Report to Cray Research, Inc.

Split-step Fourier Modelling and Migration

Paul L. Stoffa

Summary

The split-step Fourier method was evaluated as both a modelling and a migration algorithm for post-stack seismic data in two and three dimensions. This algorithm works by first applying a global phase shift in the ω-kx-ky domain based on an average or reference velocity in the current depth interval. This global phase shift is followed by the application of a local phase shift in the ω-x-y domain that accommodates lateral x-y velocity variations. These 2 phase shifts are applied for every frequency at each depth that is being imaged or modelled. A theoretical study of this method indicates that the method is exact for constant velocity depth intervals. If lateral, x-y, velocity variations are present, the method reduces naturally to a 15 degree approximation, see Appendix I. In addition to the theoretical study, numerical comparisons were made with other migration methods. In particular, several comparisons were made with finite difference reverse time migration, using operators that were 4th order in space and 2nd order in time. The most time consuming part of the split-step Fourier algorithm was identified and a series of refinements performed. Basically, the two-dimensional discrete Fourier transform, which is required twice per frequency per depth interval, was changed from generic Fortran, to the Boeing library routines and then finally to a routine provided by M. Edwards of Cray Research, Inc. A comparison of the performance on another supercomputer, the NEC SX-2 at the Houston Area Research Center, was also made.

Discussion of Results

Figure 1 shows a synthetic seismic section provided by Mobil Research for a complicated subsurface structure that includes a salt dome, see Figure 2. The seismic data were generated using a finite difference algorithm. Figure 3 shows the result of migration using Gazdag's phase shift migration method. It is obvious that the salt dome and its flanks are not correctly imaged because this method cannot accommodate lateral velocity variations. Figure 4 shows the results obtained with the split-step Fourier method. Notice in particular the image of the flanks of the salt dome and the reflection at the base of the salt. Figure 5 is a detailed comparison which includes the superposition of "delta function" reflection coefficients at the changes in the velocity field to indicate the correct image position. The split-step Fourier migration (left) produced a very accurate image while the phase shift method suffers the most in the zones which would be the most significant in any exploration objective, i.e., the flanks and base of the salt.

Figure 6 is a reverse time finite difference migration of the same data. In this example, considerable grid dispersion exists, which degrades the resulting image. In this example, the slowness (reciprocal velocity) field was first smoothed to minimize grid dispersion. This helped to reduce the dispersion, but obviously did not eliminate it. Further smoothing of the slowness field is possible, but the accuracy of the imaging would decrease. In most cases tested, the finite difference migration was initially faster than split-step Fourier migration, particularly for 2D models. But, if we include the time for all the necessary premigration processing required by the finite difference method the total required CPU time becomes nearly the same. (The additional preprocessing for finite difference migration includes: resampling by a factor of 4 in time and at least a factor of 2 in space to decrease the temporal and spatial sampling intervals and then smoothing the slowness field.)

Figure 7 is a real data stacked CDP marine section and Figure 8 is the migrated result using the split-step Fourier method. Note in particular the feature near the center of the section. Figure 9 is a detail of the stacked input data and Figure 10 the corresponding split-step Fourier migrated result. Little if any frequency dispersion is observed. Figure 11 shows the same result using reverse time finite difference migration. The image is similar, but significant grid dispersion is present and the overall frequency content is lower. These problems can be alleviated in 2D applications by decreasing the sampling interval in space and in time (even further). This will increase the cost of the finite difference migration proportionately and may require an out of core solution which would further increase the length of the computation.

The French model was used to test the 3D split-step migration. The data were generated using the Sierra modelling software to synthesize a zero offset seismic section for a true 3D subsurface model. The model results are ray theoretical and do not include true diffraction effects. Figures 12, 13, 14, and 15 show seismic lines before (upper) and after (lower) the split-step Fourier 3D migration. This migration uses the entire 3D volume simultaneously, so that both inline and crossline events are migrated correctly. This is in contrast to currently popular 2D by 2D methods which are poor approximations in complex geologic situations. In Figure 12 note in particular that the deepest event at about 3700m is nearly horizontal indicating that the overlying lateral velocity variations have correctly been taken into account. This is also evident in Figure 13, with only minor distortion occurring under the intersection of the dome and the horizontal event. Figures 16 and 17 show depth slices from the migrated volume. On the left are theoretical reflection coefficients generated by taking first differences of the velocity function. On the right is the split-step Fourier migration result. In both cases the spatial positioning of the final image is quite good. An edge effect can also been observed, see arrow, due to the limited number of lines (8) in the absorbing boundary.

The French data were also used as the basis for the timing study to determine the cost of the algorithm. In the 2D case, 1 line of the 64 available was migrated. A small, 8 line 3D case was also timed and a 32 line subvolume. Since the computation time is directly proportional to the (x by y) computation grid size, the number of depth-steps migrated and the number for frequencies per depth-step, the correct timing number for comparison purposes is the time per frequency per depth-step per wave number. Since the majority of the time is taken by the 1D complex-complex FFT (2D migration) or in the 2D complex-complex FFT (3D migration) and the phase shifting, we should be able to calibrate any combination of depths, frequencies, CDP's and lines by a knowledge of this time.

Table 1 lists measured times for several 2D complex fast Fourier transforms: Generic Fortran, The Boeing Library, and M. Edward's routine on the University of Texas Cray X-MP. Also included are several times for the NEC SX-2 using the routines provided by NEC. (The NEC routines use separate real and imaginary vectors rather than standard complex data organization. Therefore, the data were reorganized before and after each FFT via Fortran so as not to change the source code. The times listed include the packing and unpacking operation.)

Figure 18 summarizes one aspect of the timing study on the Cray. In this case, TFFT is the total time spent doing complex-to-complex Fourier transforms. Two are required per frequency per depth step. TDIF is the total time per frequency per depth less the total fast Fourier transform

time. We can see that TDIF is nearly the same in the 8 and 32 line case for M. Edwards and Boeing, indicating that the total non FFT work being accomplished is nearly the same. The only interesting difference occurs in the pure Fortran case, where no Cray optimized scaling or FFT routines are employed. In this case, the fast Fourier transform time dominates the rest of the computation time and is significantly greater than in the other two cases. In the case of the M. Edwards FFT, the FFT time has been reduced significantly both in the absolute sense and relative to the other computations, i.e., complex phase shifts, scalings, and summations. Further improvement can be made to reduce these other times thereby, improving the overall performance.

Figure 19 simply compares the total time per depth per frequency, (i.e., TFFT plus TDIF from Figure 18) after dividing by the number of wave numbers in the x and y directions. This time represents the total time per point on the computation grid. Also included is the same time for the NEC SX-2, which was timed for 1, 8 and 64 lines. In this plot, the times become nearly equal for each type of 2D FFT. That is, for the time required by the the Fortran FFT's, the time per point is nearly the same. The same is true for the Boeing FFT, M. Edwards FFT and the NEC SX-2 times. The NEC SX-2 times are significantly less than even the M. Edwards case. I would expect a 25% to 20% time reduction of the M. Edwards case by reducing the non FFT time, TDIF. This could be accomplished by further additional recoding.

Using Figure 20, it is possible to predict the time required for a specific computation grid. Multiplying the time per grid point by the number of spatial positions (x by y), the number of frequencies and the number of depth steps will approximate the total time for the migration. The additional time for initial data preparation and a final matrix transpose at the end of each job are insignificant compared to this computation time.

TABLE I

Time in msecs for 2D FFTS Complex - Complex

	32	64	128	256	512
32	5.618G 1.064B .475M .695N	11.292 2.148 .967 -	25.717 4.492 1.940 -	51.889 9.410 4.294 -	115.179 19.174 9.112 -
64	11.207 2.300 .964 1.144	22.556 4.643 1.880 -	51.436 9.635 3.891 -	103.807 19.998 8.199 -	231.870 41.466 17.564 -
128	25.506 5.070 2.032 2.001	51.331 10.199 3.927 -	115.286 21.147 7.779 -	232.560 43.777 18.180 -	516.998 90.414 34.644 -
256	51.186 11.179 4.199 2.921	102.695 22.135 8.273 -	231.474 46.568 16.927 -	466.042 96.086 35.709 -	1033.499 198.875 80.969 -
512	115.139 24.468 9.301 5.148	229.657 48.992 18.694 -	511.108 101.215 34.400 -	1034.975 205.730 75.353 -	2275.709 424.262 162.627 -

G = Generic Fortran

B = Boeing Library

M = M. Edwards

N = NEC SX-2 including Fortran complex pack and unpack

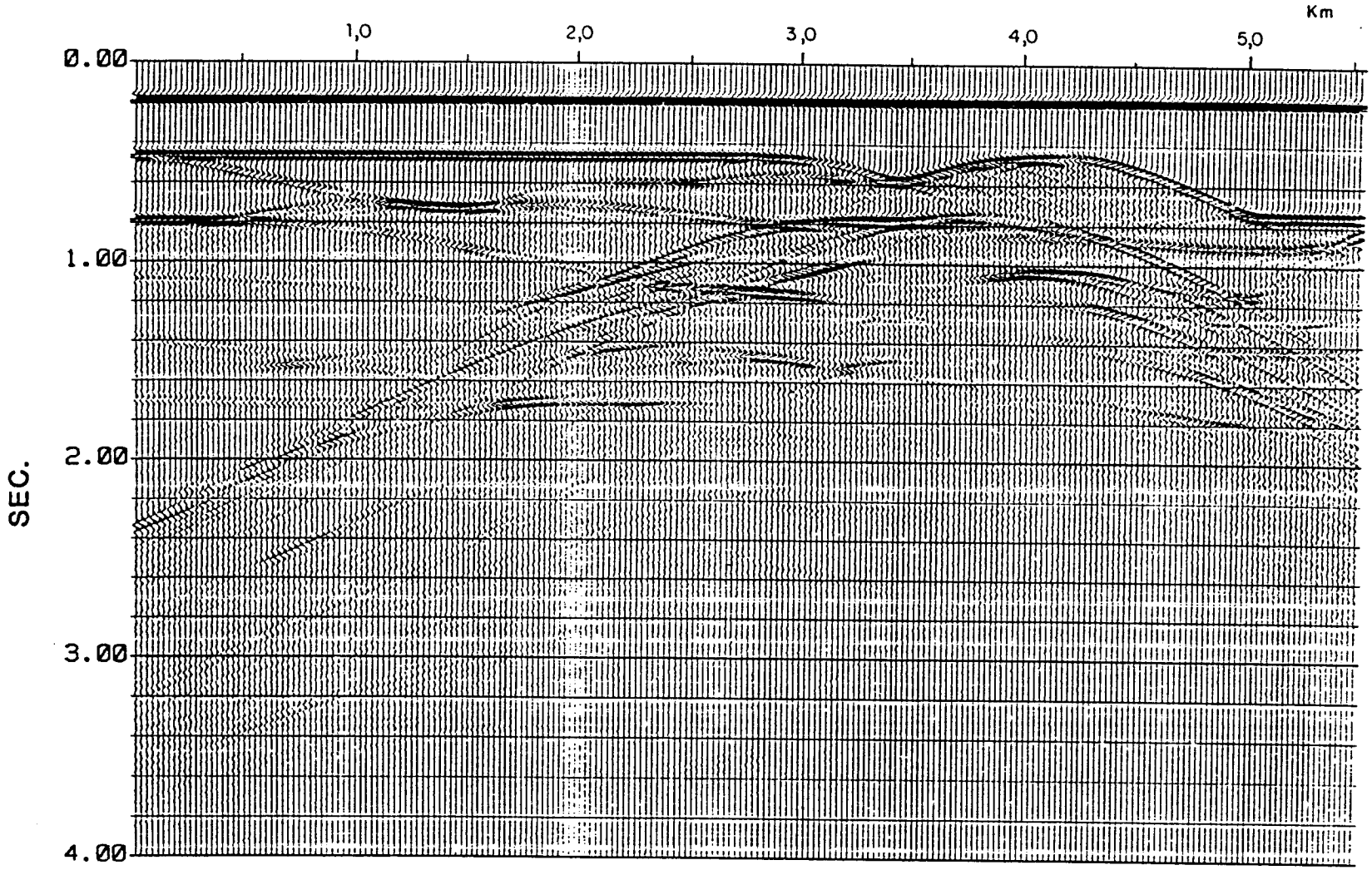


Figure 1. Near trace synthetic seismic section generated using a finite difference algorithm provided by Mobil Research. This data was used to test the 2D split-step Fourier migration.

