Stanford Exploration Project, Report 75, November 17, 1997, pages 107–120

Short Note

Least-squares Kirchhoff migration

Steve Cole & Martin Karrenbach[1]

keywords: *least squares, slant stack, velocity*

INTRODUCTION

Slant stacks of finite-aperture data contain artifacts that are a result of the limited aperture. Resolution in the transform domain is also adversely affected by the finite aperture. Clement Kostov (Kostov, 1990), using least-squares inverse operators formulated by Beylkin (1987), developed least-squares slant stack transforms that compensate for the limited aperture, reducing artifacts and improving resolution. These least-squares transforms are best implemented in the frequency domain, where a separate, small least-squares problem can be constructed for each frequency.

A frequency-domain implementation is also possible for non-linear moveout trajectories, as long as the moveout correction is time-invariant. Kostov used a time-invariant hyperbolic transform to detect the signal from a drill bit. A time-invariant transform was appropriate in this case because the drill bit operates continuously over time, so a given moveout trajectory may be observed at any time.

For impulsive-source data, however, even in the constant-velocity case, a time-invariant transform is not appropriate. Several authors have found ways around this problem, for application to the problem of multiple suppression. Hampson (1986) noted that after NMO correction, residual moveout of multiples in a CMP gather is approximately time-invariant and parabolic. He used a time-invariant parabolic transform to separate multiples and reflections. Yilmaz (1989) proposed a t^2 stretching of the time axis as an alternative to Hampson's NMO correction. Kelamis and Chiburis (1992) used the t^2 stretch, and by partially stacking the data, regularized the recording geometry to reduce the number of least-squares problems to be solved. Foster and Mosher (1992) showed that a hyperbolic moveout trajectory was more appropriate for NMO-corrected CMP gathers than a parabolic one.

In all of these cases, a time-invariant transform could be used because, after NMO correction or t^2 stretching, multiples exhibit moveout that is time-invariant.

[1]**email:** not available

Kirchhoff migration, also a summation along hyperbolic trajectories, suffers from similar aperture effects. Some work has been done on limited aperture migration; Carrion *et al.* (1991) present an overview, and offer a wavefront set method that uses deterministic and stochastic criteria to compensate for the limited aperture.

In this paper, we propose applying the least-squares transform approach described above to the Kirchhoff migration case. As in the slant stack case, a least-squares Kirchoff migration algorithm should offer improved resolution.

Using a time-invariant transform is important, as it was in the slant stack case, in reducing the computational cost of the algorithm. Unfortunately, even in the constant velocity case, the summation trajectory for Kirchhoff migration is time-variant. There are several possible alternatives at this point. We could decide to work in the time domain, where we can use time-variant transform. This approach is taken by another paper in this report (Ji, 1992). Although the system of equations to be solved is larger in the time domain, this may be the best approach when accurate velocity information is available. A second approach, which we have not tried, would be to perform NMO correction, and then assume that any residual moveout of events is approximately parabolic or hyperbolic and time-invariant. A third approach, which is the subject of this paper, is to use time-invariant transforms, despite their drawbacks, and find a way to relate them to the time-variant transforms of a conventional migration.

We describe a version of Kirchhoff migration where time-invariant transforms are used. Input 2D data, a function of position x and time t, is transformed to a three-dimensional space where two of the axes are x and t. The various panels of the 3D cube correspond to different time-invariant moveout corrections. One can refer to this as the "hyperbola shape" axis. Imaging involves selecting, for each time sample, the result of summing along the hyperbola with the correct shape. For a constant velocity, we might use nt different hyperbolas, where nt is the number of points on the time axis, each appropriate for the given velocity and a different time zero. In this case, the third axis is the t_0 or "pseudo depth" axis. The migrated result is obtained by extracting the $t = t_0$ slice from the cube.

This is a costly procedure. The cost of computing the 3D migration cube is typically 40 times greater for the least-squares case than for a conventional Kirchhoff approach. We believe that it is worth considering, however, for three reasons:

- The time-invariant transform allows the various frequencies to be processed in parallel, reducing the effective cost on parallel computers.
- While straightforward imaging involves throwing away all but one slice of the 3D cube computed by the program, the rest of the cube actually contains information, similar to a migration velocity analysis, that allows us to determine whether the velocity used to migrate the data was correct.
- The added cost may be justified if this scheme allows us to reduce the aperture in migration of field data.

We begin our explanation of the least-squares scheme by a review of Kostov's least-squares slant-stack transforms, and show how a simple modification leads to a Kirchhoff migration scheme.

LEAST-SQUARES SLANT STACK TRANSFORMS

The conventional slant-stack transform takes us from the recorded data $d(x,t)$ to the slant stack domain $m(p,\tau)$:

$$Ld = m. \tag{1}$$

In the frequency domain, a separate system of equations can be constructed for each frequency. The matrix L contains time shifts, expressed as complex exponentials:

$$L \;=\; \begin{pmatrix} e^{j\omega p_1 x_1} & \dots & e^{j\omega p_1 x_{n_x}} \\ \vdots & \vdots & \vdots \\ e^{j\omega p_{n_p} x_1} & \dots & e^{j\omega p_{n_p} x_{n_x}} \end{pmatrix}. \tag{2}$$

The conjugate operator L^H brings us back to time and space:

$$L^H m = d. \tag{3}$$

The conjugate operator is the more straightforward of the two. If we have a point in (p,τ) space, the operator L^H does a good job of turning it into a line in (x,t) space, regardless of the number of traces in x; there is no aperture effect. The forward transform, however, is affected by the aperture. If we transform a single dipping event in (x,t) to (p,τ) there will be artifacts and a loss of resolution, caused by the limited aperture.

