Stanford Exploration Project, Report 111, June 9, 2002, pages 231–238

Short Note

Speeding up wave equation migration

Robert G. Clapp[1]

INTRODUCTION

Wave equation migration is gaining prominence over Kirchhoff methods both as an imaging tool (Biondi and Palacharla, 1996) and for velocity analysis (Clapp, 2001). The relatively high cost of downward continuation methods, especially in iterative schemes (Biondi and Sava, 1999; Prucha et al., 2000), has limited their adoption.

In this paper I discuss three different methods to speed up midpoint-offset domain downward continuation based migration. I compare the migration results with the more standard downward continuation method. I show that a factor of two speed up is achievable with little discernible loss in image quality. In addition I show that for velocity analysis purposes a factor of three to four speed up is achievable.

THEORY

When doing downward continuation in the offset domain, we begin by organizing our data cube as a function of midpoint x, offset h, and frequency f. We then apply the double square root (DSR) equation to move the wavefield down one depth step Δz (Claerbout, 1995). We apply an imaging condition, and then repeat the procedure. This methodology can be quite expensive even in 2-D because the cost C is approximately

$$\begin{aligned} C &\approx nz * f * (FFT(nx,nh) + nx * nh * CEXP) \\ C &\approx nz * f * (nh * nx log(nh) + nx * nh log(nx) + CEXP(nx,nh)), \end{aligned} \tag{1}$$

where $FFT(nx,nh)$ is the expense of doing a 2-D FFT on a nx by nh dataset and $CEXP$ is the cost of multiplying by a complex exponential. In 3-D the cost is even more substantial.

Equation (1) indicates that the number of depths can greatly affect the cost of the migration. As a result, the choice of depth sampling is a major decision. Too fine a depth sampling will make the cost exorbitant; too coarse will cause resolution and aliasing problems.

[1]**email:** bob@sep.stanford.edu

The required depth sampling is also depth varying. We need finer depth sampling near the surface, while coarser depth sampling is appropriate in the deeper section. So the first obvious way to speed up our migration is to vary the sampling as a function of depth.

Sampling the wavelet

If we put in a time function, we can charactize the wavelength λ as it travels through the earth with

$$\lambda = \frac{v}{f}, \tag{2}$$

where f is the frequency and v is the velocity. In practical terms, this says that our input is a time function that travels in a given interval as a function of the media's velocity, and that different frequencies will travel at different velocities. What is important is that we adequately sample the wavelet. As a result, we can take large depth steps at large velocities and small frequencies. Our migration cost is then generally concentrated at higher frequencies and shallower depths.

Attenuation

Our second observation is that all signals eventually attenuate, and this attenuation (Q) is a function of frequency and media properties. How much they attenuate is generally a function of the media, but a decent first approximation is that after a certain number of cycles, a given frequency attenuates a given percentage(Kjartansson, 1979). Therefore, after a certain number of cycles, the wave components at that frequency will no longer be of usable strength so we can stop downward continuing it. This observation nicely compliments the wavelet sampling observation that higher frequencies will attenuate quicker in depth, reducing the distance we need to downward continue them. We can combine these two ideas and replace our original cost equation with,

$$C \approx \sum_{f} * nz(f) * (nh * nxlog(nh) + nx * nhlog(nx) + CEXP(nx, nh)), \tag{3}$$

where our number of depth steps $nz(f)$ is a function of frequency.

Focusing energy

Our imaging condition provides a third idea for decreasing cost. As we go down in depth, energy focuses at zero offset. The outer offsets will either have no energy, or energy that we don't care about. As we step down in depth, we can decrease our offset domain. With this final savings, we can rewrite our cost function as:

$$C \approx \sum_{f} \sum_{nz(f)} nh(z) * nxlog(nh) + nx * nh(z) log(nx) + CEXP(nx, nh(z))). \tag{4}$$

PRACTICAL ASPECTS

Applying the previous cost saving ideas introduces some new challenges. First, our final image is going to be regularly sampled so we must resample our variably sampled wavefield. I chose Lagrange interpolation for my resampling. It has the advantage of allowing higher order interpolation than simple linear interpolation, and you can easily pre-build interpolation tables, making it fast.

The second thing we have to be concerned with is the speed of our FFT. We generally use FFTW (Frigo and Johnson, 1999) which can handle any size axes but transforming an axis of length 512 will be significantly faster than transforming an axis of length 511. The easiest way to handle this problem it to make a list of 'good' axis lengths (e.g., combination of prime factors and power of two) and only decrease the offset domain at depths where the next lowest 'good' axis length is reached.