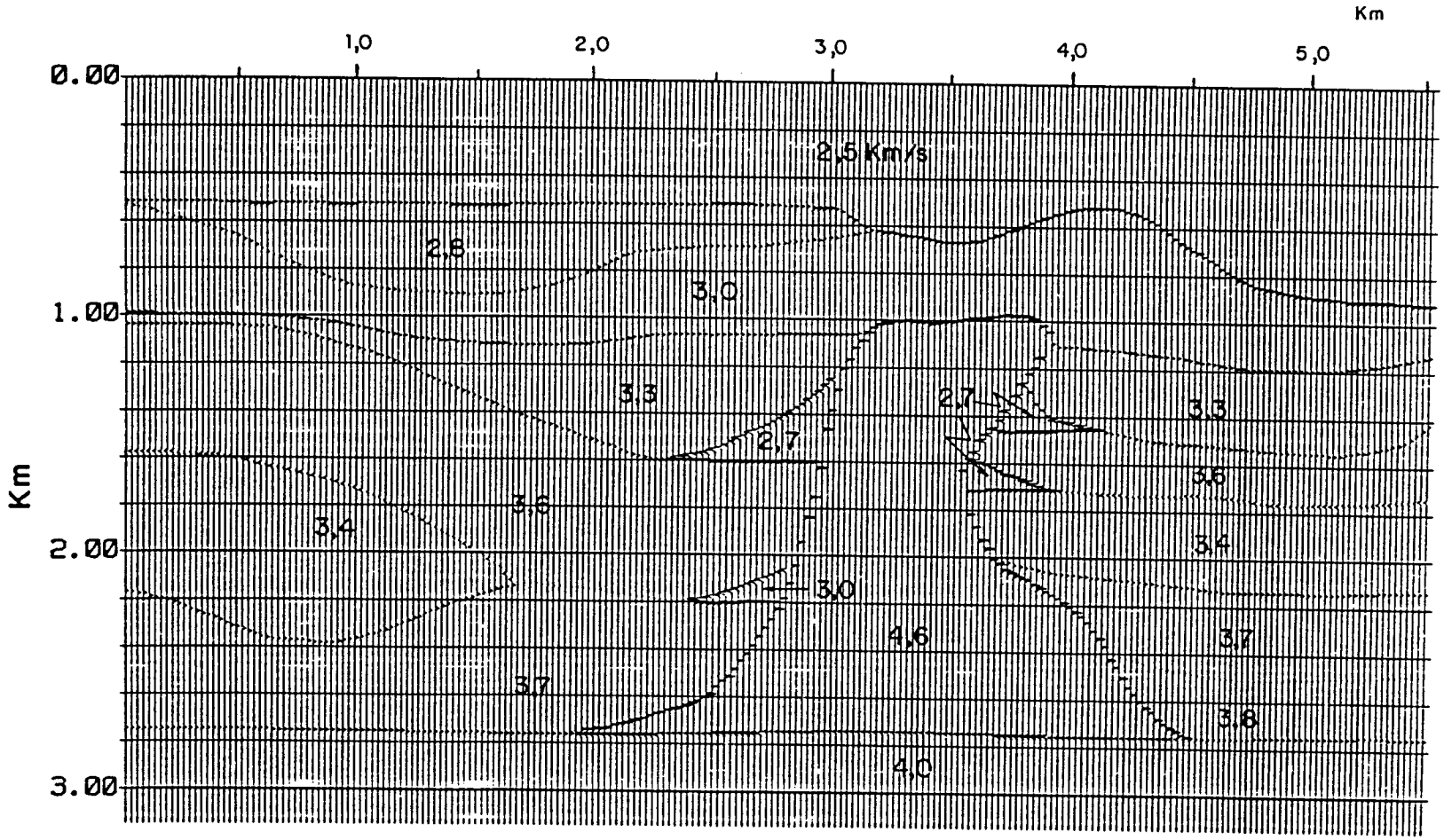


Figure 2. Velocity structure and reflection coefficients that correspond to the seismic data of Figure 1. This velocity function was used in the split-step Fourier migration examples, but had to be smoothed for the finite difference RTM of Figure 6. (Actually the slowness, reciprocal velocity, was smoothed.)

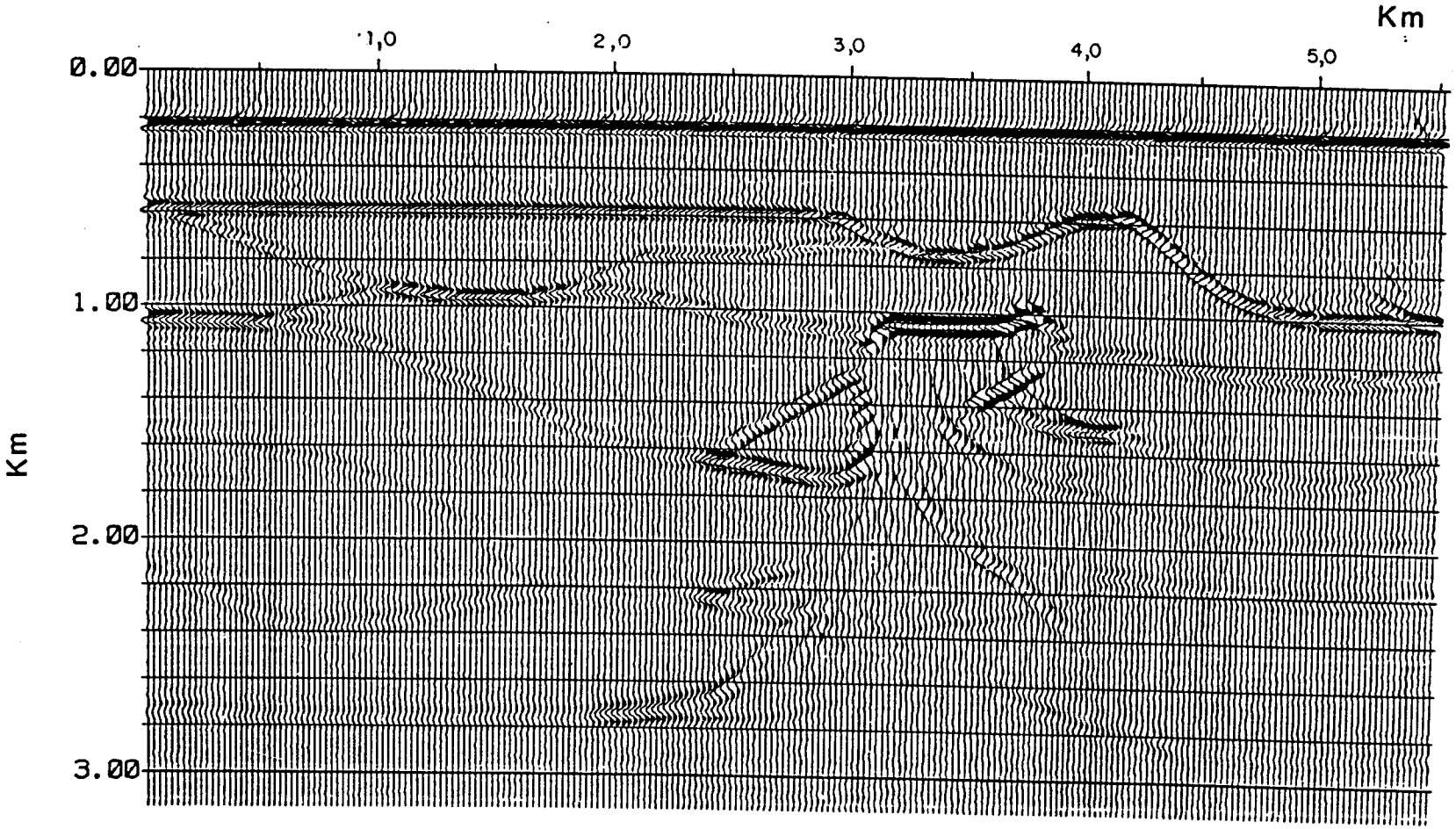


Figure 3. Phase shift migration using a constant velocity per depth interval. Note the imaging errors near the flanks of the salt dome.

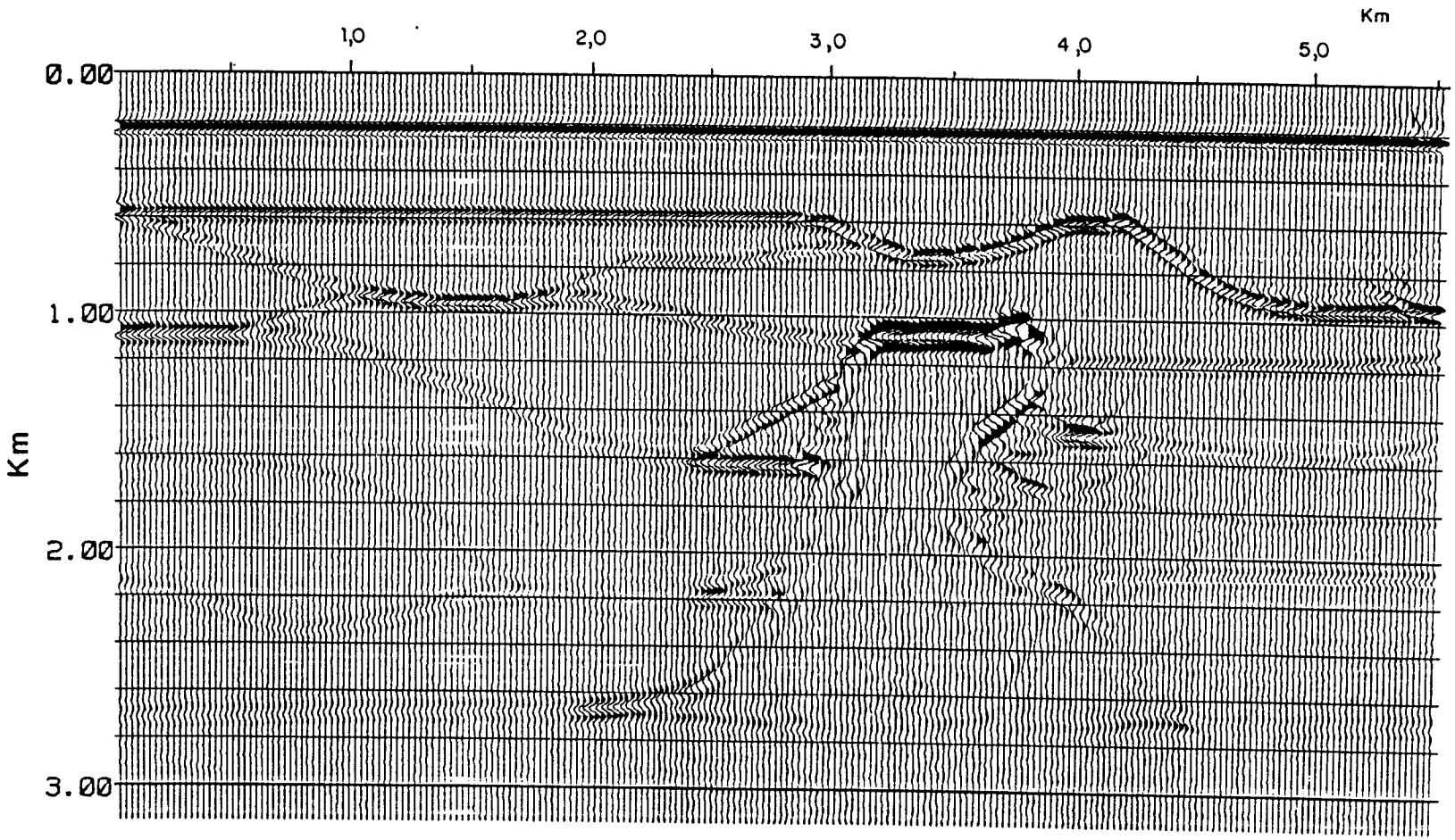


Figure 4. Split-step Fourier migration of the data of Figure 1. This image should be compared to Figures 3 and 6. Note the image in the vicinity of the flanks of the salt dome.

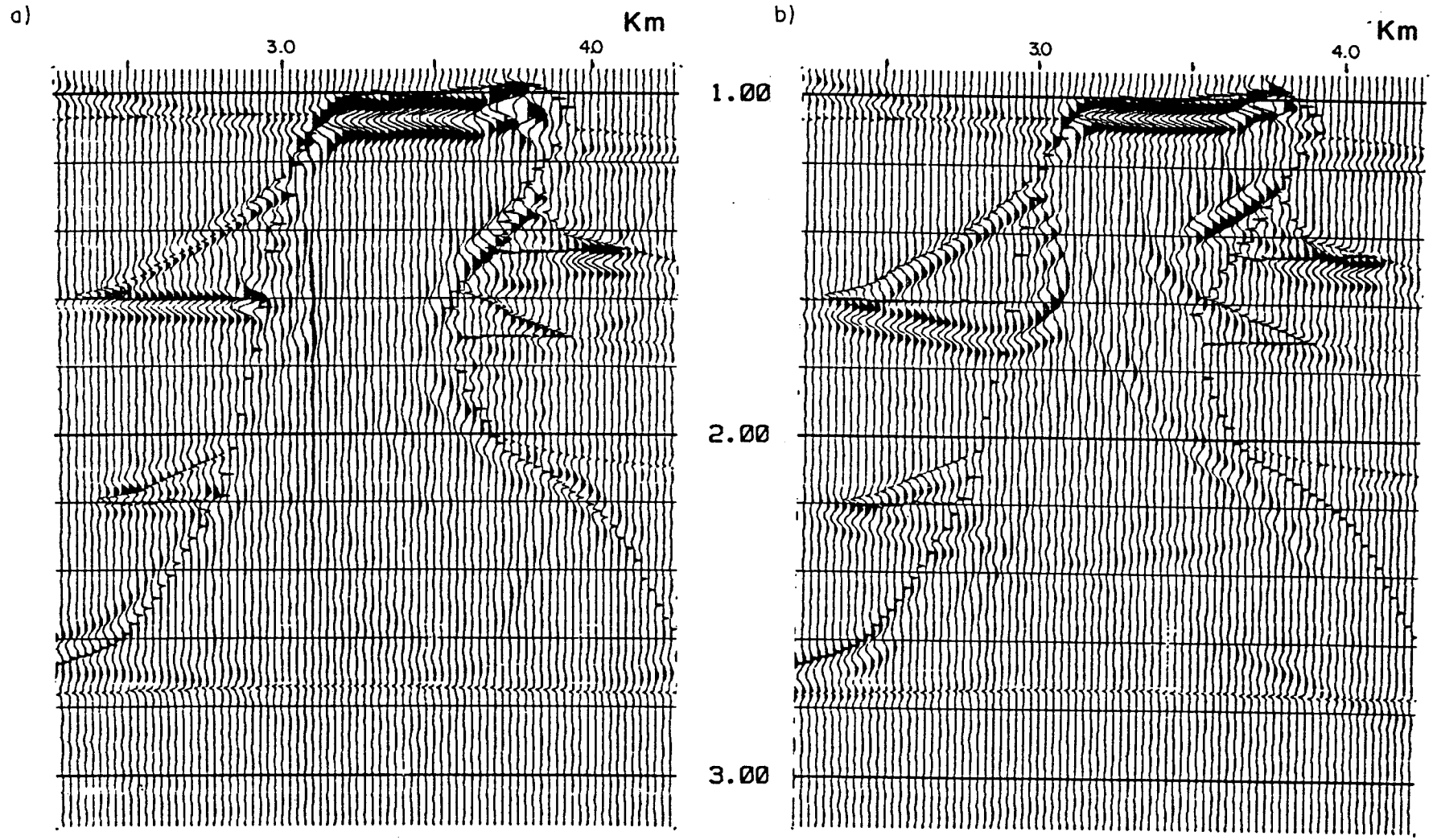


Figure 5. A detailed comparison of phase shift migration (right) and split-step Fourier migration (left). Superimposed on the migrated data are the reflection coefficients of Figure 2. These serve to indicate the correct image positions.

