Effective AMO implementation in the log-stretch, frequency-wavenumber domain[a]

[a]Published in SEP Report, 110, 63-70 (2001)

Ioan Vlad and Biondo Biondi[1]

INTRODUCTION

Azimuth moveout (AMO), introduced by Biondi et al. (1998), is used as part of the styling goal (in conjunction with a derivative as a roughener) in Biondi and Vlad (2001). This paper describes the implementation of AMO for the above-stated purpose, with a historical background, proof, and discussion of pitfalls and practical steps.

THE AZIMUTH MOVEOUT

AMO is conceived as a cascade of forward and reverse dip moveout (DMO) operators. Thus, the accuracy and speed of the DMO operator used is highly important. Computing the DMO in the frequency domain is accurate and simple, but computationally expensive because the DMO operator is temporally nonstationary. The technique of logarithmic time-stretching, introduced by Bolondi et al. (1982) increases the computational efficiency because the DMO operator is stationary in the log-stretch domain, and Fast Fourier Transforms can be used instead of slow Discrete Fourier Transforms. Gardner (1991), Black et al. (1993) and Zhou et al. (1996) derived equivalent and accurate log-stretch, frequency-wavenumber DMO operators. The implementation of the AMO presented in this paper is based on the derivation and algorithm in Zhou et al. (1996).

THE LOG-STRETCH, FREQUENCY-WAVENUMBER AMO IN 3D

Starting from the parametric DMO relations of Black et al. (1993), Zhou et al. (1996) derives an expression for a DMO applicable on 2D NMO-ed data. In order to extend the expression to 3D, we only have to replace the product kh between the wavenumber and half offset with the dot product of the same quantities, which are vectors in the case of 3D data. In order to perform AMO from the offset $\vec{h}_1$ to the offset $\vec{h}_2$, we need to cascade one forward DMO from offset $\vec{h}_1$ to zero offset with a reverse DMO from zero offset to offset $\vec{h}_2$. Thus, applying log-stretch, frequency-wavenumber AMO on a 3D cube of data $P\left(t, m_x, m_y\right)|_{\vec{h}_1}$ in order to obtain $P\left(t, m_x, m_y\right)|_{\vec{h}_2}$ involves the following sequence of operations:

1. Apply log-stretch along the time axis on the $P\left(t, m_x, m_y\right)|_{\vec{h}_1}$ cube, with the formula:

$$\tau = \ln\left(\frac{t}{t_c}\right), \tag{1}$$

[1]**e-mail:** nick@sep.stanford.edu, biondo@sep.stanford.edu

where t_c is the minimum cutoff time introduced to avoid taking the logarithm of zero. All samples from times smaller than t_c are simply left untouched, the rest of the procedure will be applied to the cube $P\left(t > t_c, m_x, m_y\right)|_{\vec{h}_1}$.

2. 3D forward FFT of the $P\left(\tau, m_x, m_y\right)|_{\vec{h}_1}$ cube. The 3D forward Fourier Transform is defined as follows:

$$P\left(\Omega, k_x, k_y\right) = \int\int\int P\left(\tau, m_x, m_y\right) e^{i\left(\Omega\tau - k_x m_x - k_y m_y\right)} d\tau\, dm_x\, dm_y \qquad (2)$$

It can be seen that the sign of the transform along the τ-axis is opposite to that over the midpoint axes.

3. For each element of the cube, perform the AMO shift:

$$P\left(\Omega, k_x, k_y\right)|_{\vec{h}_2} = e^{i\left(\Phi_1 - \Phi_2\right)}\, P\left(\Omega, k_x, k_y\right)|_{\vec{h}_1}, \, where \qquad (3)$$

$$\Phi_j = \begin{cases} 0, \quad for\ \vec{k}\cdot\vec{h} = 0 \\ \vec{k}\cdot\vec{h}, \quad for\ \Omega = 0 \\ \frac{\Omega}{2}\left\{\sqrt{1+\left(\frac{2\,\vec{k}\cdot\vec{h}}{\Omega}\right)^2} - 1 - \ln\left[\frac{\sqrt{1+\left(\frac{2\,\vec{k}\cdot\vec{h}}{\Omega}\right)^2}+1}{2}\right]\right\} \quad otherwise \end{cases}, \qquad (4)$$

$$where\ \vec{k}\cdot\vec{h} = k_x h_x + k_y h_y \qquad (5)$$

and j can take the values 1 or 2. The frequency domain variables must have incorporated in their value a 2π constant (they are defined according to equation (2))