We would do better to transform from (x,t) to (p,τ) using the inverse of the matrix L^H:

$$m = (L^H)^{-1} d. \tag{4}$$

However, the matrix L^H is typically not square; the problem is either overdetermined or underdetermined. In this case, we can find the best least-squares solution by multiplying both sides of the conjugate transform by L:

$$LL^H m = Ld, \tag{5}$$

giving:

$$m = \left(LL^H\right)^{-1} Ld. \tag{6}$$

The matrix $\left(LL^H\right)^{-1} L$ is the least-squares inverse of L^H.

The least-squares slant stack is summarized in Figure **??**.

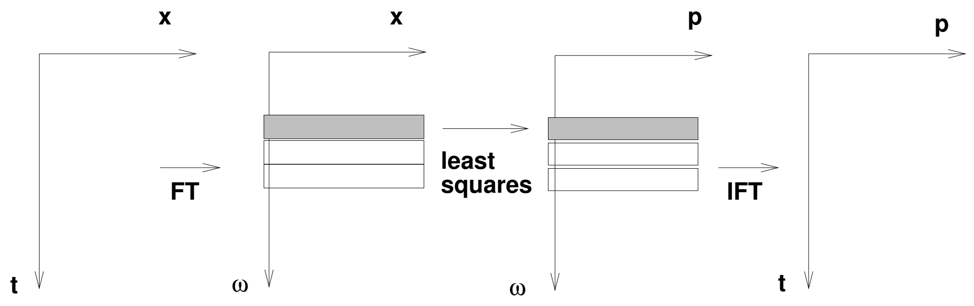

Figure 1: Schematic of least-squares slant stack. A small least-squares problem is solved for each frequency. This is less costly than working in the time domain, where one very large system of equations would need to be solved. The time-invariant moveout correction of slant stacks makes this separation possible. steve1-slant [NR]

LEAST-SQUARES KIRCHHOFF MIGRATION

To do least-squares Kirchhoff migration, we either have to work in the time domain, as is done by Ji elsewhere in this report, or use time-invariant transforms, despite their apparent inappropriateness.

A Kirchhoff migration analog to least-squares slant stacking is summarized in Figure **??**. Since we are using time-invariant transforms, a separate least-squares problem is set up for each frequency. In the slant stack case, this problem involved transformation to a one-dimensional space (ray parameter p), since only one parameter is needed to describe a linear moveout trajectory, its slope.

In the migration case, two parameters are needed to describe a migration hyperbola: the position beneath which the apex of the hyperbola lies (x), and another parameter that describes the curvature of the hyperbola. If we fix velocity, this second parameter can be called t_0. For a given constant velocity, the hyperbola will have greater curvature for an early t_0, and less for a later one. The transform space, then, contains two parameters (x and t_0) instead of one in the slant stack case.

This means that the least-squares problem is likely to be larger than in the slant-stack case, and is typically underdetermined. But in principle there is nothing wrong with the transform space having an extra dimension.

SOLVERS

Clement Kostov observed that in 2-D, even if the sampling in offset is irregular, the matrix LL^H is Toeplitz, as long as the sampling in ray parameter is uniform. For a Toeplitz matrix, an efficient means of solving the system, using Levinson recursion, exists with a cost proportional to the square of the length of the unknown vector.

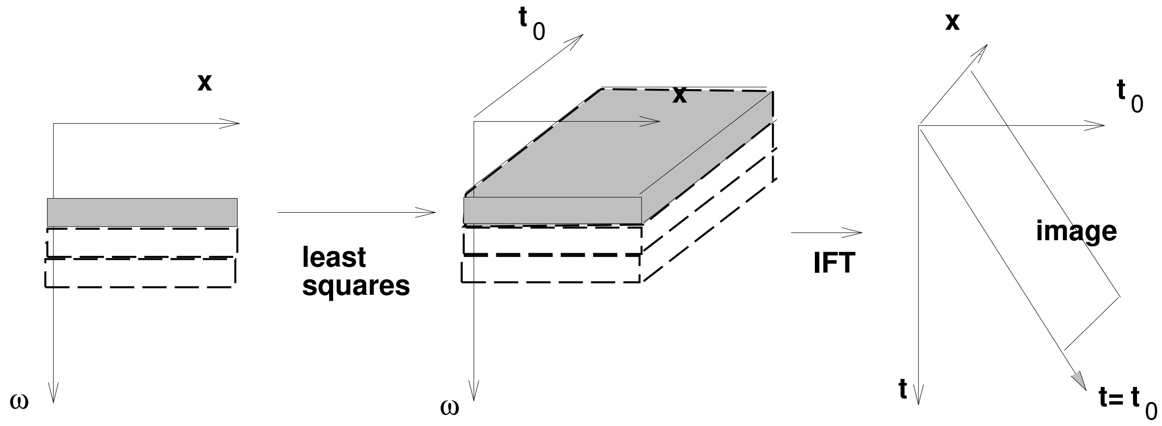

Figure 2: Overview of least-squares migration. An underdetermined problem is solved for each frequency to find sources, as a function of x, the position of the hyperbola apex, and t0, the curvature. The image is contained in a single plane of the inverse Fourier transformed cube, but other planes contain information about focusing at other velocities. steve1-mig [NR]