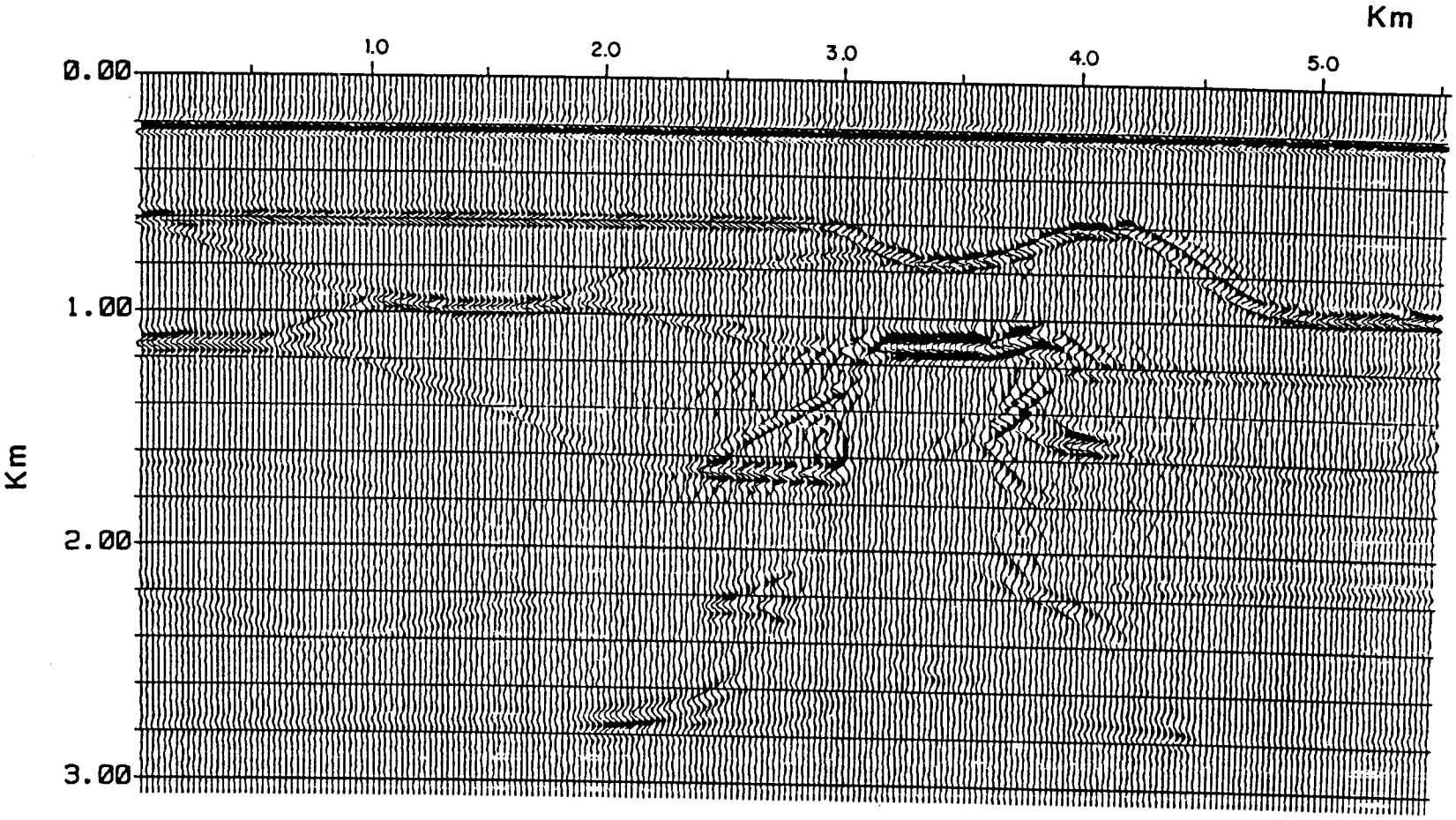


Figure 6. RTM finite difference of the data of Figure 1. This image is of lower quality due to grid dispersion. The velocity (actually reciprocal velocity) grid had to be smoothed prior to the migration to minimize the grid dispersion. Additional smoothing would help to further reduce this problem, but the resulting image would be less accurate.

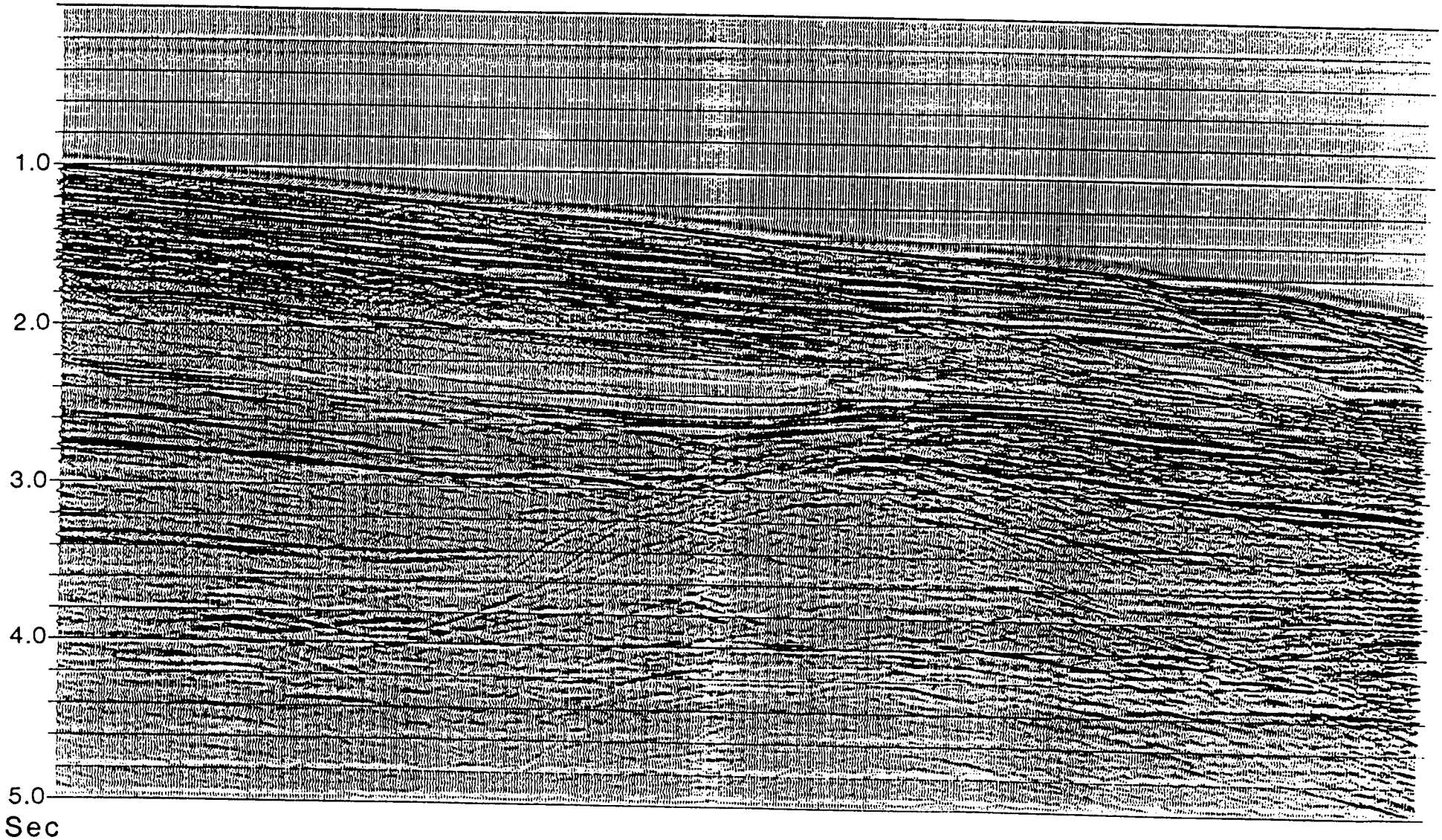


Figure 7. CDP stacked marine line prior to migration.

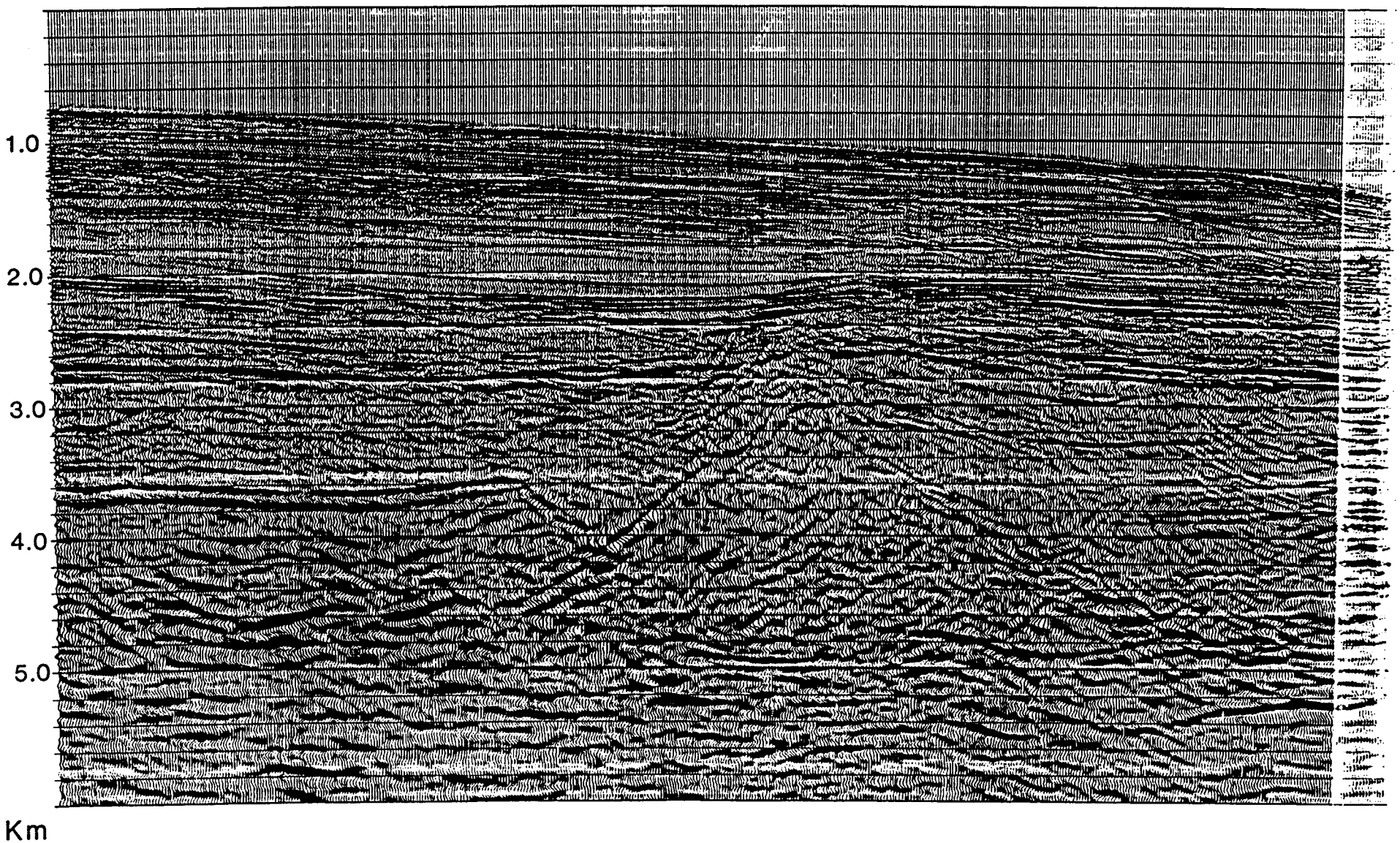


Figure 8. Migrated data using the split-step Fourier method corresponding to Figure 7. Note the complex structure in the center of the section.