Finally, when this process is parallelized to be run on a linux cluster load, balancing becomes more difficult. Each frequency block has a different number of depth steps, a different interpolation cost, and a fixed IO cost. Ensuring that each processor finishes at approximately the same time is a non-linear optimization problem. Presently I assign costs for depth steps, interpolation, and IO and then try 1000 different random solutions, selecting the one that shows the least cost differential between the nodes.

RESULTS

To test how much the migration could be sped up I chose a 2-D line from a 3-D land dataset provided by Ecopetrol. Figure 1 shows the result using the conventional approach. The left plot shows the zero offset image (the standard imaging condition), the right panel shows three selected angle CRP gathers (Sava and Fomel, 2000). Using five reference velocities and four processors it took 2034 seconds to run.

Figure 3 shows the same migrated image calculated with variable depth sampling, accounting for attenuation, and reducing the size of the offset domain as we go down in depth. The left panel of Figure 2 shows the sampling in depth for several frequencies. The right panel shows how the offsets downward continued decrease as the depth increased. The result was achieved in 1042 seconds (almost half of the time) and is nearly identical to the result in Figure 1.

For velocity analysis the image quality requirements are reduced. By using the sampling in Figure 4 image gathers can be produced that give accurate moveout information (Figure 5) while further reducing the migration time to 758 seconds.

FURTHER SPEEDUP

In theory it should also be possible to use sparser sampling in midpoint at low frequencies. Since this would only affect the relatively inexpensive portion of the downward continuation,

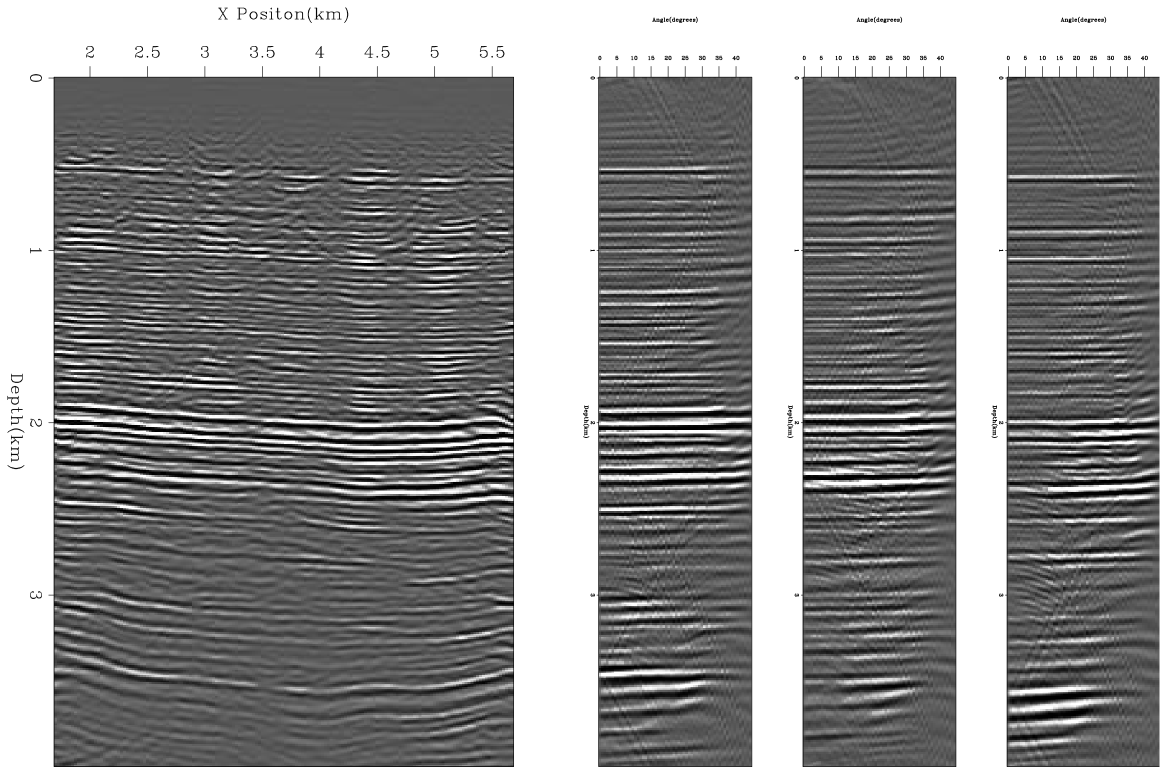

Figure 1: The left panel is the result of split-step migration. The right panel is three gathers from the same migration. bob3-mig-slow [CR]