In the migration case, because of the extra axis in the transform domain, the matrix is typically not Toeplitz. Though it is often block Toeplitz, and solution methods exist that take advantage of that form, we have elected to use a simple conjugate gradient scheme to solve the least-squares problem in these initial tests.

SYNTHETIC TESTS

The impulse response of the migration algorithm is shown in Figure 3. To test the effect of a limited aperture, we generated a simple synthetic zero-offset section containing a single hyperbolic event, as shown in Figure 4. A conventional Kirchhoff migration, using all the traces, is shown in Figure 5, and a migration where the aperture is limited to the nearest 15 traces (the section contains 75 traces) is shown in Figure 6. The resolution in the limited aperture migration is not as good. The same two plots are shown for the least-squares case in Figures 7 and 8. The limited aperture result has improved resolution compared to the non least-squares case.

Some noise appears at the top of the least-squares sections in Figures 7 and 8. These first few samples correspond to the earliest t_0 values (hyperbolas with the largest amount of moveout). The noise does not appear to be due to frequency-domain wraparound, or aliasing, and damping the least-squares method doesn't seem to help. The noise is sufficiently strong, particularly away from the single image plane shown in the figures, that it is a cause for concern. We hope to come up with a solution in the near future, but weren't able to do so in time for this report. This problem doesn't take away from the main result, as shown in the figures, that the focusing of migration is less affected by the limited aperture in the least-squares case.