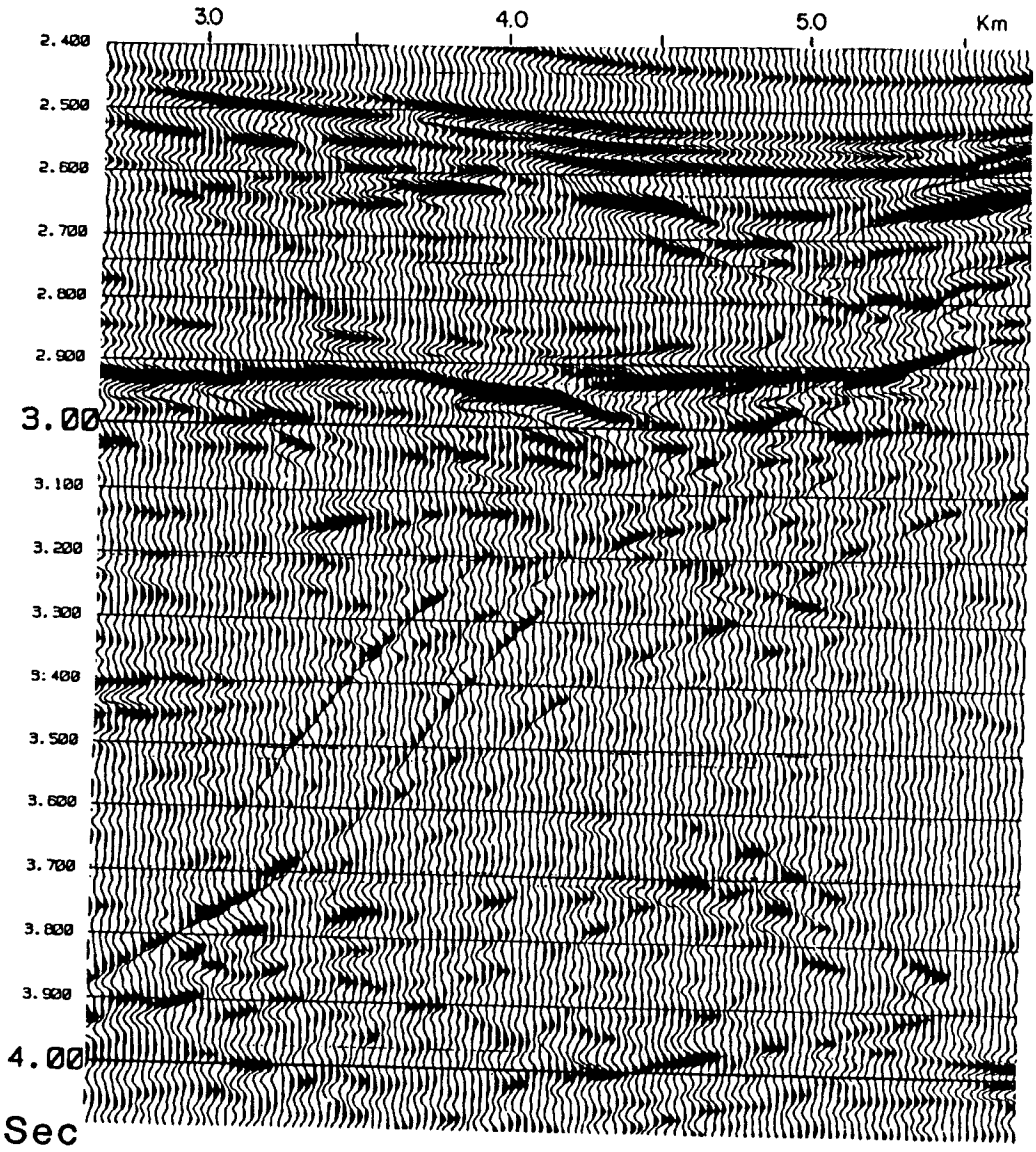


Figure 9. Detail of the original stacked data from Figure 7 near the center of the section.

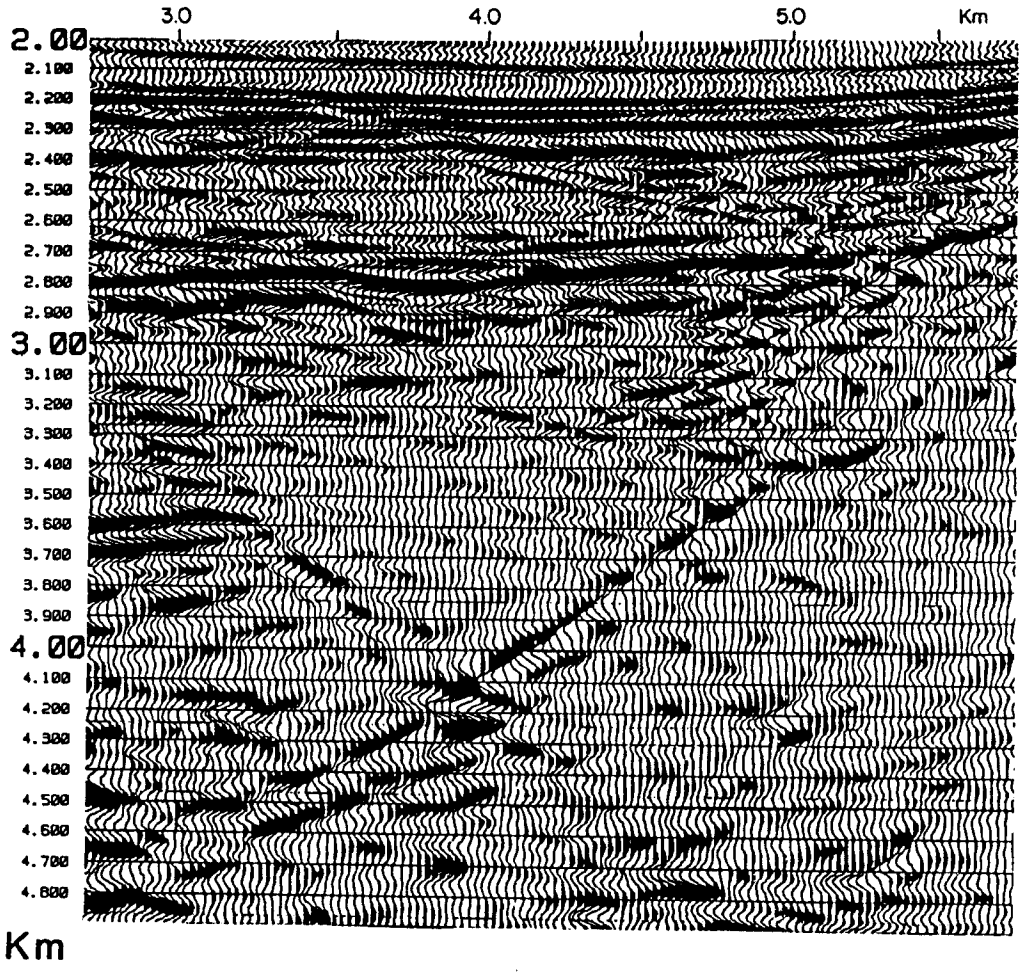


Figure 10. Migrated data using the split-step Fourier method corresponding to the data of Figure 9.

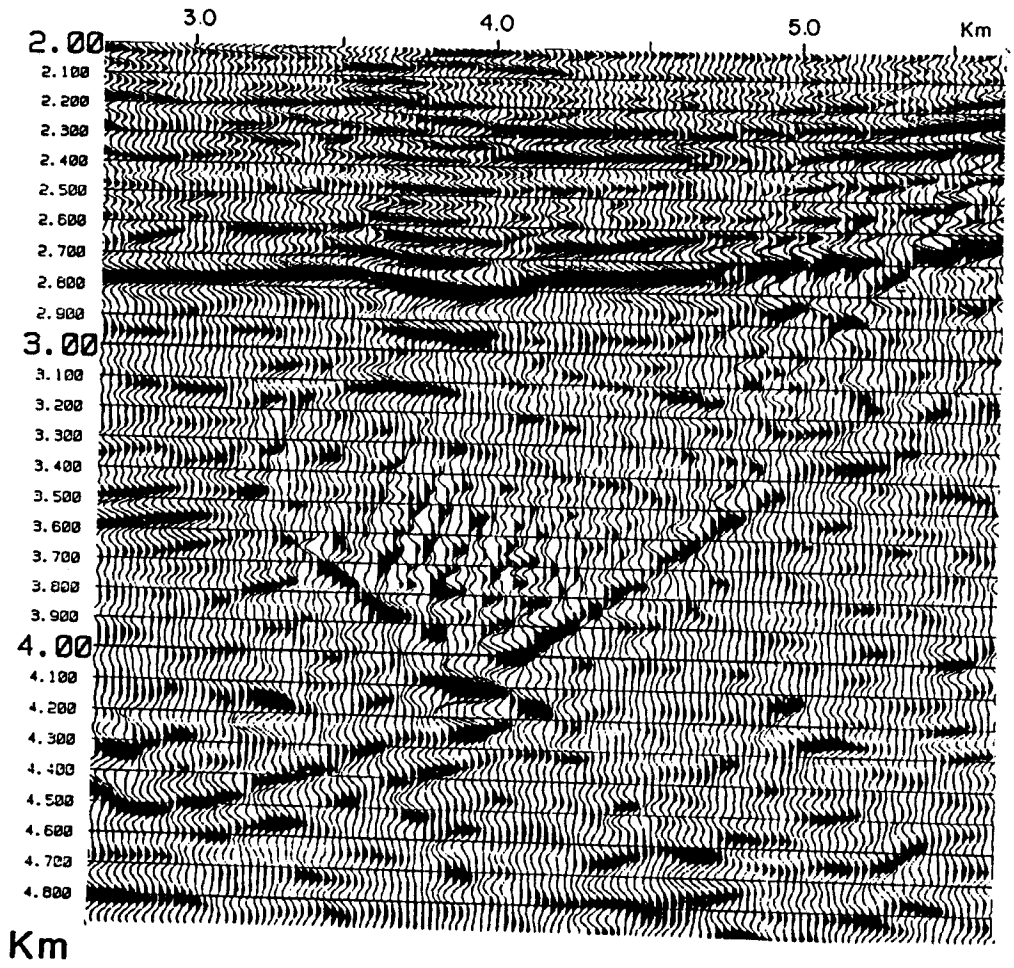


Figure 11. Finite difference RTM corresponding to the data of Figures 9 and 10. Note the similarity of the images, but the generally lower frequency character and grid dispersion of this finite difference generated image.

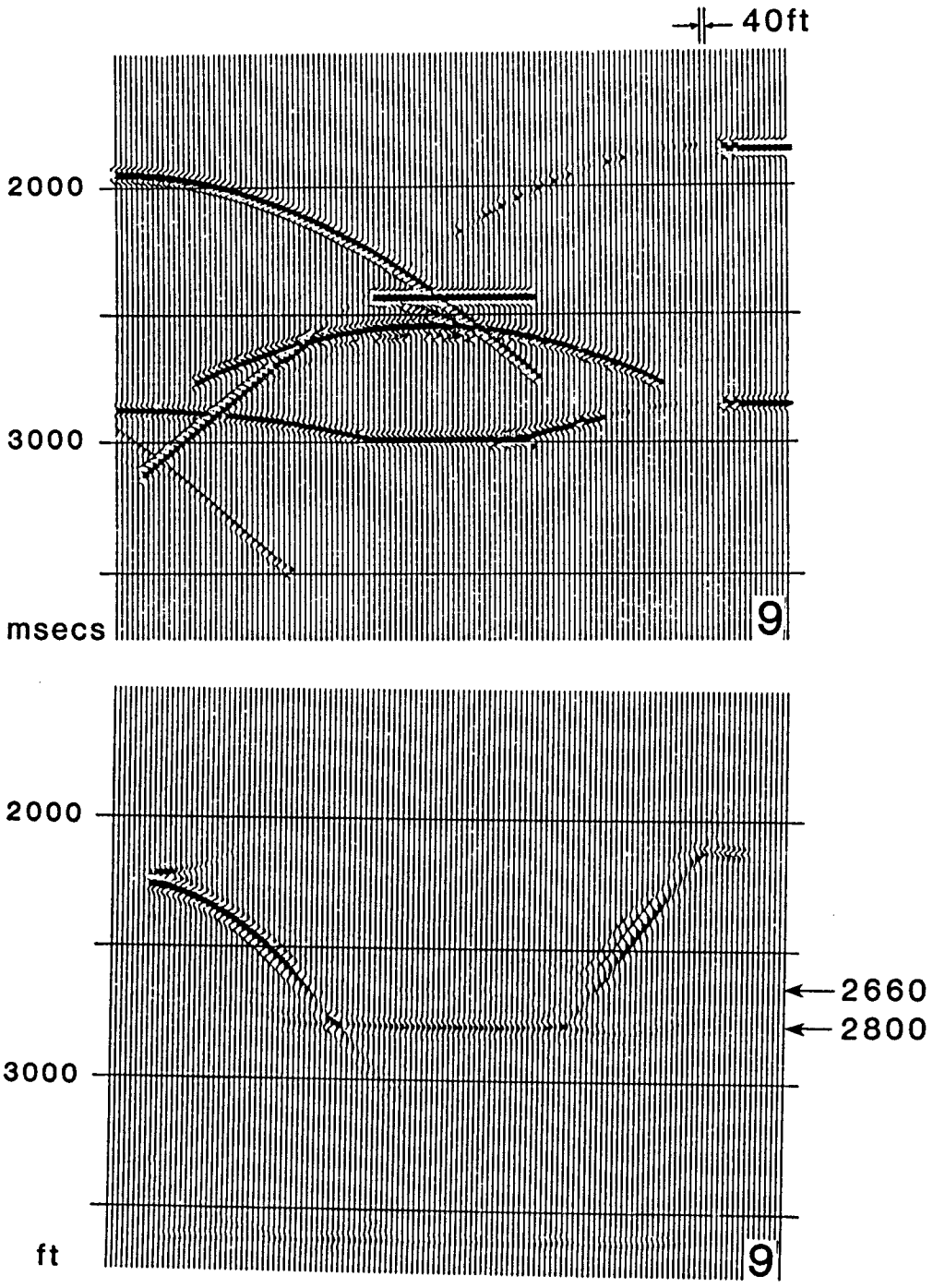


Figure 12. (Upper) line 9 from the synthetic 3D zero offset data volume for the French 3D model generated using the Sierra ray trace modelling software. (Lower) 3D split-step Fourier migrated result for Line 9.

