Stanford Exploration Project, Report 70, December 18, 1997, pages 1–35

Stanford Exploration Project, Report 70, December 18, 1997, pages 1–35

Short Note

Three-dimensional time-slice migration on the Connection Machine

Martin Karrenbach[1]

keywords: *diffraction, Fourier transform, three-dimensional, scattering*

INTRODUCTION

Three-dimensional migration of zero-offset data using a depth-variable velocity can be performed in one pass using Fourier transforms of time slices. The migration process is carried out entirely in the two-dimensional spatial Fourier domain. Three-dimensional time-slice migration is inherently a parallel algorithm and maps perfectly onto the natural layout of processors on the Connection Machine. This method is especially suitable for target-oriented migration of a subset of a three-dimensional data volume.

The exploding reflector concept leads to a model that enables migration to be described as a surface integral. The basic step in this formulation is the summing of energy over the diffraction surface of a scattering point and placing the result at the apex of that conic. Thus migration causes the energy which was spread over the surface to be concentrated at one point. The way in which this is achieved determines efficiency and accuracy. This paper shows an implementation of the algorithm on a massively parallel computer, for which the algorithm is well suited.

Integral formulation

The theory of the principal of diffraction summation can be derived using the inhomogeneous acoustic wave equation (Schneider, 1978; Claerbout, 1986). The simplified solution to the wave equation in an arbitrary volume is given by the Kirchhoff integral. A wavefield D can be expressed at any time and at any point as a surface integral over the measured data $D(x_0, y_0, z_0, t)$. Here x_0, y_0, z_0 denote the coordinates of a coincident source-receiver pair, and S is the surface on which the data were recorded.

Evaluating the extrapolated recorded wavefield for all depths at time zero results

[1]**email:** not available

in $D(x,y,z,0)$, a three-dimensional image of the subsurface:

$$D(x,y,z,0) = -\frac{1}{2\pi}\frac{\partial}{\partial z}\int_S\int \frac{D(x_0,y_0,z_0,t=\frac{R}{V})}{R}dx_0dy_0, \tag{1}$$

where $R = 2\{(x-x_0)^2+(y-y_0)^2+(z-z_0)^2\}^{\frac{1}{2}}$. This is the final formula used to migrate 3D zero-offset seismic data. Therefore, in order to image the subsurface, the first step is to carry out the integration over the diffraction surface. Let x,y,z be Cartesian coordinates with the origin at the surface of the earth and the z-axis pointing vertically downwards. A single scattering point (x_1,y_1,z_1) in the subsurface has an arrival time T for a coincident shot and receiver,

$$T \;=\; \{T_0^2+\frac{4}{V^2}\left((x-x_1)^2+(y-y_1)^2\right)\}^{\frac{1}{2}}, \tag{2}$$

where V is the velocity of propagation and T_0=$\frac{2z_1}{V}$. Equation (1) represents the mapping process of a point in the spatial domain (x,y,z) into a surface in the space-time domain (x,y,t). If V is a function of T_0 only, the diffraction surface is a surface of revolution. The time slice T=*const* cuts the diffraction surface in a circle with center at (x_1,y_1) and with radius $R = \frac{V}{2}\{T^2-T_0^2\}^{\frac{1}{2}}$. In mapping from one domain to the other, scattering points are replaced by circles. The convolution which replaces points by circles can be expressed in the frequency domain by a multiplication with a cosine filter.

SUMMATION OVER THE DIFFRACTION SURFACE

In order to sum along the diffraction surface of a scattering point, the energy can be summed along circles in the time-slice representation of a 3D data volume. This process can be approximated by multiplying the 2D Fourier transform of each time slice by a cosine filter, corresponding to the radius of the intersection of the time slice with the diffraction surface.

The spatial Fourier domain is computationally attractive, since convolution in space reduces to a simple multiplication in the spatial frequency domain. The first task is then to perform a 2D Fourier transform of all the time slices. For that I used the CMSSL library on the Connection Machine, which includes an optimized multidimensional FFT subroutine. Figure **??** shows the layout of the transformed time slices and their filters. The 3D data cube has now one serial (temporal) dimension and two parallel axes, namely k_x and k_y (spatial frequencies). Each spatial frequency in a time slice is handled by a separate processor, and all the spatial frequencies are calculated independently from each other. For each spatial frequency component the values for all times and three filter coefficients are stored in local memory. No interprocessor communication is necessary to migrate each frequency component. All filtered time-slice components that are summed, come directly from each processor's local memory. Thus the spatial frequency domain maps perfectly onto the Connection