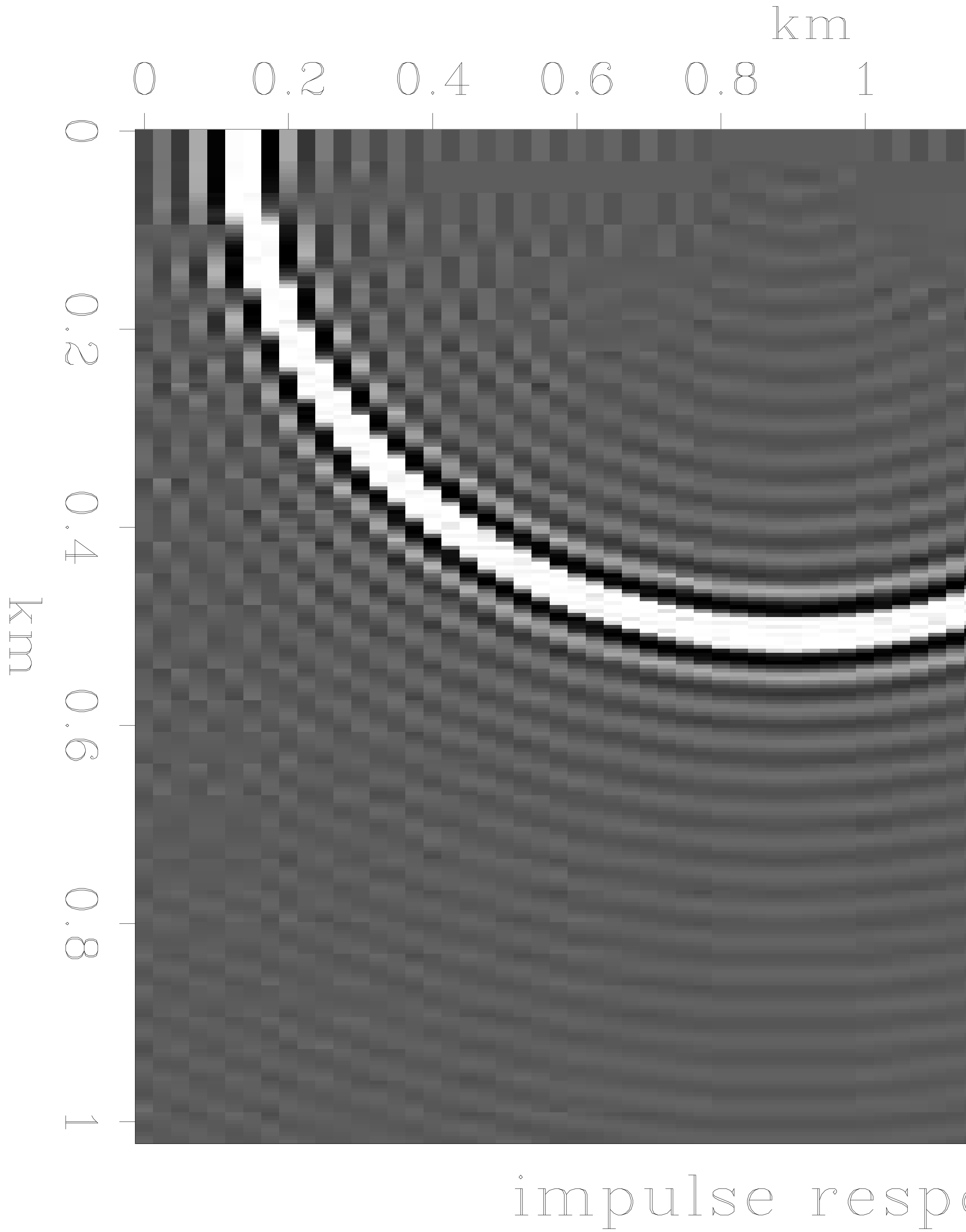

Figure 3: Impulse response of migration algorithm. This is a single slice from the 3D cube, taken along the $t = t_0$ plane. steve1-impulse [NR]

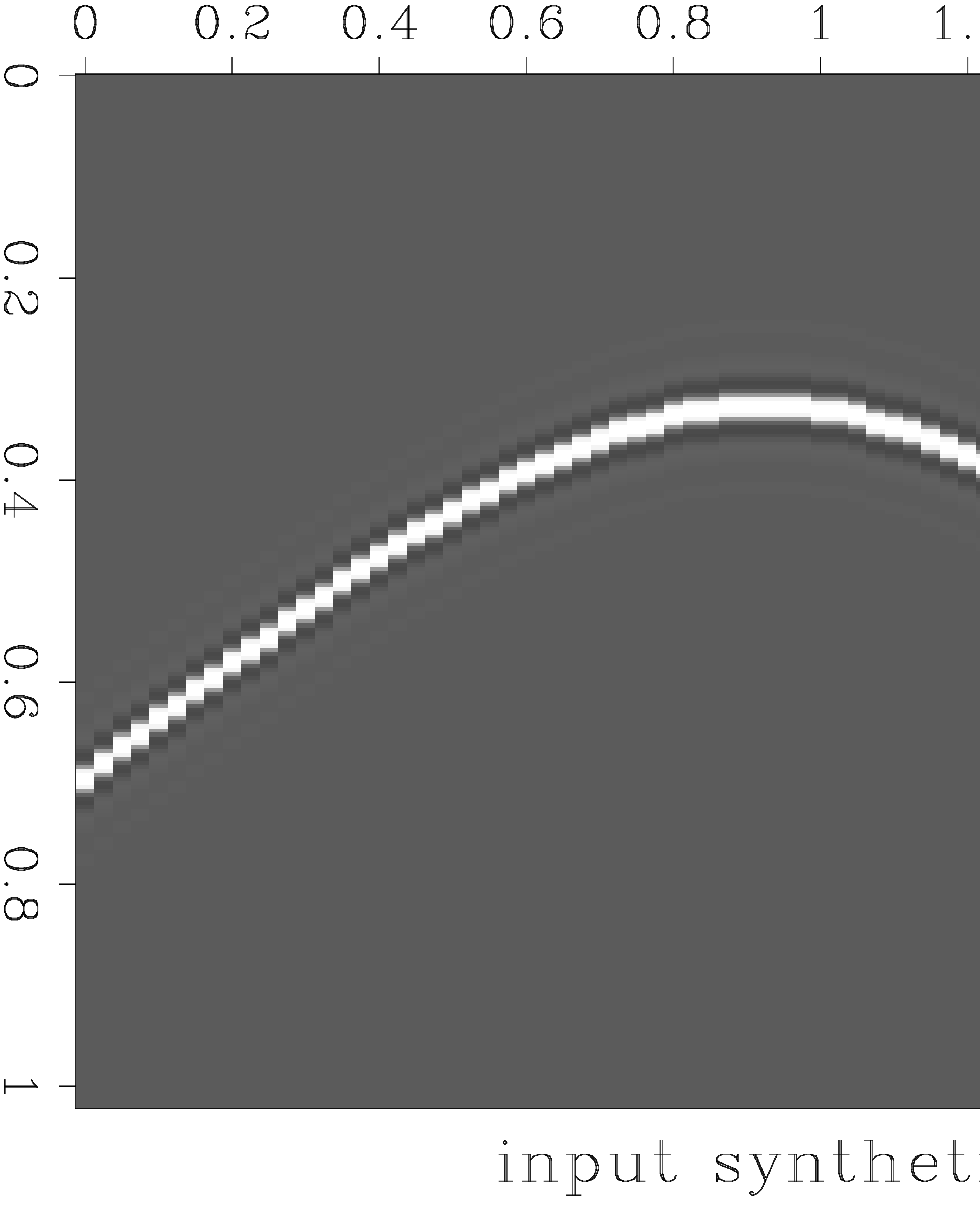

Figure 4: Synthetic dataset used to test the migration scheme. Single point scatterer, constant velocity of 3 km/sec. 75 traces with a trace spacing of 25 meters. steve1-model [ER]