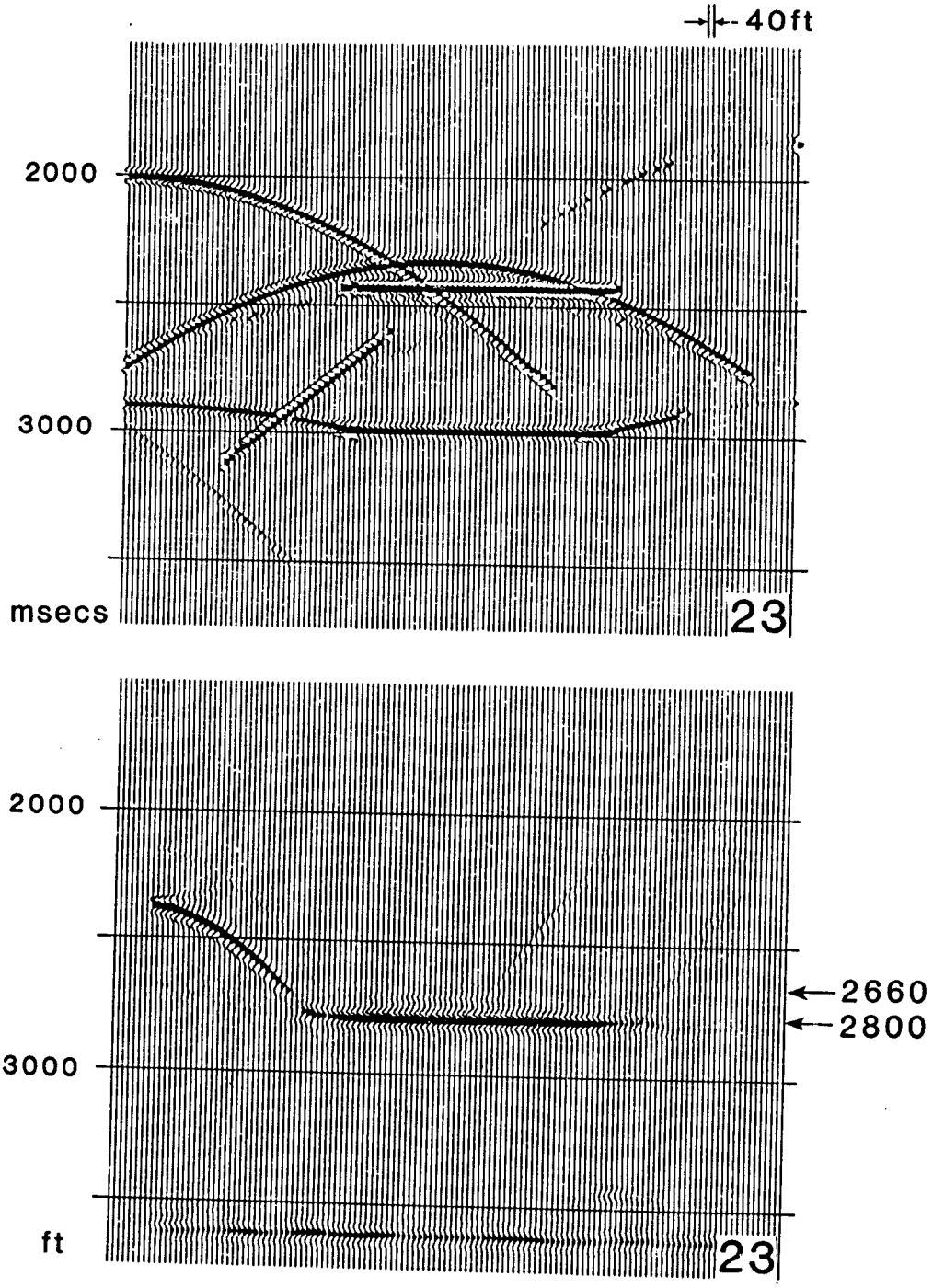


Figure 13. (Upper) line 23 from the synthetic 3D zero offset data volume for the French 3D model generated using the Sierra ray trace modelling software. (Lower) 3D split-step Fourier migrated result for Line 23.

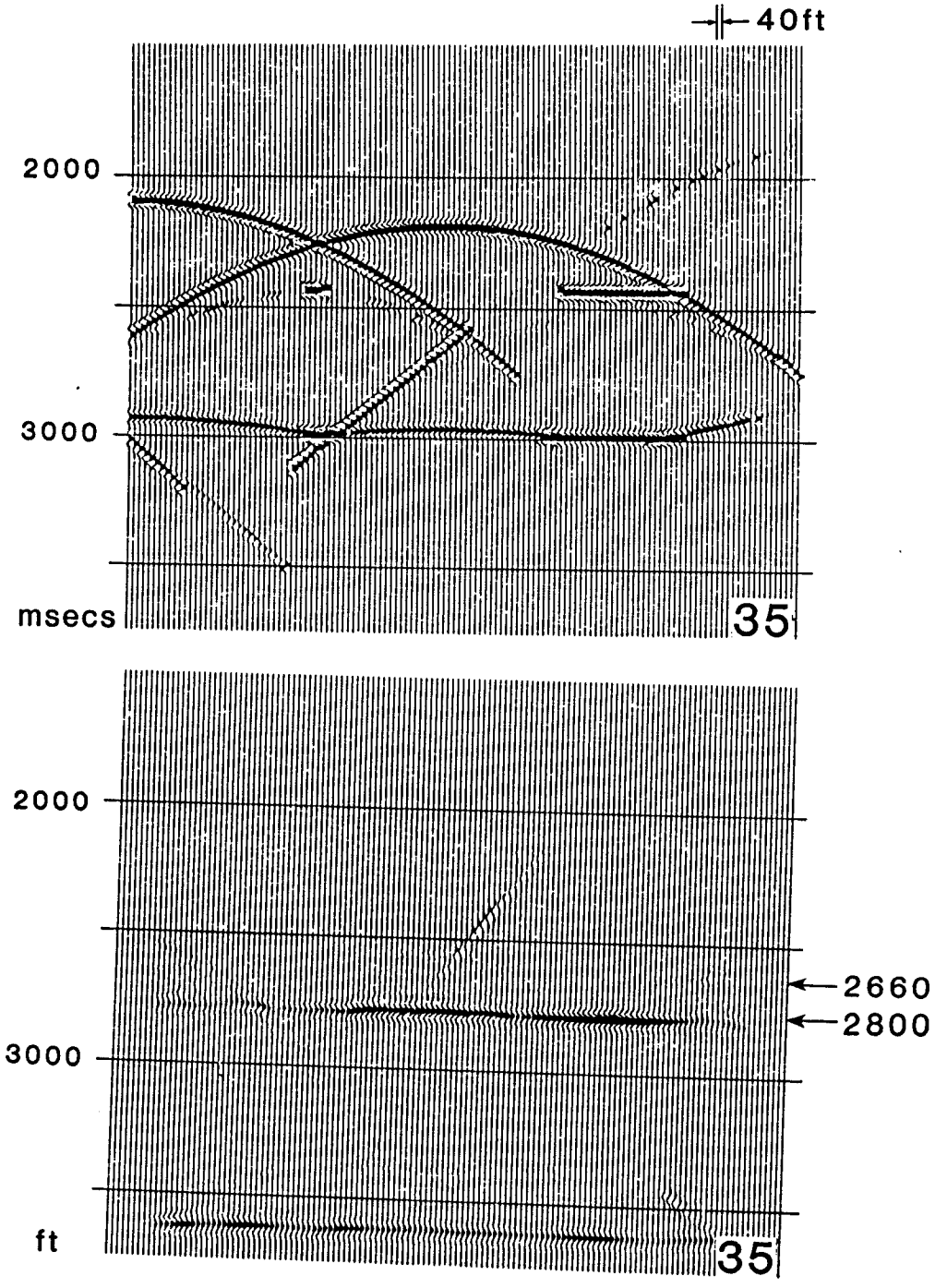


Figure 14. (Upper) line 35 from the synthetic 3D zero offset data volume for the French 3D model generated using the Sierra ray trace modelling software. (Lower) 3D split-step Fourier migrated result for Line 35.

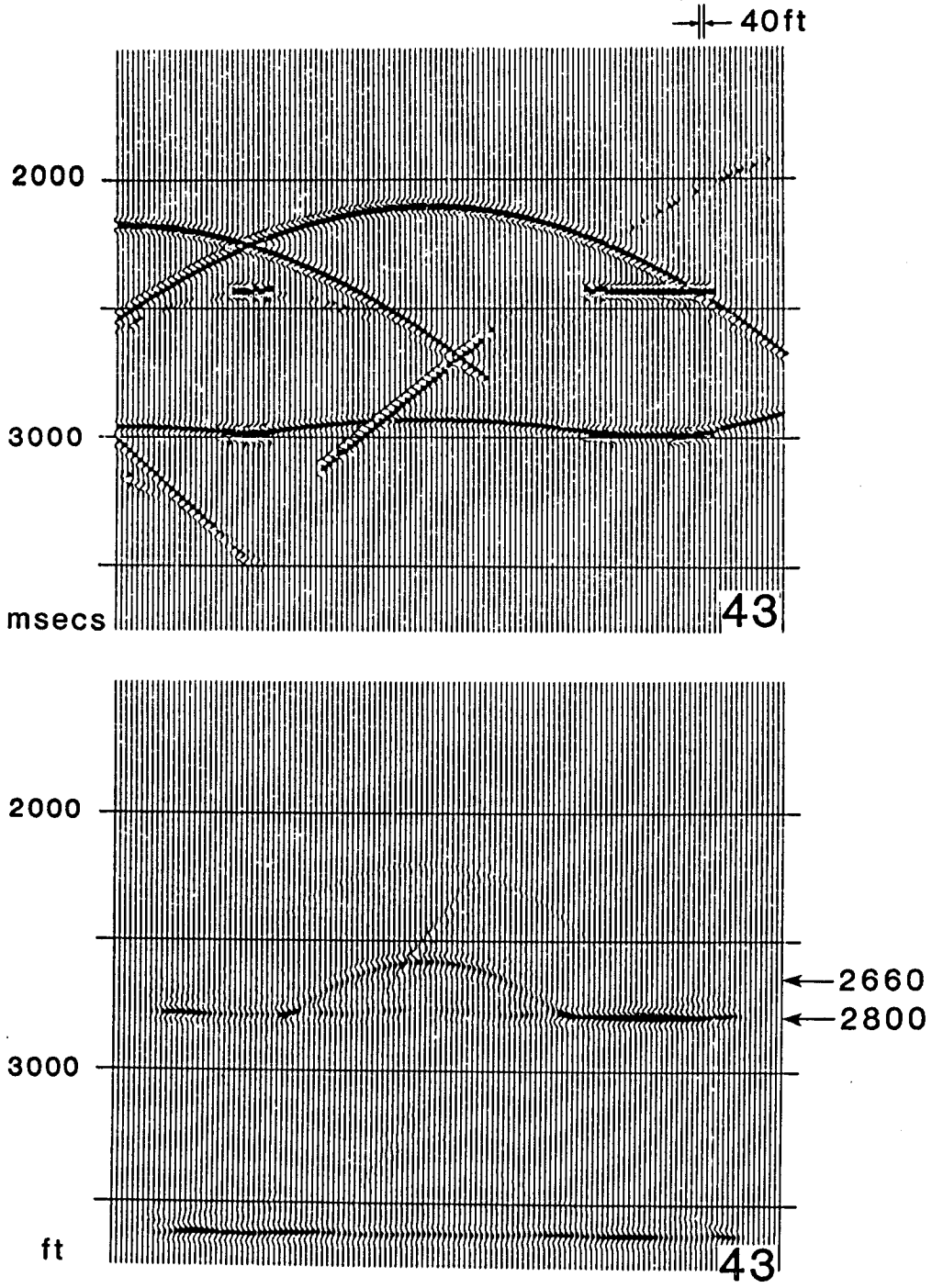


Figure 15. (Upper) line 43 from the synthetic 3D zero offset data volume for the French 3D model generated using the Sierra ray trace modelling software. (Lower) 3D split-step Fourier migrated result for Line 43.