4. Do reverse 3D FFT in order to obtain the $P\left(\tau, m_x, m_y\right)|_{\vec{h}_2}$ cube.

5. Do reverse log stretch along the time axis and affix to the top of the cube the slices from times smaller than t_c. The final result is a $P\left(t, m_x, m_y\right)|_{\vec{h}_2}$ cube.

Figure 1 shows the impulse response of the above described AMO.

STRETCHING AND ALIASING

For the purpose of this discussion we define stretching of a single-dimension space as any transformation from one space to another that has the following property: at least an arbitrarily chosen sequence of two consecutive, equal in length, intervals in the input space is transformed into a sequence of two consecutive, *not* equal in length, intervals in the output space. Stretching an x-space to a y-space will be denoted as

$$y = f(x) \qquad (6)$$

Two obvious examples of stretching are

$$NMO:\ y = \sqrt{x^2+\alpha},\ and$$
$$log-stretch:\ y = \log\left(\frac{x}{\alpha}\right),$$

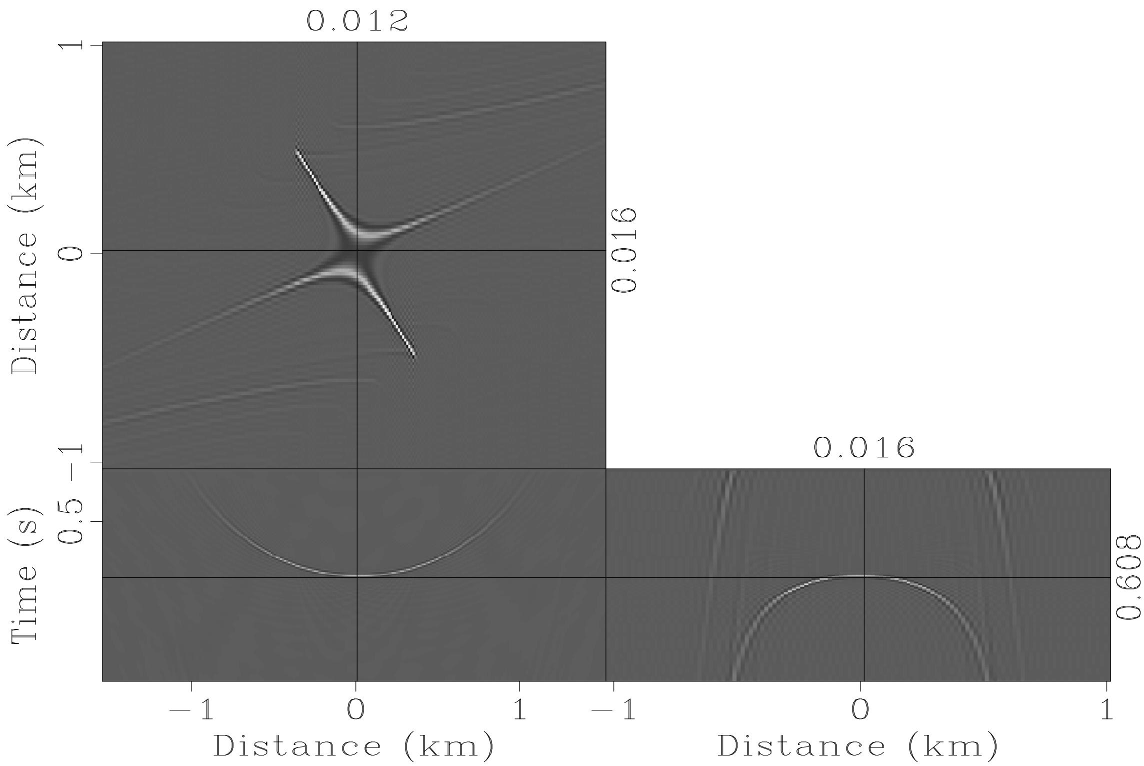

Figure 1: AMO impulse response

where α is a positive real number whose value does not matter for the purpose of this discussion. As it can be seen in Fig. 2, if we keep the same sampling rate ($\Delta y = \Delta x$), aliasing can occur when doing the reverse transformation, from x to y. In order to avoid aliasing, we need to compute $\Delta y_{\max}$, the largest accceptable sampling rate in the y domain. This can sometimes lead to a larger number of samples in the y domain, and thus to larger computational expense. This can be limited to some extent if the signal in the x-space has been bandpassed, as is often the case with seismic data, with the largest frequency present in the data ($f_{\max}$) smaller than the Nyquist frequency given by the sampling rate (f_{Ny}). Thus, we can replace in our calculations Δx with

$$\Delta x_{\max} = \frac{1}{2 f_{\max}},$$

which will result in a $\Delta y_{\max}$ larger than that computed using Δx, the sampling rate in the x space.

In order to compute $\Delta y_{\max}$, we will consider two points in the x space, as seen in Fig. 2, such as

$$x_b = x_a + \Delta x_{\max} \tag{7}$$

and y_a and y_b, the images of x_a and x_b in the y space. Thus,

$$\Delta y = y_b - y_a = f_{(x_a + \Delta x_{\max})} - f_{(x_a)}$$

The largest sampling rate in the y-space that will not result in aliasing is $\Delta y_{\max}$, the