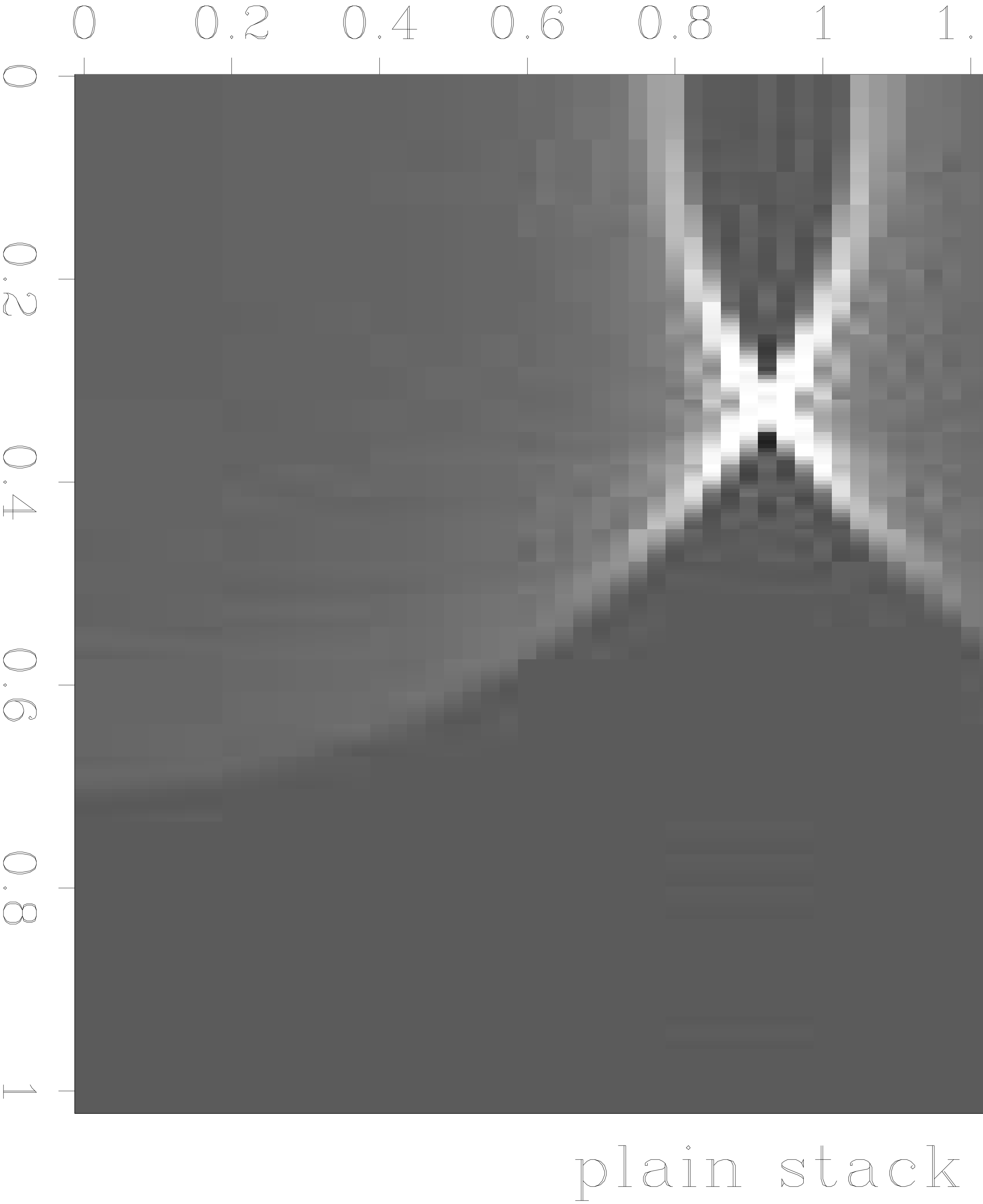

Figure 5: Conventional Kirchhoff migration of synthetic dataset. Single slice from 3D cube at $t = t_0$. steve1-stack [NR]