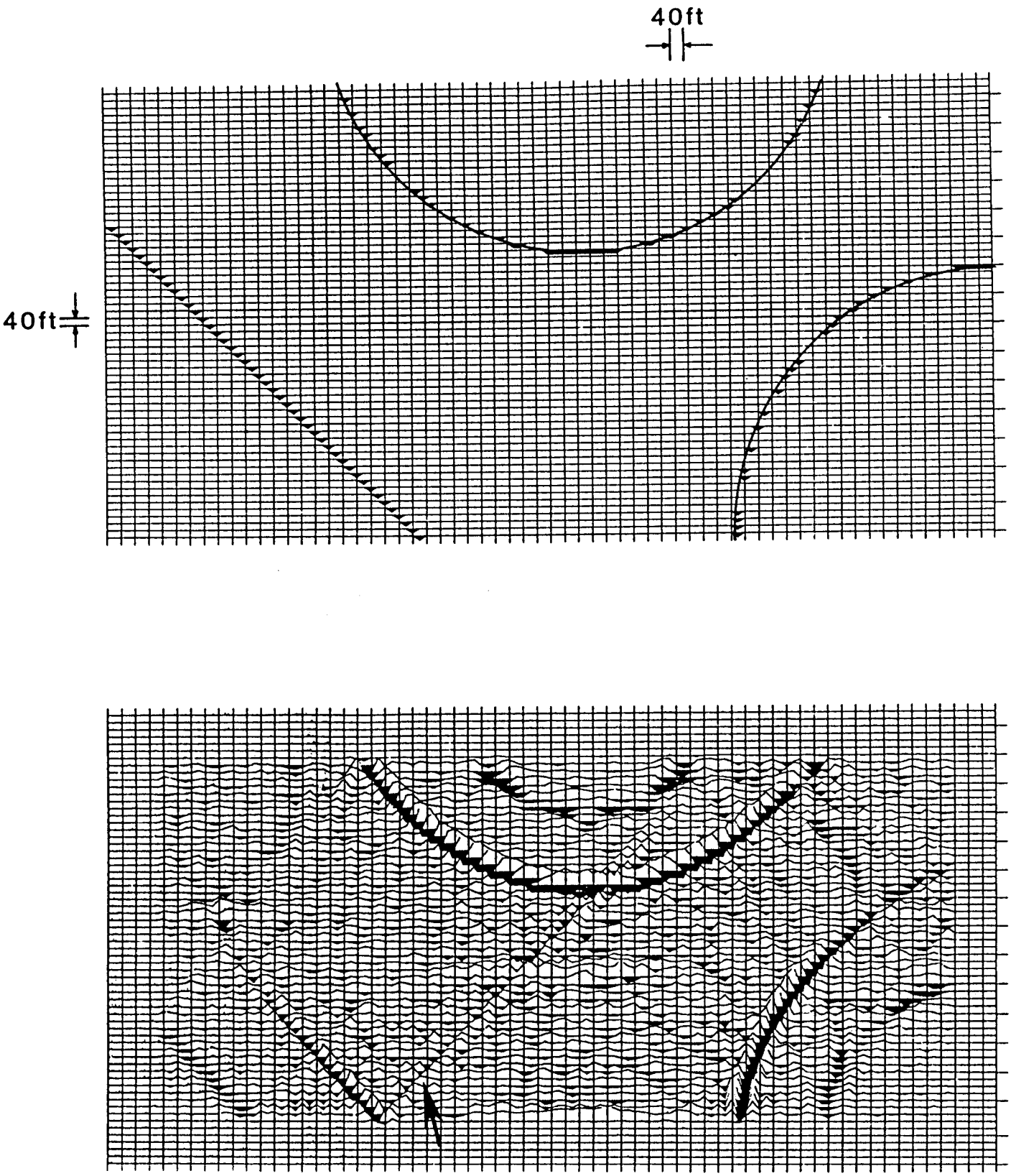


Figure 16. Synthetic reflection coefficients (upper left) obtained by taking first differences of the 3D velocity field at a depth of 2,660 ft. Corresponding depth slice after the split-step Fourier migration. Note that the edges of each dome are correctly imaged as is the edge of the fault. Note also the spurious reflection from the edge of the data that was not attenuated by the absorbing boundary (see arrow). Figures 12 to 15 indicate the position of this depth in section format.

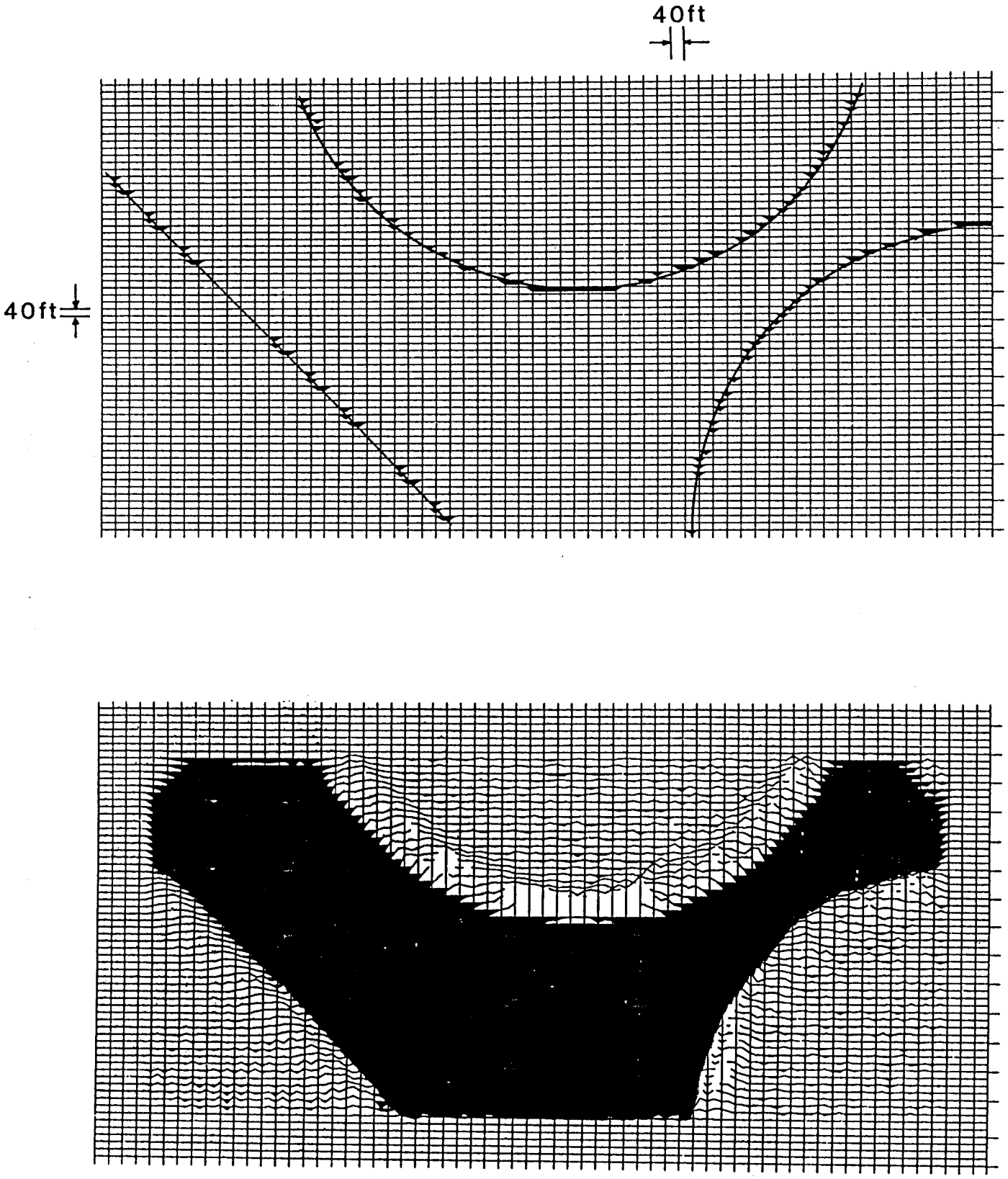
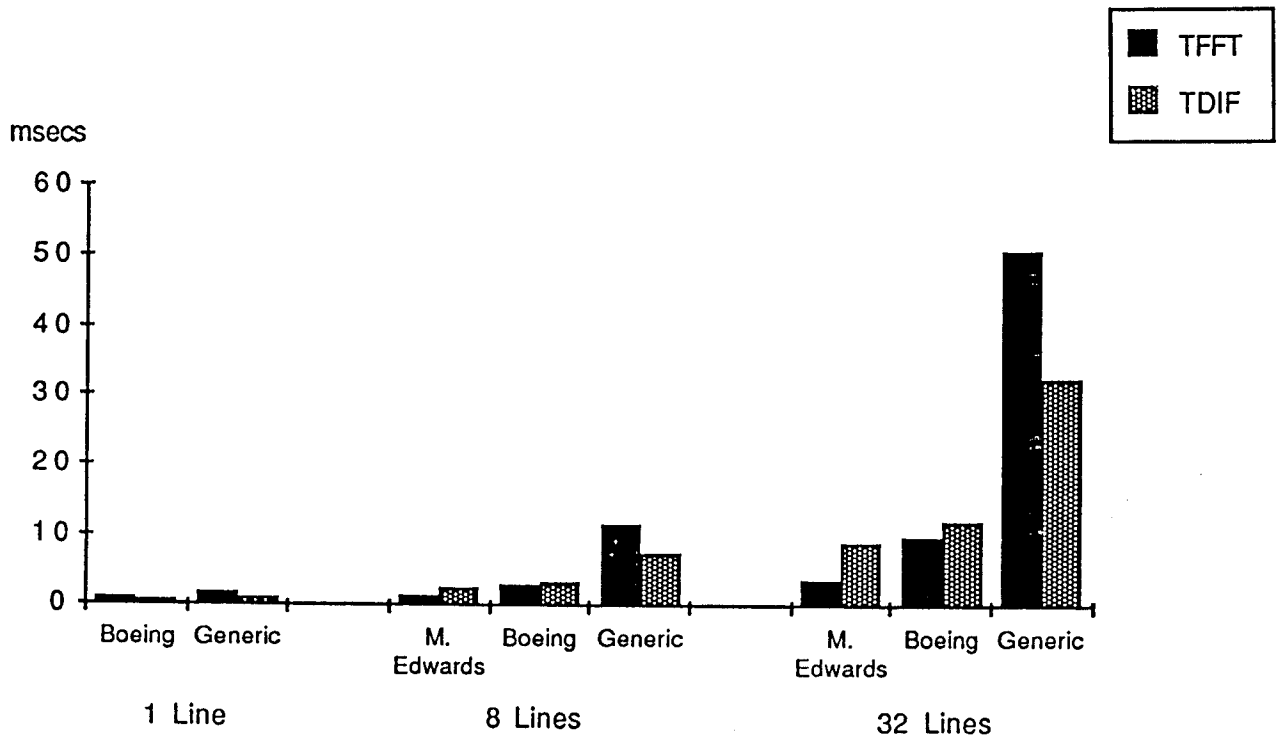


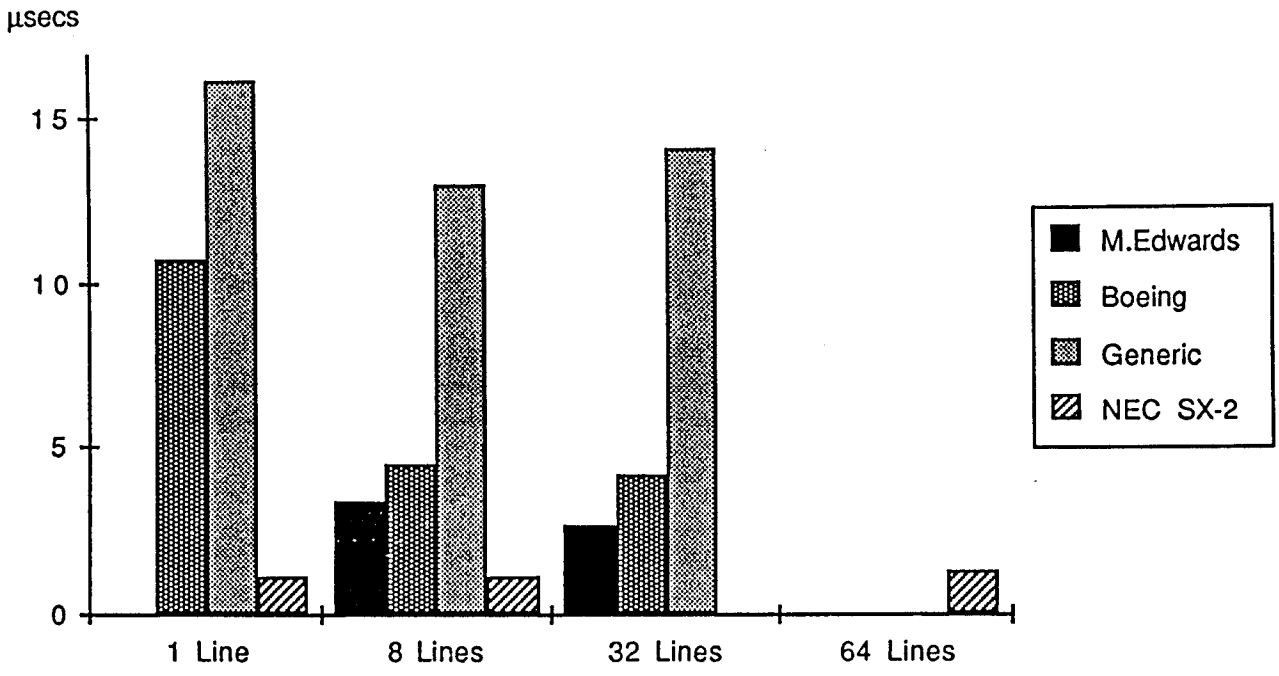
Figure 17. Synthetic reflection coefficients (upper left) obtained by taking first differences of the 3D velocity field at a depth of 2,800 ft. Corresponding depth slice after the split-step Fourier migration. The edges of each dome are correctly imaged. Figures 12 to 15 indicate the position of this depth in section format.



Time in msecs

	1 line			8 Lines			32 Lines	
	Boeing	Generic	M. Edwards	Boeing	Generic	M. Edwards	Boeing	Generic
TFFT	1.13664	1.81979	1.64891	2.81178	11.4658	3.8525	9.76533	50.3077
TDIF	0.79539	1.13452	2.56357	3.15169	7.54583	8.8544	11.7729	32.0565

Figure 18. Comparison of the total fast Fourier transform time, TFFT and all other computation times, TDIF per frequency per depth for the split-step Fourier method, using a Fortran FFT, the Boeing Library and the FFT of M. Edwards of Cray Research.



Time in µsecs

	1 Line	8 Lines	32 Lines	64 Lines
M.Edwards		3.35878	2.66738	
Boeing	10.7691	4.49018	4.10843	
Generic	16.1112	12.9351	13.9818	
NEC SX-2	1.134	1.13		1.248

Figure 19. Comparison of the time per grid point (x by y) per frequency per depth for the split-step Fourier method. These normalized times are now nearly the same for each coding method in the 2D cases. The NEC SX-2 times which were timed for 1, 8 and 64 lines are also included for comparison. The total migration time can be approximated by multiplying any individual time by the number of points in the spatial grid, (x by y) by the number of frequencies, by the number of depths migrated (or modelled).