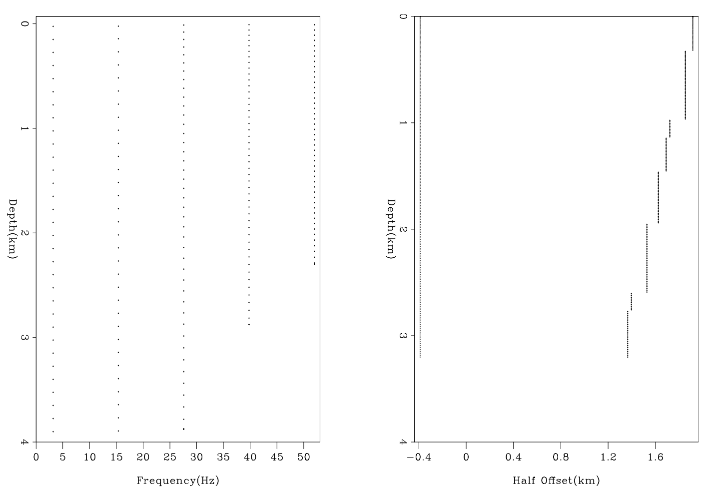

Figure 2: The left panel shows every tenth depth for various frequencies for the migration show in Figure 3. The right panel shows the reduction in the migrated offset domain as a function of depth. Note how the sampling in depth sparser, the offset domain is decreased quicker, and the frequencies are assumed to be of inconsequential energy than in Figure 3. bob3-fast-sample [CR]

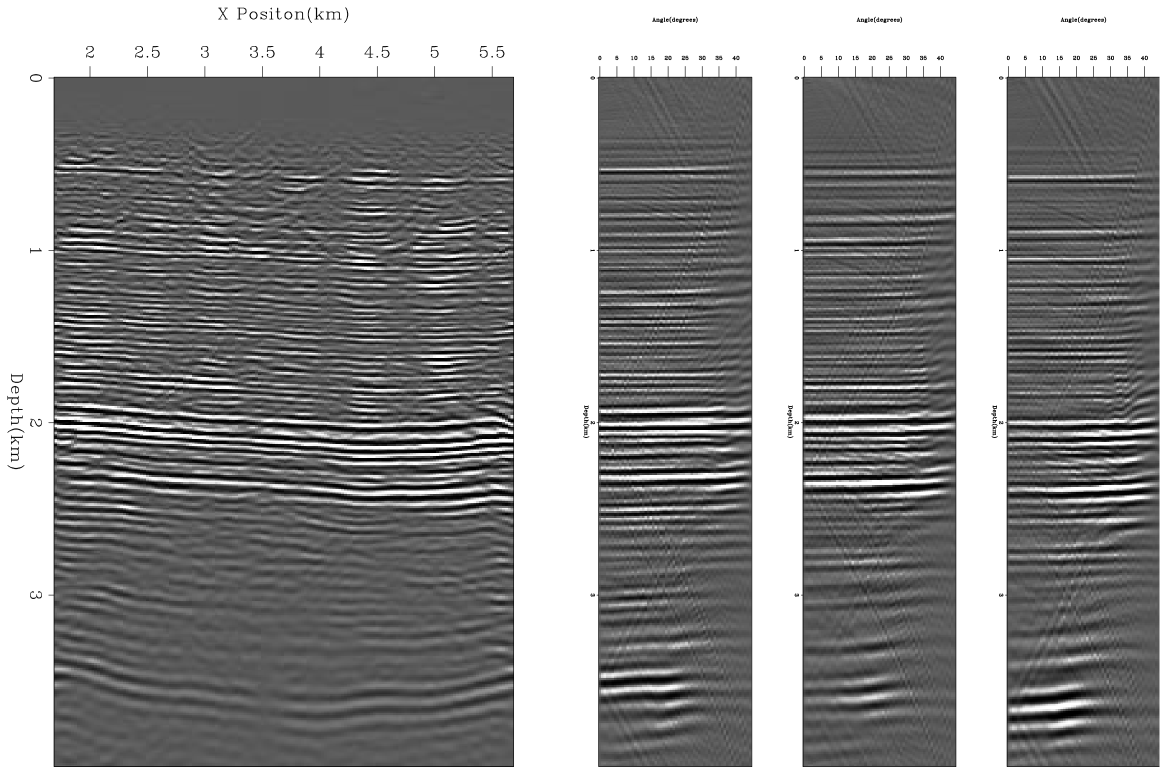

Figure 3: The left panel is the result of split-step migration. The right panel are three gathers from the same migration. Note how the image is almost identical to Figure 1 but is calculated three times faster. bob3-mig-fast [CR]