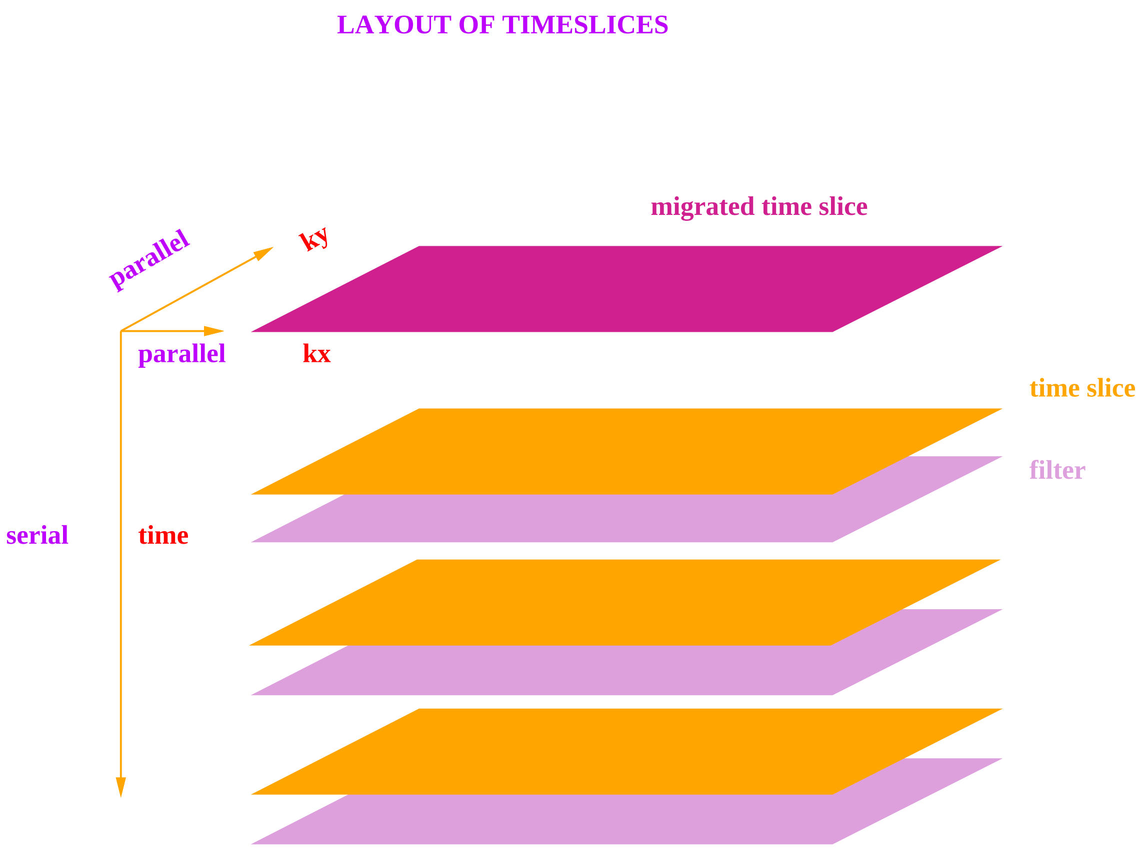

Figure 1: The spatial frequencies of individual time slices and filters are laid out on the Connection Machine in parallel. The time axis is kept as a serial dimension. No interprocessor communication is necessary to filter and sum individual spatial frequency components. martin2-cmlayout [NR]