Appendix I

Consider the propagation of compressional waves in an acoustic constant density medium using the wave equation:

$$\nabla^2 p - u^2 \frac{\partial^2}{\partial t^2} p = 0, \tag{1}$$

Where $p = p(x, z, t)$ is the pressure and $u = u(x, z)$ is the media slowness which is defined as the inverse of the propagation velocity:

$$u(x, z) = 1/v(x, z). \tag{2}$$

After transforming equation 1 to the frequency domain we have:

$$\nabla^2 P + \omega^2 u^2 P = 0, \tag{3}$$

where:

$$P(x, z, \omega) = \int_{-\infty}^{\infty} p(x, z, t) \exp(-i\omega t)\, dt. \tag{4}$$

We decompose the slowness field, $u(x, z)$ into two components:

$$u(x, z) = u_0 + \Delta u(x,z), \tag{5}$$

where we define u_0 as the reference slowness and all variations are accommodated by the $\Delta u(x, z)$ component. Substituting into equation 3 we have:

$$\nabla^2 P + \omega^2 u_0^2 P = - \omega^2 (2u_0 \Delta u + \Delta u^2) P \tag{6}$$

or

$$\nabla^2 P + \omega^2 u_0^2 P = - S(x, z, \omega), \tag{7}$$

where S is defined as:

$$S(x, z, \omega) = \omega^2 [2u_0 \Delta u(x,z) + \Delta u^2(x,z)] P(x,z,\omega) . \quad (8)$$

Thus, the homogeneous acoustic wave equation, 3, has been transformed to the inhomgeneous wave equation, 7, by the inclusion of a source-like term, S (x, z, ω), due to the slowness variations.

Equation 7, after a spatial Fourier transform can be expressed in terms of the horizontal wave number, k_x, as:

$$\frac{\partial^2}{\partial z^2} \tilde{P} + (\omega^2 u_0^2 - k_x^2) \tilde{P} = - \tilde{S}, \quad (9)$$

where $\tilde{P} = \tilde{P}(k_x, z, \omega)$ is defined as:

$$\tilde{P}(k_x, z, \omega) = \int_{-\infty}^{\infty} P(x, z, \omega) \exp(ik_x x) dx, \quad (10)$$

and $\tilde{S} = \tilde{S}(k_x, z, \omega)$ is defined as:

$$\tilde{S}(k_x, z, \omega) = \int_{-\infty}^{\infty} S(x, z, \omega) \exp(ik_x x) dx. \quad (11)$$

We now re-write equation 9 in terms of a reference vertical wave number k_{z_0}:

$$\frac{\partial^2}{\partial z^2} \tilde{P} + k_{z_0}^2 \tilde{P} = -\tilde{S}, \quad (12)$$

where k_{z_0} corresponds to the constant slowness case described by the reference slowness, u_0:

$$k_{z_0} = (\omega^2 u_0^2 - k_x^2)^{1/2} \tag{13}$$

where we choose, the real and imaginary parts as follows:

$$\text{Re } \{k_{z_0}\} \geq 0$$

and (14)

$$\text{Im } \{k_{z_0}\} \geq 0$$

The general solution of equation 12 is:

$$\tilde{P}(k_x, z, \omega) = A\_e^{ik_{z_0}z} + A_+ e^{ik_{z_0}z} + \int_{-\infty}^{\infty} \frac{e^{ik_{z_0}|z-z'|}}{2ik_{z_0}} \tilde{S}(k_x, z', \omega)\, dz' \tag{15}$$

where $A\_e^{ik_{z_0}z}$ and $A_+ e^{-ik_{z_0}z}$ describe the upgoing and downgoing waves respectively. (The coefficients $A\_$, and A_+ will be determined by the boundary conditions.)

The complete wavefield, $\tilde{P}(k_x, z, \omega)$ is the superposition of the upgoing wavefield $\tilde{P}\_(k_x, z, \omega)$ and the downgoing wafefield $\tilde{P}_+(k_x, z, \omega)$:

$$\tilde{P}(k_x, z, \omega) = \tilde{P}\_(k_x, z, \omega) + \tilde{P}_+(k_x, z, \omega) \tag{16}$$

where:

$$\tilde{P}_+(k_x, z, \omega) = A_+ e^{-ik_{z_0}z} + e^{ik_{z_0}z} \int_{-\infty}^{z} \frac{e^{ik_{z_0}z'}}{2ik_{z_0}} \tilde{S}(k_x, z', \omega)\, dz' ,$$

and (17)

$$\tilde{P}\_(k_x, z, \omega) = A\_e^{ik_{z_0}z} + e^{ik_{z_0}z} \int_{z}^{\infty} \frac{e^{ik_{z_0}z'}}{2ik_{z_0}} \tilde{S}(k_x, z', \omega)\, dz' .$$

During migration, we extrapolate the upgoing wavefield downward to successfully greater depths, ignoring any wavefield interactions. In post-stack migration of common mid-point data, this is a reasonable approximation since the effect of multiples can often be ignored. The wavefield extrapolation is initiated by solving for the constant A_, using the upgoing wavefield recorded at the surface, $\tilde{P}_-(k_x, z = 0, \omega)$. Substituting into equation 17 we have:

$$A_- = \tilde{P}_-(k_x, z = 0, \omega) - \int_0^{\infty} \frac{e^{-ik_z z'}}{2ik_z} \tilde{S}(k_x, z, \omega)\, dz' \quad . \tag{18}$$

To extrapolate the upgoing wavefield downward, level by level, e.g., from level z_n to z_{n+1}, we first define:

$$z_{n+1} = z_n + \Delta z . \tag{19}$$

Then, the wavefield at level z_n is:

$$\tilde{P}_-(k_x, z_n, \omega) = A_- e^{ik_{z_0} z_n} + e^{ik_{z_0} z_n} \int_{z_n}^{\infty} \frac{e^{-ik_{z_0} z'}}{2ik_{z_0}} \tilde{S}(k_x, z', \omega)\, dz' . \tag{20}$$

and the wavefield at level z_{n+1} is:

$$\tilde{P}_-(k_x, z_{n+1}, \omega) = A_- e^{ik_{z_0} z_{n+1}} + e^{ik_{z_0} z_{n+1}} \int_{z_{n+1}}^{\infty} \frac{e^{-ik_{z_0} z'}}{2ik_{z_0}} \tilde{S}(k_x, z', \omega)\, dz' \tag{21}$$

Combining the above, we can describe the wavefield $\tilde{P}_-(k_x, z_{n+1}, \omega)$ in terms of the wavefield at the previous level, $\tilde{P}_-(k_x, z_n, \omega)$ as:

$$\tilde{P}_-(k_x, z_{n+1}, \omega) = \tilde{P}_-(k_x, z_n, \omega)\, e^{ik_{z_0} \Delta z} - \int_{z_n}^{z_{n+1}} \frac{e^{-ik_{z_0}(z' - z_{n+1})}}{2ik_{z_0}} \tilde{S}(k_x, z', \omega)\, dz' . \tag{22}$$

Until now, no restrictions have been made on the slowness field. Equation 22 could be solved numerically by integrating the source term for any arbitrary slowness variation.

To develop the split-step Fourier method, we will assume that the lateral variation of the slowness is small compared to the reference slowness, u_0:

$$u_0 >> |\Delta u(x,z)|\,, \tag{23}$$

Consequently, source terms of the order Δu^2 in equation 6 can be dropped and the source contribution, S (x, z, ω) is approximated by:

$$S\,(x,\,z,\,\omega) \;=\; 2\omega^2\, u_0 \Delta u(x,z)\; P_-\,(x,z,\omega)\,. \tag{24}$$

For migration, the reference slowness u_0, is specified for the depth interval z_n to z_{n+1}. That is, for each depth interval we are free to redefine u_0, based on the true slowness field u (x, z) at this level. Since we expect most of the lateral slowness variation to be less than the vertical variation, this is a reasonable approximation.

Now, we must consider the source term of equation 24 in more detail:

$$\tilde{S}(k_x,z,\omega) = 2\omega^2 u_0 \int_{-\infty}^{\infty} \Delta\tilde{u}\,(k_x - k'_x,z)\;\tilde{P}_-(k'_x,z,\omega)\;dk'_x\;, \tag{25}$$

where $\Delta\tilde{u}\,(k_x,\,z)$ is the spatial Fourier transfer of $\Delta u(x,z)$.

Substituting into the integral (I) of equation 22 and substituting for k_{z_0} we have:

$$(I) = \int_{z_n}^{z_{n+1}} dz' \,\frac{e^{i\omega u_0[1-k_x/\omega u_0)^2]^{1/2}(z_{n+1}-z')}}{2i\omega u_0[1-(k_x/\omega u_0)^2]^{1/2}}\; 2\omega^2 u_0 \int_{-\infty}^{\infty} \Delta\tilde{u}(k_x-k'_x,z')\;\tilde{P}_-(k'_x,\,z',\,\omega)\;dk'_x \tag{26}$$

which after expanding the denominator and keeping only the first term simplifies to:

$$(I) = -i\omega \int_{z_n}^{z_{n+1}} dz' \int_{-\infty}^{\infty} \tilde{\Delta u}(k_x - k_x', z')\ \tilde{P}_-(k_x', z', \omega)\ e^{ik_{z_0'}(z_{n+1} - z')}\, dk_x' + \varepsilon \tag{27}$$

where:

$$k_{z_0'} = \omega u_0 \left[1 - (k_x'/\omega u_0)^2\right]^{1/2} \tag{28}$$

We now define $\tilde{P}_1(k_x, z, d, \omega)$ as the upgoing wavefield extrapolated to a depth between levels z_n and z_{n+1}. First, considering the slowness constant between z and z+d we have:

$$\tilde{P}_1(k_x, z, d, \omega) = \tilde{P}_-(k_x, z, \omega) \exp(ik_{z_0} d) \tag{29}$$

and considering

$$z_n \le z \le z_{n+1}$$

and (30)

$$z_n \le z + d \le z_{n+1}$$

where d is non-negative and is the depth of the extrapolation. In particular, we note:

$$\tilde{P}_1(k_x, z, 0, \omega) = \tilde{P}_-(k_x, z, \omega)\ . \tag{31}$$

Substituting (29) into equation (22) we have:

$$(I) = -i\omega \int_{z_n}^{z_{n+1}} dz' \int_{-\infty}^{\infty} \tilde{\Delta u}(k_x - k_x', z')\ \tilde{P}_1(k_x', z', d_{n+1}, \omega)\, dk_x' \tag{32}$$

where:

$$d_{n+1} = z_{n+1} - z'.$$

Substituting 32 into 22 we have:

$$\tilde{P}_{-}(k_x,z_{n+1},\omega) = \tilde{P}_1(k_x,z_n,\Delta z,\omega) + i\omega \int_{z_n}^{z_{n+1}} dz' \int_{-\infty}^{\infty} \Delta\tilde{u}(k_x-k'_x,z')\ \tilde{P}_1(k'_x,z',d_{n+1},\omega)\ dk'_x \quad . \quad (33)$$

Taking the inverse Fourier transform from k_x to x, we can reduce the convolution of $\Delta\tilde{u}$ with $\tilde{P}_1$ in the k_x domain to the product of their inverse Fourier transforms in the x domain:

$$P_{-}(x,z_{n+1},\omega) = P_1(x,z_n,\Delta z,\omega) + i\omega \int_{z_n}^{z_{n+1}} dz'\ \Delta u(x,z')\ P_1(x,z',d_{n+1},\omega), \quad (34)$$

where $P_1(x,z,d,\omega)$ is defined as the inverse Fourier transform of $\tilde{P}_1(k_x,z,d,\omega)$, which is:

$$P_1(x,z,d,\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-ik_x x}\, e^{ik_{z_0} d}\, \tilde{P}_{-}(k_x,z,\omega) dk_x \quad . \quad (35)$$

Considering that $\Delta z = z_{n+1}-z_n$ is sufficiently small, the integral of equation 34 can be evaluated using the trapezoid rule:

$$\int_{z_n}^{z_{n+1}} dz'\ \Delta u(x,z')\ P_1(x,z',z_{n+1}-z',\omega)$$

$$= \frac{\Delta z}{2} [\Delta u(x,z_{n+1})\ P_1(x,z_{n+1},0,\omega) + \Delta u(x,z_n)\ P_1(x.z_n,\Delta z,\omega)] \quad . \quad (36)$$

or, using equation 31:

$$\int_{z_n}^{z_{n+1}} dz'\, \Delta u(x,z')\, P_1(x,z',z_{n+1}-z',\omega) =$$

$$= \frac{\Delta z}{2} [\Delta u(x,z_{n+1})\, P_-(x,z_{n+1},\omega) + \Delta u(x,z_n)\, P_1(x,z_n,\Delta z,\omega)] \quad . \tag{37}$$

Substituting 36 into 34 gives:

$$P_-(x,z_{n+1},\omega)\,[1 - \frac{1}{2} i\omega\Delta u(x,z_{n+1})\Delta z] = P_1(x,z_n,\Delta z,\omega)\,[1 + \frac{1}{2} i\omega\Delta u(x,z_n)\Delta z] \quad . \tag{38}$$

Remembering that we have ignored terms on the order of $\Delta u^2(x, z)$, equation 38 is equivalent to:

$$P_-(x,z_{n+1},\omega)\, e^{-(1/2)i\omega\Delta u\,(x,z_{n+1})\,\Delta z} = P_1(x,z_n,\Delta z,\omega)\, e^{(1/2)i\omega\Delta u(x,z_n)\Delta z} \tag{39}$$

or solving for $P_-(x,z_{n+1},\omega)$:

$$P_-(x,z_{n+1},\omega) = e^{(1/2)i\omega[\Delta u(x,z_{n+1}) + \Delta u(x,z_n)]\Delta z} P_1(x,z_n,\Delta z,\omega) \quad . \tag{40}$$

If we stop and consider the case where there is no vertical slowness variation in the depth interval, then:

$$P_-(x,z_{n+1},\omega) = e^{i\omega\Delta u(x)\Delta z} P_1(x,z_n,\Delta z,\omega) \quad . \tag{41}$$

where the spatial Fourier transform of $P_1(x,z_n,\Delta z,u)$ is:

$$\tilde{P}_1\,(k_x,z_n,\Delta z,\omega) = \tilde{P}_-(k_x,z_n,\omega)\, e^{ik_{z_0}\Delta z} \tag{42}$$

which are the equations used in split-step migration described earlier.