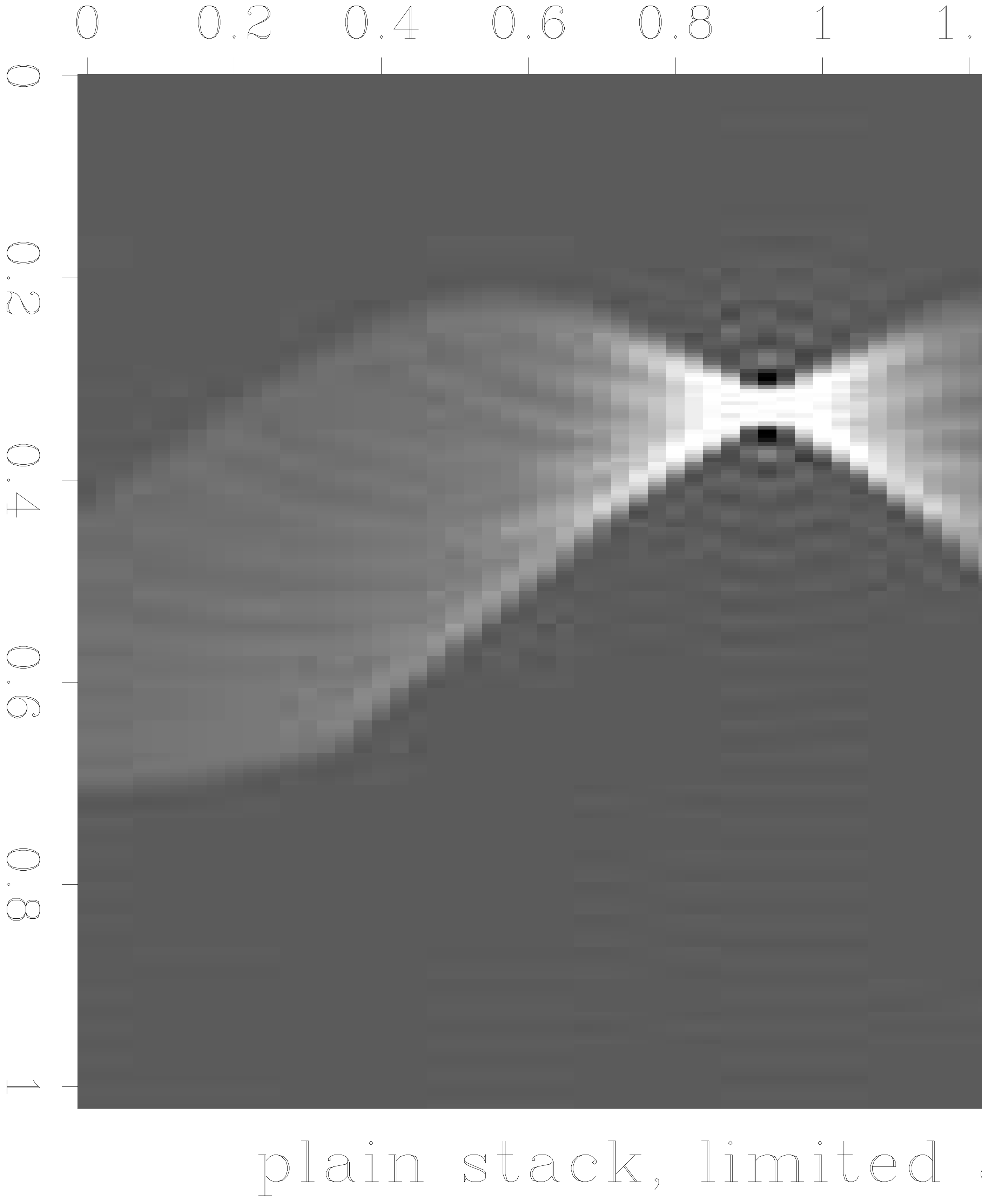

Figure 6: Conventional Kirchhoff migration of synthetic dataset, but migration aperture has been limited to 15 of 75 traces. steve1-stacklim [NR]