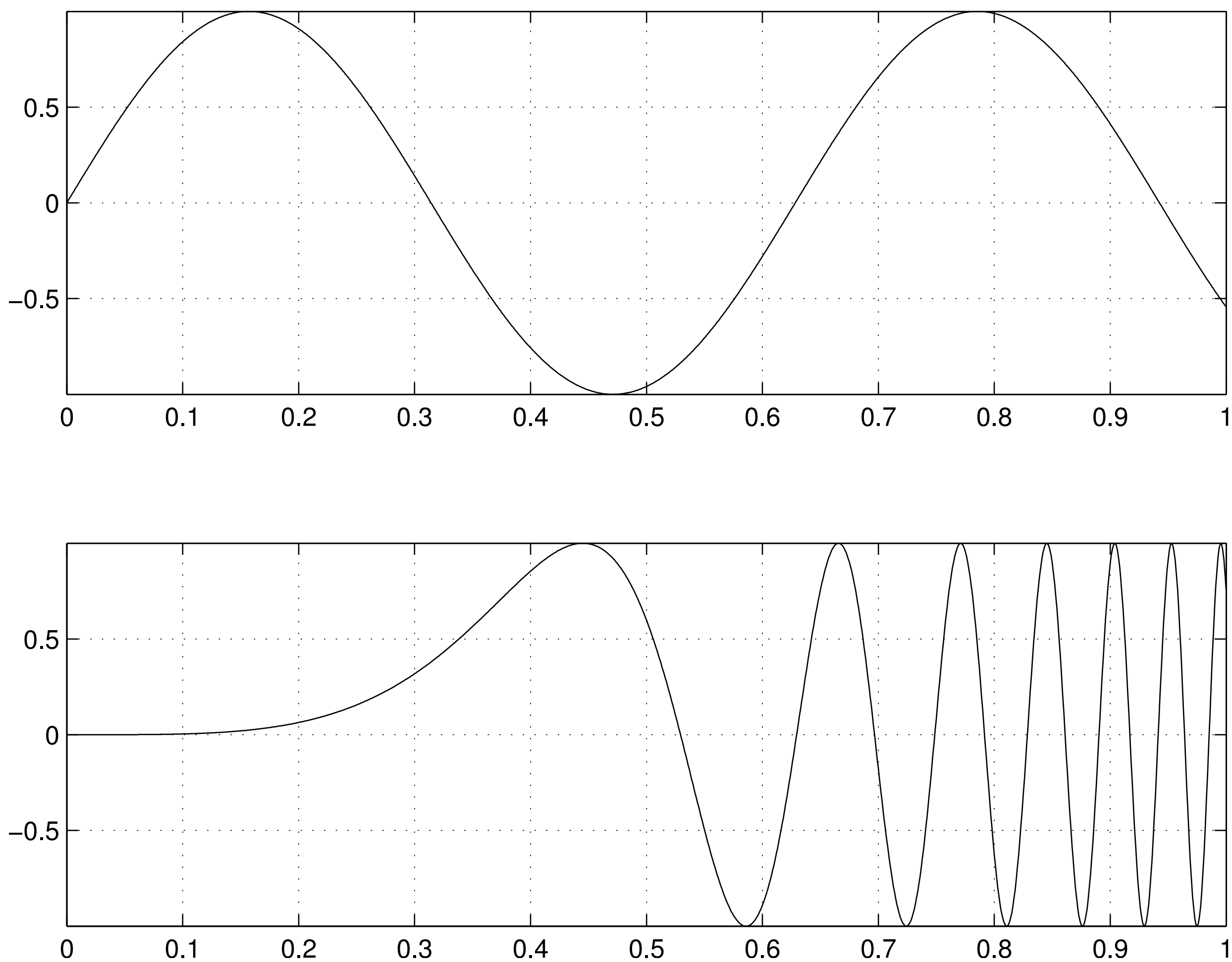

Figure 2: Illustration of how aliasing can occur while stretching: if the same sampling rate is used for the y-space (lower plot) as for the x-space (upper plot), serious aliasing will occur when transforming back to x-space. This will not happen if the sampling rate in the y-space is smaller than or equal to $\Delta y_{\max}$