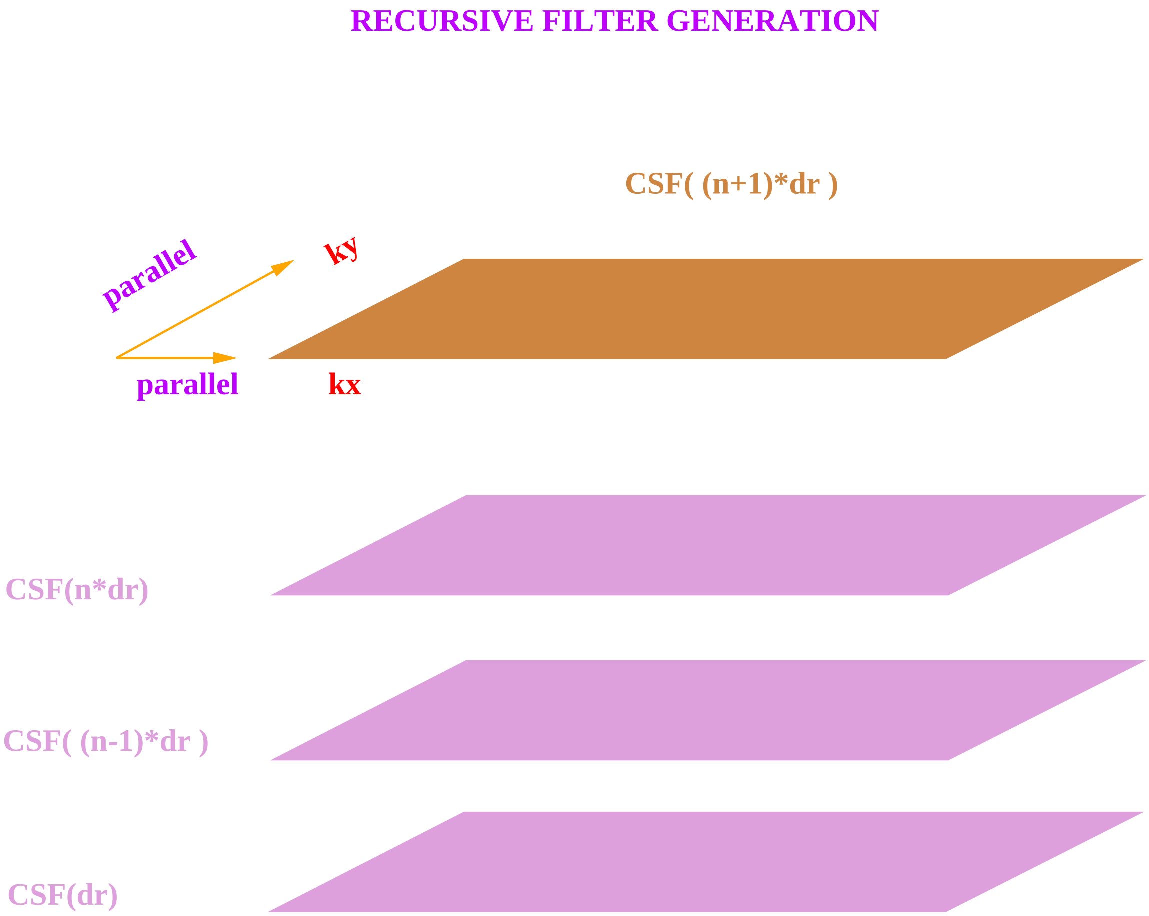

Figure 2: Time-slice filters $CSF((n+1) * dr)$ are generated by recursion as needed. This can be done for each spatial frequency independently. No interprocessor communication is necessary. martin2-filter [NR]

Machine layout. The summation over the diffraction surface proceedes along the serial time axis, an is carried out for all spatial frequency components simultaneously.

The pseudocode for migrating time slices is

- 2D FFT of all time slices to be summed
- For each increment in diffraction radius

```
{
Generate time-slice filter CSF
Interpolate time slice
Filter time slice
Sum into migrated slice
}
```

- Inverse 2D FFT of migrated time slices

The most efficient way to obtain the filter values for each slice that is summed, is to apply the trigonometric recursion

$$CSF_{n+1} = 2\ CSF_n\ CSF_1 - CSF_{n-1}, \qquad n > 1, \tag{3}$$

where the subscript is a multiplication factor in the argument of CSF, such that $CSF_i = CSF(\ i\ DR\ 2\pi\sqrt{k_x^2 + k_y^2}\ )$, with the initialization $CSF_0 = 1$ and $CSF_1 = CSF(\ DR\ 2\pi\sqrt{k_x^2 + k_y^2}\ )$. Thus each processor can calculate the new filter coefficient for each slice which is summed, with just three-floating point operations. Starting the summation at the vertex of the diffraction surface is advantageous in that the migration aperture does not have to be fixed at the beginning of the process. One can use this for an interactive session where the migration aperture can be chosen by inspection. The algorithm, as implemented now, is running typically at a speed of 190 Mflops on a 4k processor machine. It seems to me fast enough to be used as a tool for a very simple interactive 3D migration velocity analysis.

CONCLUSIONS

The time-slice migration scheme shown here sums energy over the diffraction surface of scattering points by filtering time slices with an appropriate operator which replaces points by circles. The filtered time slices are then added into the migrated time slice. After the summation procedure is complete, the entire time slice is migrated, not just one point in it. The algorithm maps well onto a massively parallel computer like the Connection Machine.

ACKNOWLEDGMENTS

Thanks to Biondo for his patience in showing me how to get the most speed out of the code. I was very lucky that Biondo was around, when I took my first steps in the "parallel" direction.

REFERENCES

Claerbout, J.F., 1986, Imaging the Earth's Interior: Blackwell Scientific Publications, Inc.

Karrenbach, M., 1990, Three-dimensional time-slice migration: Geophysics, **55**, 10-19.

Schneider, W.A., 1978, Integral formulation for migration in two-and three- dimensions: Geophysics, **43**, 49-76.