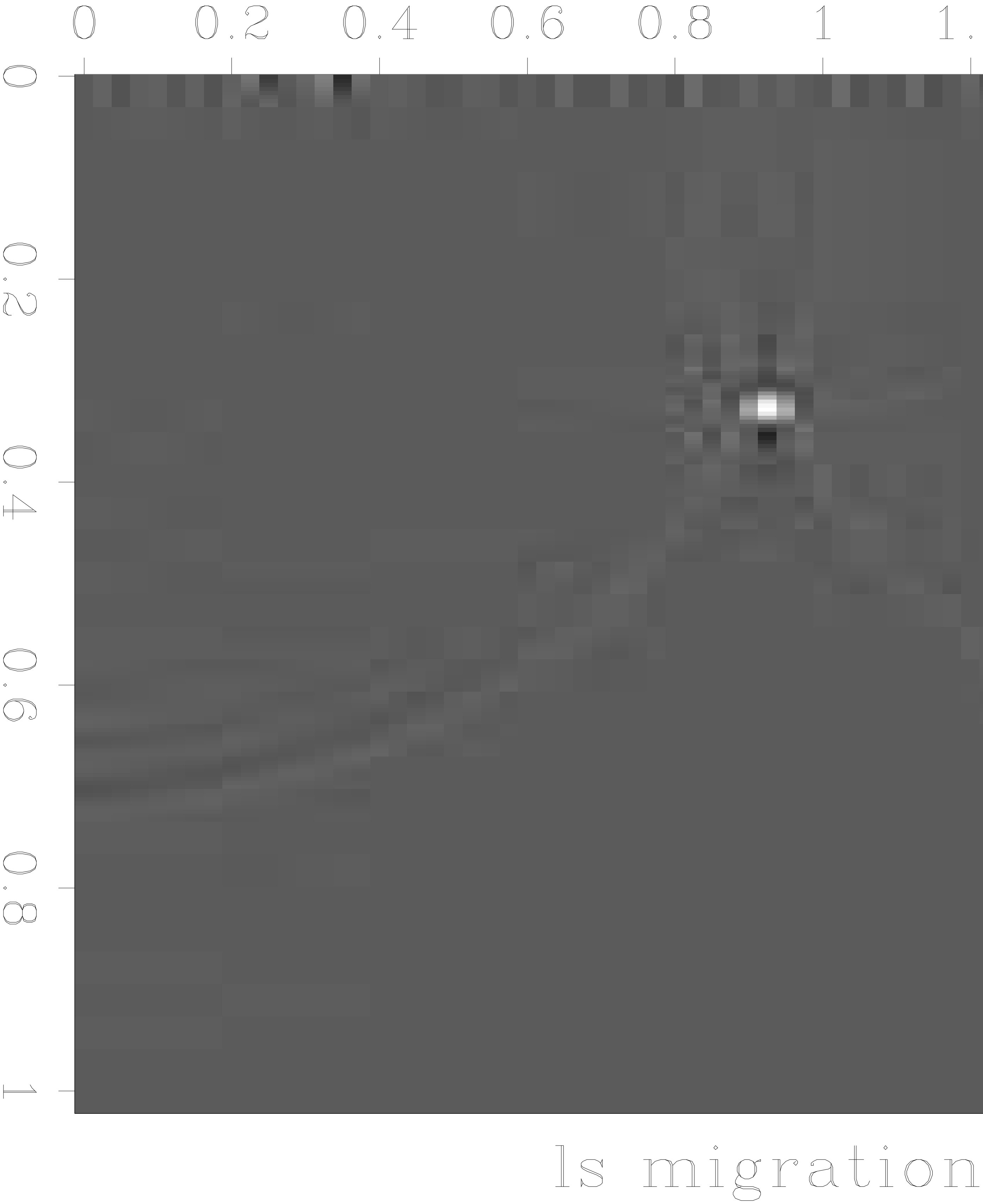

Figure 7: Least-squares Kirchhoff migration. Gives a better focus because effect of aperture has been removed. steve1-lsmig [NR]