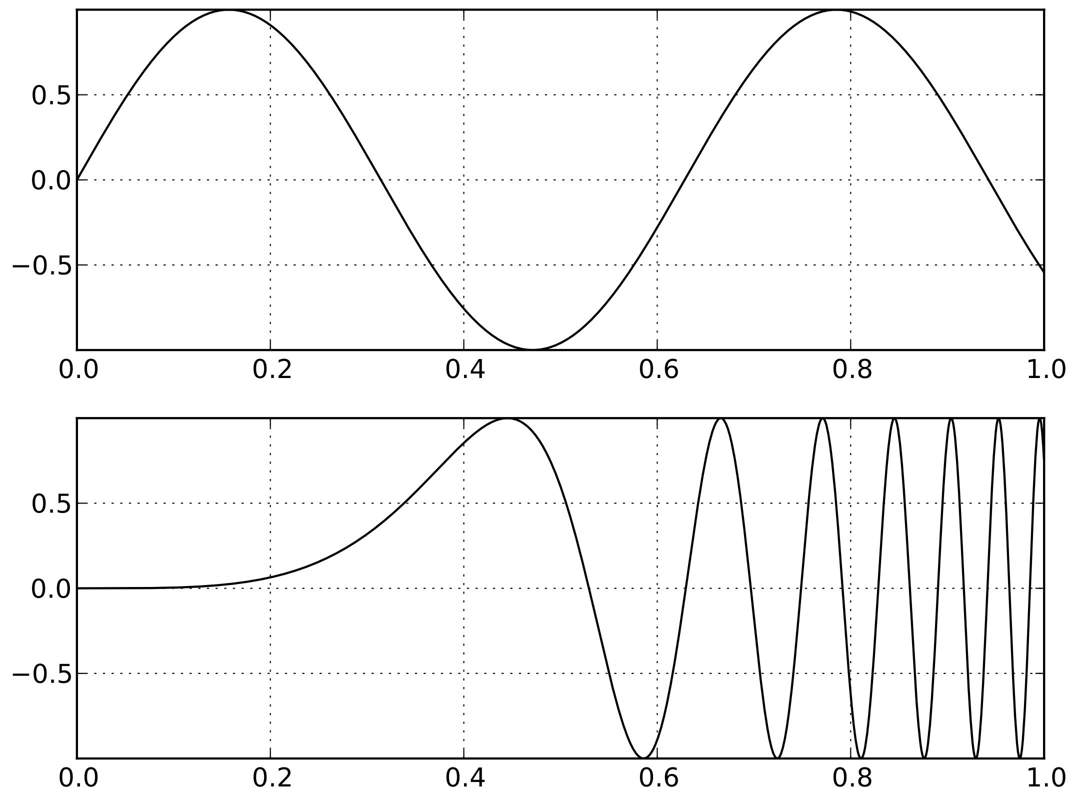

Figure 3: Illustration of how aliasing can occur while stretching: if the same sampling rate is used for the y-space (lower plot) as for the x-space (upper plot), serious aliasing will occur when transforming back to x-space. This will not happen if the sampling rate in the y-space is smaller than or equal to $\Delta y_{\max}$

minimum possible value of Δy. Suppose there is a value x_m that minimizes Δy. Then,

$$\Delta y_{\max} = \left[f_{(x+\Delta x_{\max})} - f_{(x)} \right] \Big|_{x_m}$$

In particular, in the case of log-stretch, given by equation (1), if t_m plays the role of x_m from the equation above, then

$$\Delta \tau_{\max} = \left[\log\left(\frac{t+\Delta t_{\max}}{t_c}\right) - \log\left(\frac{t}{t_c}\right) \right] \Big|_{t_m} = \log\left(1 + \frac{\Delta t_{\max}}{t_m}\right) \tag{8}$$

$\tau_{\max}$ will be minimum when t_m is as large as possible, thus minimizing the expression under the logarithm. How large can t_m get? Since the length of the seismic trace is limited to a value $t_{\max}$,

$$t_{\mathrm{m}} = t_{\max} - \Delta t_{\max}$$

because t_m is the equivalent of x_a from eq. (7) and Fig. 2. Thus, we get

$$\Delta \tau_{\max} = \log\left(\frac{t_{\max}}{t_{\max} - \Delta t_{\max}}\right) \tag{9}$$

F-K FILTERING