It is possible to improve this result. The integral, equation 34, used in the depth extrapolation was evaluated using the trapezoid rule. We can improve the accuracy by using a linear approximation for $P_1(x, z', d_{n+1}, \omega)$ in the interval $z_n \le z' \le z_{n+1}$. The result we obtain will in general be different than equation 40 but, reduces to that case when there is no vertical slowness variation across the interval. The depth extrapolation through a constant slowness media is approximated by:

$$P_1(x,z',d_{n+1},\omega) = P_1(x,z_n,\Delta z,\omega) + [P\_(x,z_{n+1},\omega) - P_1(x,z_n,\Delta z,\omega)]\ (z'-z_n)/\Delta z \quad . \qquad (43)$$

Using equations 43 and 34 we can write:

$$P\_(x,z_{n+1},\omega) = P_1(x,z_n,\Delta z,\omega) + i\omega\ P_1(x,z_n,\Delta z,\omega)\ I\ \Delta u(x) +$$

$$i\omega[P\_(x,z_{n+1},\omega) - P_1(x,z_n,\Delta z,\omega)]\ I\Delta u2(x) \qquad (44)$$

To simplify the notation we introduce:

$$I\Delta u(x) = \int_{z_n}^{z_{n+1}} \Delta u(x,z')\ dz' \qquad (45)$$

and

$$I\Delta u2(x) = \int_{z}^{z_{n+1}} \Delta u(x,z')\ [z'-z_n]/\Delta z\ dz' \quad . \qquad (46)$$

Collecting the terms $P\_(x,z_{n+1},\omega)$ and $P_1(x,z_n,\Delta z,\omega)$ we have

$$P\_(x,z_{n+1},\omega)\ [1 - i\omega\ I\Delta u2(x)] = P_1(x,z_n,\Delta z,\omega)\ [1 - i\omega\ I\Delta u2(x) + i\omega\ I\Delta u(x)] \quad . \qquad (47)$$

Again, realizing that we have neglected terms on the order of $\Delta u^2(x,z)$, equation 47 can be rewritten as:

$$P\_(x,z_{n+1},\omega)\ e^{-i\omega I\Delta u2(x)} = P_1(x,z_n,\Delta z,\omega)\ e^{i\omega[I\Delta u(x) - I\Delta u2(x)]} \quad . \qquad (48)$$

or simplifying:

$$P_{-}(x,z_{n+1},\omega) = P_1(x,z_n,\Delta z,\omega)\, e^{i\omega\, I\Delta u(x)} \quad . \tag{49}$$

If we again assume that the region between z_n and z_{n+1} has no vertical slowness variation, the integrals of equation 45 reduce to the product $\Delta u(x)\Delta z$ and equation 49 is again equivalent to equation 41 which defines the split-step Fourier method.

In general we can always evaluate the integral $I\Delta u(x)$ for a known slowness perturbation. For example consider that between depths z_n and z_{n+1} we have two slownesses which are separated by an interface which makes an angle θ with respect to the vertical. We can describe the perturbation of the slowness, $\Delta u(x,z)$ by:

$$\Delta u(x,z) = (u_2-u_1)\, H[x-x_1-\alpha(z-z_n)] + u_1-u_0 \tag{50}$$

where $-\infty < x < \infty$ and $z_n \le z \le z_{n+1}$. In equation 50 u_1, u_2 and x_1 are defined by Figure A1, α is defined as the tangent of the angle θ of the interface and H is the heavyside function:

$$H(x) = \begin{cases} 0, & x < 0 \\ 1/2, & x = 0 \\ 1, & x > 0 \end{cases} \tag{51}$$

Inspecting this equation for $\Delta u(x,z)$ we note that:

$$\Delta u(x,z) = \frac{1}{2}\,(u_2 + u_1) - u_0 \ , \tag{52}$$

for a point on the interface, which forms an angle θ with the vertical. Since we have chosen u_o to be the average slowness in the interval, $\Delta u(x,z) = 0$, at the interface.

We now write the integral, $I\Delta u(x)$ of equation 45 as:

$$\int_{z_n}^{z_{n+1}} \Delta u(x,z')\, dz' = (u_2 - u_1) \int_{z_n}^{z_{n+1}} H[x-x_1 - \alpha(z' - z_n)]\, dz' + (u_1 - u_0)\, \Delta z \ . \qquad (53)$$

To calculate the integral of equation 53, we will consider three distinct regions. The first region, R_1, is for $-\infty < x < x_1$, the second region, $R_{1,2}$, is for $x_1 \le x \le x_1 + \alpha\Delta z$ and the third region R_2 is for $x_1 + \alpha\Delta z < x < \infty$. With this subdivision we can easily calculate the integral for each region. In region R_1 because of the heavyside function we have:

$$\int_{z_n}^{z_{n+1}} \Delta u(x,z')\, dz' \ = \ (u_1 - u_0)\Delta z \ ; \ x \in R_1 \ . \qquad (54)$$

In region R_2, the heavyside function is always equal to 1 and we have:

$$\int_{z_n}^{z_{n+1}} \Delta u(x,z')\, dz' \ = \ (u_2 - u_0)\Delta z \ ; \ x \in R_2 \ . \qquad (55)$$

In region $R_{1,2}$, which is the transition region between the two constant slownesses, their variation is depth dependent and where:

$$z_n \le z' < (x-x_1) / \alpha + z_n \ , \qquad (56)$$

the heavyside function is always equal to 1, and where:

$$(x-x_1) / \alpha + z_n \ < \ z' \ \le \ z_{n+1} \ , \qquad (57)$$

it is equal to zero.The result for region $R_{1,2}$ is:

$$\int_{z_n}^{z_{n+1}} u(x,z')\, dz' = [(u_2 - u_1)\, (x-x_1) / \alpha] + (u_1 - u_0)\, \Delta z \ ; \qquad x \in R_{1,2} \ . \qquad (58)$$

This result can be generalized for the case of M interfaces corresponding to M + 1 different slowness regions between z_n and z_{n+1}:

$$\int_{z_n}^{z_{n+1}} \Delta u(x,z')dz' = \begin{cases} (u_1 - u_0)\,\Delta z\ ; & x \in R_1 \\ [\,(u_2 - u_1)\,(x - x_1)/\alpha\,] + (u_1 - u_0)\,\Delta z\ ; & x \in R_{1,2} \\ (u_2 - u_0)\,\Delta z\ ; & x \in R_2 \\ \quad\vdots & \\ (u_m - u_0)\,\Delta z\ ; & x \in R_m \\ [(u_{m+1} - u_0)\,(x - x_m)/\alpha] + (u_m - u_0)\,\Delta z\ ; & x \in R_{m,m+1} \\ (u_{m+1} - u_0)\,\Delta z\ ; & x \in R_{m+1} \\ \quad\vdots & \\ (u_M - u_0)\,\Delta z\ ; & x \in R_M \\ [(u_{M+1} - u_0)\,(x - x_M)/\alpha] + (u_M - u_0)\,\Delta z\ ; & x \in R_{M,M+1} \\ (u_{M+1} - u_0)\,\Delta z\ ; & x \in R_{M+1} \end{cases} \tag{59}$$

If we eliminate the transition regions, $R_{i,j}$, we are assuming that there are no vertical slowness variations in the depth interval between z_{n+1} and z_n and the result once again reduces to the original split-step Fourier method.

The error in the split-step Fourier method can be quantified based on the way the lateral slowness variations are decomposed into a reference slowness $u_0(z)$ and a perturbation $\Delta u(x, z)$ and the way this perturbation is taken into account. First, the split-step Fourier method assumes that the perturbation term is sufficiently small with respect to the reference slowness for the current depth extrapolation interval, Δz, so that terms on the order of $\Delta u^2(x, z)$ can be ignored. For depth extrapolation across intervals where no slowness variation exists, i.e., $\Delta u(x, z) = 0$, the method is exact and reduces naturally to the phase shift migration method. When the slowness does vary in the depth interval, the error in the method can be attributed to the neglect of the second order variations in the slowness and the higher order terms of equation 26 which are neglected in equation 27. Comparing equation 26 and 27, we find that this error, ε, as:

$$\varepsilon = i\omega \int_{z_n}^{z_{n+1}} dz' \int_{-\infty}^{\infty} dk_x' \, \Delta\tilde{u}\,(k_x - k_x', z')\, \tilde{P}_-\,(k_x', z', \omega) \;\cdot$$

$$\cdot \left\{ e^{i\omega u_0 [1-(k_x'/\omega u_0)^2]^{1/2} (z_{n+1}-z')} - \frac{e^{i\omega u_0 [1-(k_x/\omega u_0)^2]^{1/2} (z_{n+1}-z')}}{[1-(k_x/\omega u_0)^2]^{1/2}} \right\} \;. \qquad (60)$$

To find the order of this approximation we expand the two square root terms in each of the exponentials within the brackets, drop all terms greater than second order, and expand the denominator term. The result is:

$$e^{i\omega u_0 (z_{n+1}-z')} \left\{ e^{-1/2\, i\omega u_0 (k_x'/\omega u_0)^2 (z_{n+1}-z')} - e^{-1/2\, i\omega u_0 (k_x/\omega u_0)^2 (z_{n+1}-z')} \left[1 + 1/2\,(k_x/\omega u_0)^2 + \ldots\right] \right\} \qquad (61)$$

The leading exponential does not depend on k_x' and can be removed from the integral over k_x' in equation 60. Expanding the remaining exponentials and dropping all but the second order terms we have:

$$1 - 1/2\, i\omega u_0 (k_x'/\omega u_0)^2 (z_{n+1}-z') - \left[1 - 1/2\, i\omega u_0 (k_x/\omega u_0)^2 (z_{n+1}-z')\right] \left[1 + 1/2\,(k_x/\omega u_0)^2\right] \;, \qquad (62)$$

which reduces to:

$$- \frac{1}{2} (k_{x'}^2 - k_x^2) \frac{(z_{n+1} - z')}{\omega u_0} - \frac{1}{2} (k_x / \omega u_0)^2 \ . \qquad (63)$$

This reduces the error expression of equation 60 to:

$$\varepsilon = \int_{z_n}^{z_{n+1}} dz' \ e^{i\omega u_0 (z_{n+1} - z')} \int_{-\infty}^{\infty} dk_x' \ [(k_x'^2 - k_x^2) (z_{n+1} - z') / 2u_0$$

$$- \ ik_x^2 / 2\omega u_0^2 + 0(k_x^4) + 0(k_x'^4)] \ \Delta\tilde{u} \ (k_x - k_x', z') \ \tilde{P}_- \ (k_x', z', \omega) \qquad (64)$$

which shows that the second and higher order derivatives of the slowness perturbation will be neglected.

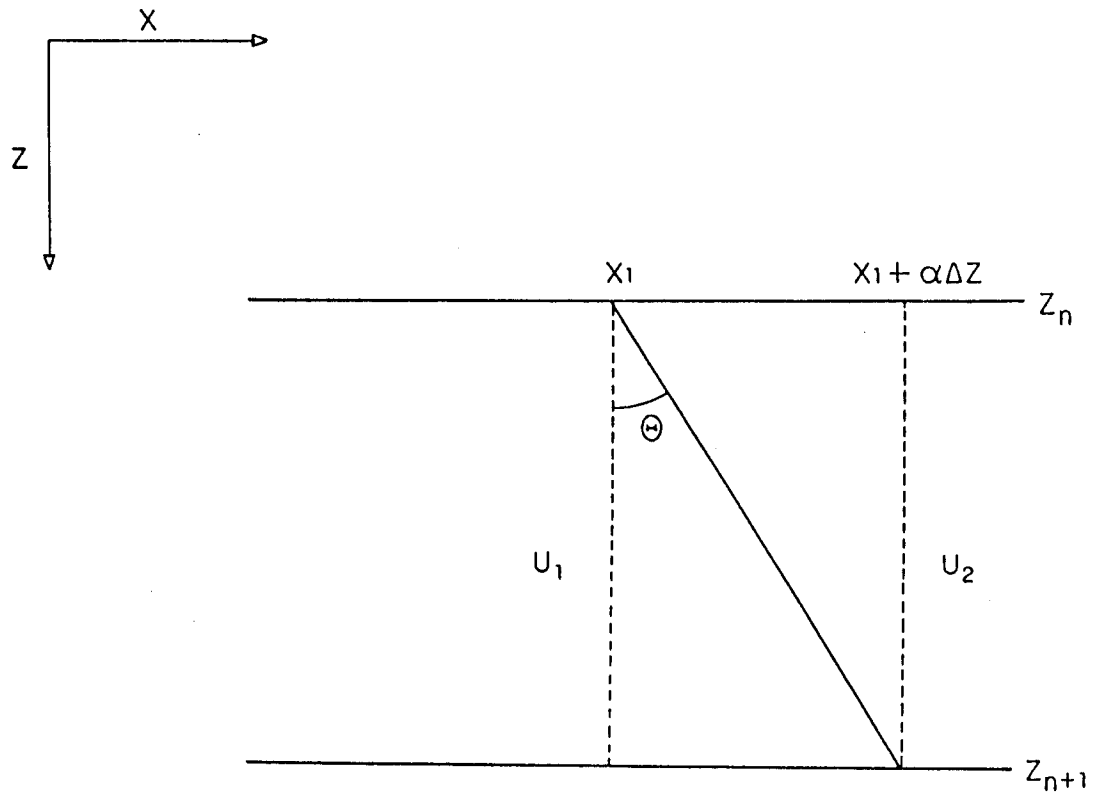


Figure A1. Geometry that describes a horizontal variation in slowness. To the left of the interface which has an angle θ with the vertical, the slowness is u_1 and to the right the slowness is u_2. The thickness of the interval is $\Delta z = z_{n+1} - z_n$.