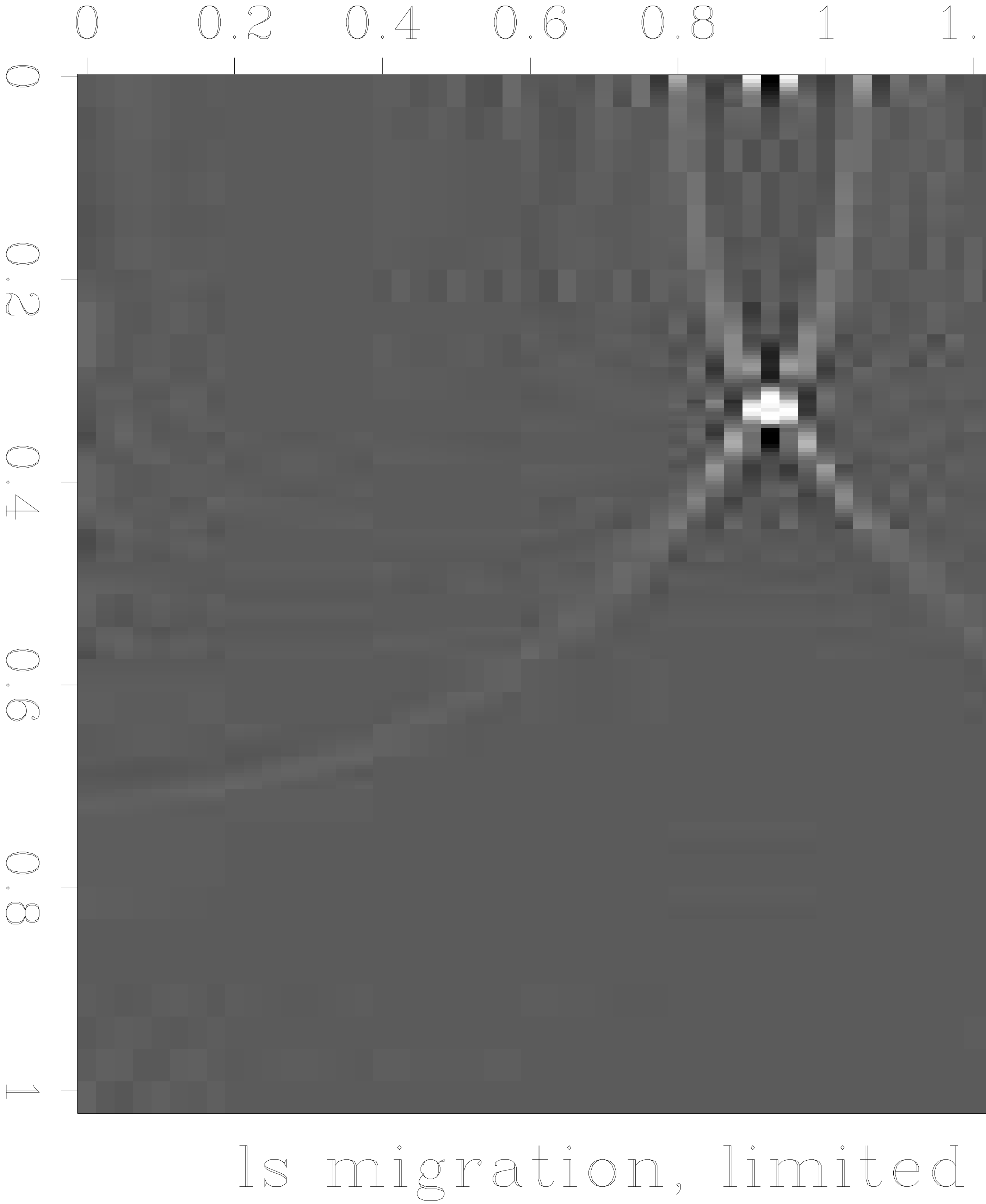

Figure 8: Least-squares Kirchhoff migration, aperture limited to 15 of 75 traces. steve1-lsmiglim [NR]

The cost of this method is considerable. The conjugate gradient algorithm runs for an average of about twenty iterations on each frequency. This means multiplication by the L or L^H matrix an average of forty times, versus a single multiplication for the non least-squares case. The aperture compensation provided by least-squares must be considerable if this method is to be worth the extra cost. If least-squares allows us to migrate using fewer traces, however, then the system of equations will be smaller, and the extra cost will not be so large. Another way to reduce the cost is to consider a target oriented scheme, where only those hyperbola shapes appropriate for the target zone are used for the third axis of the migration cube. This would again make the least-squares problems smaller, and reduce the cost.

Migration velocity analysis

As mentioned earlier, least-squares Kirchhoff migration involves using only a single slice out of a much larger transform domain cube. If the rest of the cube contained no interesting information, then the extra computations of the least-squares method would not be justified. However, the rest of the transform domain cube does serve a purpose. Figure 9 shows a single slice taken from two such cubes. This is for a non least-squares example. The x coordinate is fixed at the location of the single point scatterer; the axes are t and t_0. The cube has been shifted so that the single slice that corresponds to the migrated image lies along the top of the cube. On the left is the result of migrating with the correct constant velocity; we see energy focusing near the top of the plot. On the right is the result of migrating with too high a velocity (3.5 km/sec instead of 3 km/sec). The focusing of energy away from the $t = t_0$ plane tells us that an incorrect migration velocity was used. The cube contains important information, similar to a migration velocity analysis, away from the image plane. This gives us reason to think that the added cost of the least-squares method might be worthwhile.

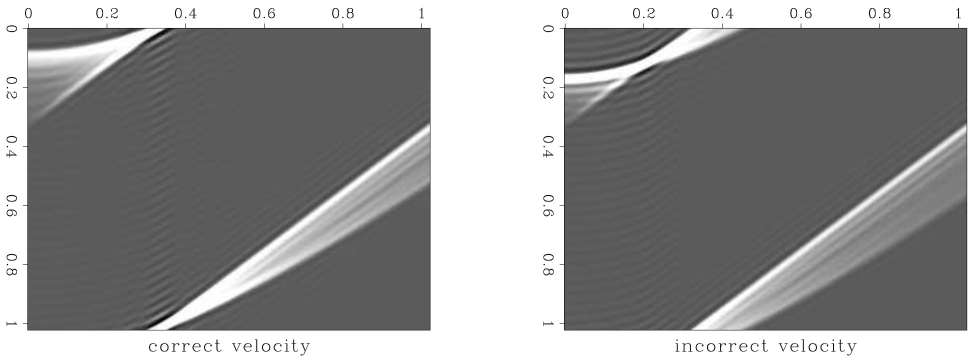

Figure 9: Left, migration with correct velocity. Slice for a single x coordinate, at the scatterer location. Focus is on the $t = t_0$ line, which has been shifted to lie along the top of the plot. Right, migration with incorrect velocity gives focus that is away from the $t = t_0$ line. steve1-focus [NR]

CONCLUSIONS

While least-squares Kirchhoff migration means doing more computations than in a conventional migration, the extra work serves two purposes - it compensates for the limited aperture of the migration, and it gives a migration velocity analysis cube as a by-product. Further work is neeeded to show whether these benefits justify the added cost, which is considerable.

REFERENCES

Beylkin, G., 1987, Discrete Radon transform: IEEE Trans. Acoust., Speech, and Signal Proc., ASSP–**35**, No. 2, 162-172.

Carrion, P.M., Sato, H.K., and Buono, A.V.D., 1991, Wavefront sets analysis of limited aperture migration sections: Geophysics, **56**, 778–784.

Foster, D.J., and Mosher, C.C., 1992, Suppression of multiple reflections using the Radon transform: Geophysics, **57**, 386–395.

Hampson, D., 1987, The discrete Radon transform: a new tool for image enhancement and noise suppression: 57th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts.

Ji, J., 1992, Least squares imaging, datuming and interpolation using the wave equation: SEP–**75**.

Kelamis, P.G., and Chiburis, E.F., 1992, Land data examples of Radon multiple suppression: First Break, **10**, 275–280.

Kostov, C., 1990, Multichannel seismic experiment with a drill-bit source: SEP–**63**.

Yilmaz, O., 1989, Velocity-stack processing: Geophysical Prospecting, **37**, 357–382.