As it can be seen in Fig. 4, the impulse response of the AMO computed in the log-stretch, frequency-wavenumber domain has some artifacts: high amplitude, large saddle corners. Low temporal frequencies and high spatial slopes are also present. These artifacts can be eliminated easily using a f-k filter, which is described below.

Suppose we want to attenuate all spatial frequencies k that are larger than a certain threshold $k_{\max}$, where

$$\begin{array}{l} k = \sqrt{k_x^2 + k_y^2} \quad \textit{and} \\ k_{\max} = \frac{2|\omega|}{v_{\min}}, \end{array} \tag{10}$$

with ω, k_x and k_y being the coordinates in the frequency-wavenumber domain (without logstretch), and v being the minimum apparent velocity of the events that we want the filtered data cube to contain. Thus, the data cube will become:

$$P_{filtered}\left(\omega, k_x, k_y\right) = \left\{ \begin{array}{c} P\left(\omega, k_x, k_y\right) \quad if \quad k \leq k_{\max} \\ e^{-\varepsilon(k-k_{\max})^2} P\left(\omega, k_x, k_y\right) \quad if \quad k > k_{\max} \end{array} \right. \tag{11}$$

Too small an ε will result in an abrupt transition in the f-k domain, and thus ringing artifacts in the t-x domain. An ε which is too big will result in no visible filtering of the targeted artifacts. Moreover, ε depends on the choice of units and the number of samples for the m_x and m_y axes: since the exponential needs to be dimensionless, we have

