, 2013, Tomographic full waveform inversion (TFWI) by successive linearizations and scale separations: SEP-Report, **149**, 51–58.

———, 2014a, Preconditioned tomographic full waveform inversion by wavelength continuation : SEP-Report, **152**, 11–18.

———, 2014b, Preconditioned tomographic full waveform inversion by wavelength continuation: SEG Expanded Abstracts, 944–948.

Shen, X., 2014, Early-arrival waveform inversion for near-surface velocity estimation: PhD thesis, Stanford University.