$$\varepsilon = \frac{\varepsilon_0}{dk_x dk_y}$$

where

$$dk_x = \frac{1}{n_x d_x} \; \textit{and} \; dk_y = \frac{1}{n_y d_y}.$$

Thus, the final expression of ε is

$$\varepsilon = \varepsilon_0 n_x d_x n_y d_y, \tag{12}$$

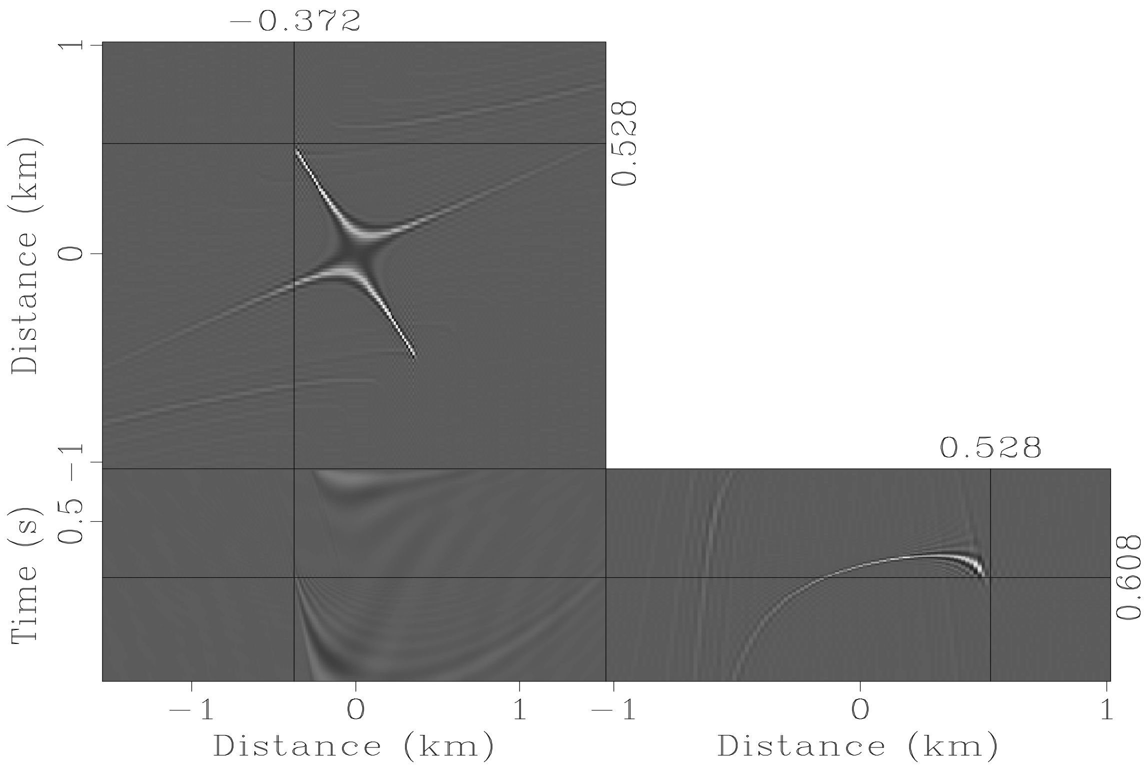

Figure 4: AMO impulse response artifacts

where ε_0 is a value that is hand-picked only once, and embedded in the code. This way, we will not have to change anything at all in the code or in the parameters in order to set ε_0, no matter what the units of the data cube may be.

The result of the filtering can be seen in Fig. 5: the slices through the cube are taken at exactly the same locations as those in Fig. 4, but now the artefacts are gone.

COST-CUTTING AVENUES

The largest computational savings come from the use of FFTs for AMO, instead of slow Fourier integration necessary in the absence of log-stretch. Standard means of minimizing the CPU time and the amount of memory used to compute the AMO have also been employed. They include computing the AMO shift for only half of the elements of the cube in the complex domain, since the Fourier transform F of a real function is Hermitian:

$$F(s) = F^*(-s) \tag{13}$$

(where s denotes the frequency domain variable and the star symbol denotes the complex conjugate). Another way of reducing computational expenses was through the use of RFFTW and FFTW type Fourier Transforms (Frigo and Johnson, 1998), adaptive to hardware architecture, and taking advantage of the property stated in (13). Also, the code was divided into subroutines in such a way that some quantities were not computed unnecessarily several times when AMO was applied to more than one cube of data. Finally, shared

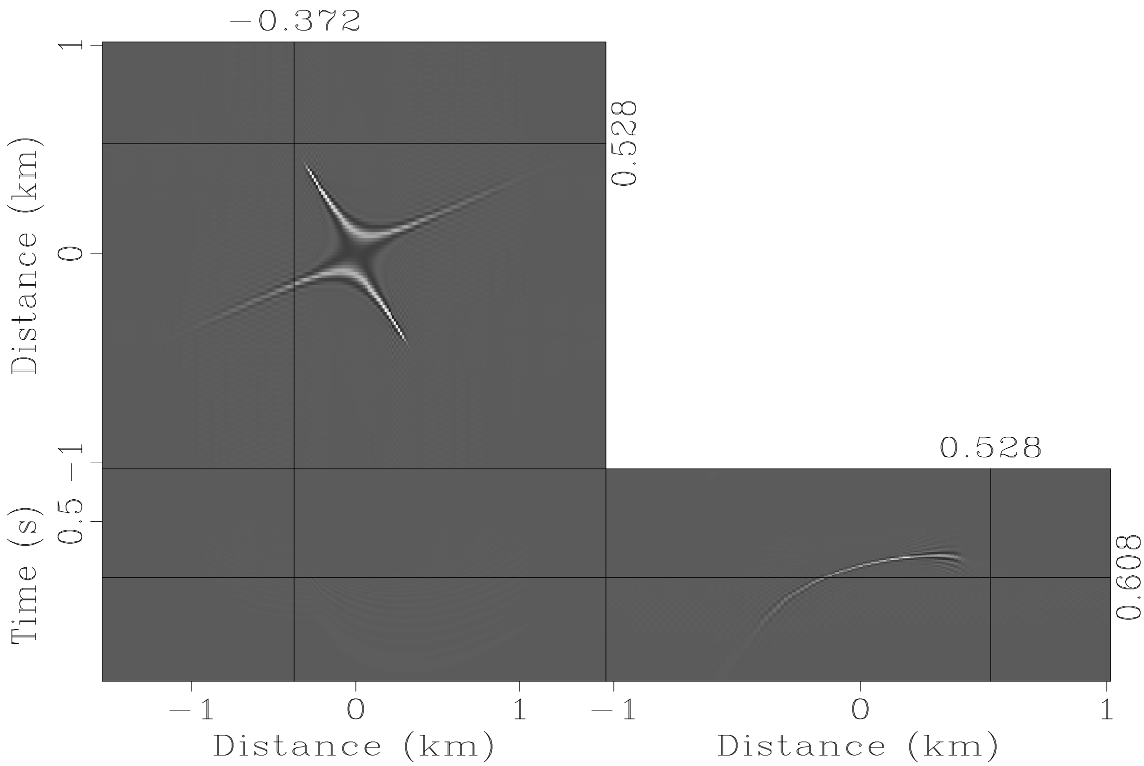

Figure 5: AMO impulse response after f-k filtering

memory parallelization with the OpenMP standard was applied to all the computationally intensive do loops in the code.

CONCLUSIONS

Azimuth moveout can be successfully implemented in the log-stretch, frequency-wavenumber domain. It is accurate, fast, and furthermore it does not have any characteristics that can result in coding difficulties.

REFERENCES

Biondi, B., S. Fomel, and N. Chemingui, 1998, Azimuth moveout for 3-D prestack imaging: Geophysics, **63**, 574–588.

Biondi, B., and I. Vlad, 2001, Amplitude preserving prestack imaging of irregularly sampled 3-D data, *in* SEP-110: Stanford Exploration Project, 1–18.

Black, J. L., K. L. Schleicher, and L. Zhang, 1993, True-amplitude imaging and dip moveout: Geophysics, **58**, 47–66.

Bolondi, G., E. Loinger, and F. Rocca, 1982, Offset continuation of seismic sections: Geophys. Prosp., **30**, 813–828.

Frigo, M., and S. G. Johnson, 1998, FFTW: An adaptive software architecture for the FFT: ICASSP Conference, Proceedings, 1381–1384.

Gardner, G. H. F., 1991, Interpolation, crossline migration and inline depth migration of 3-D marine surveys: SAL Annual Progress Review, **24**, 57–69.

Zhou, B., I. M. Mason, and S. A. Greenhalgh, 1996, An accurate formulation of log-stretch dip moveout in the frequency-wavenumber domain: Geophysics, **61**, 815–820.