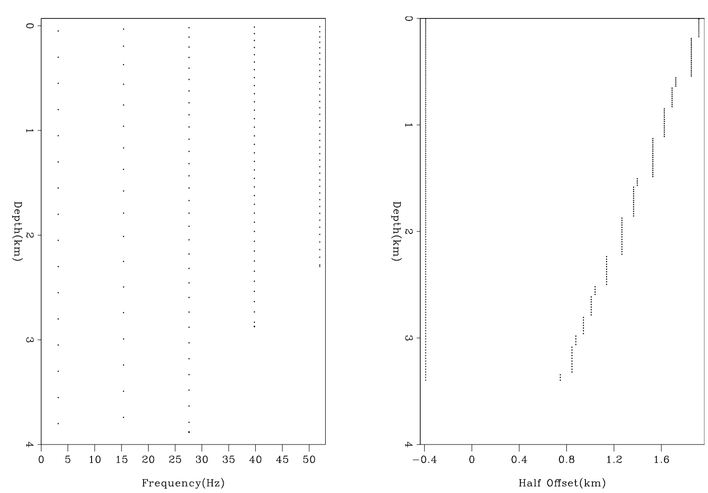

Figure 4: The left panel shows every tenth depth for various frequencies. The right panel shows the reduction in the migrated offset domain as a function of depth. Note how the sampling in depth sparser, the offset domain is decreased quicker, and the frequencies are assumed to be of inconsequential energy than in Figure 5. bob3-faster-sample [CR]

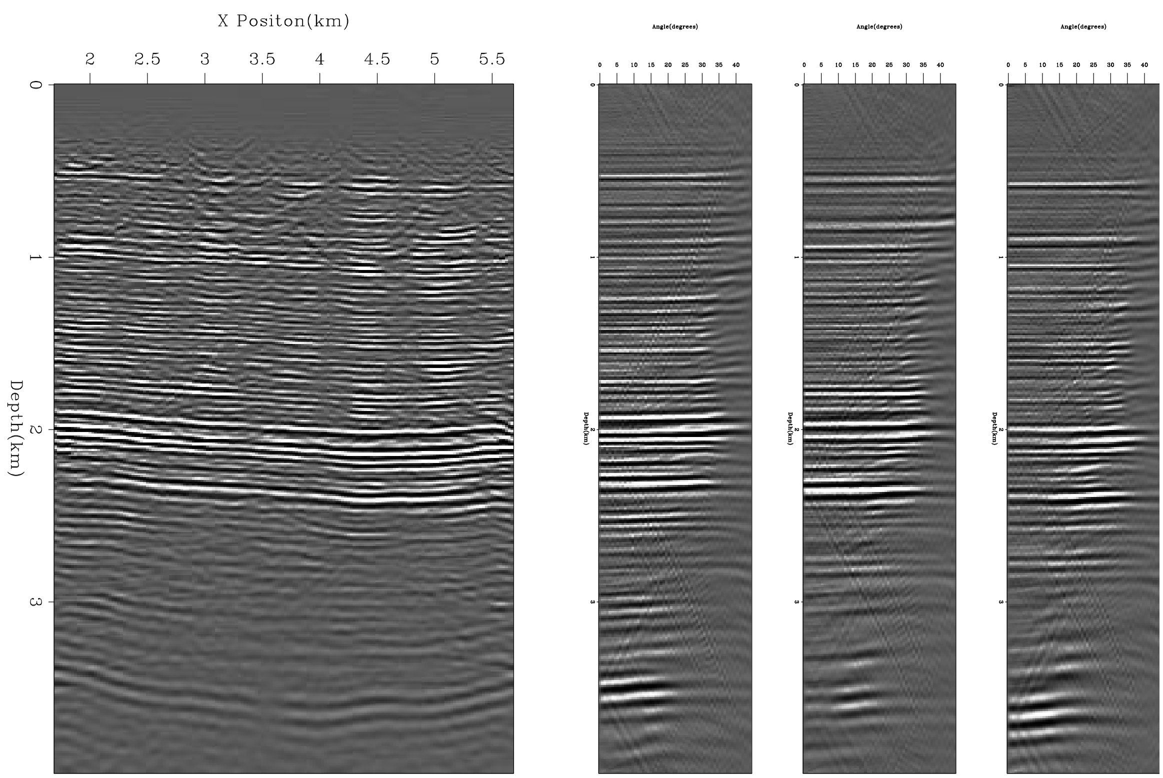

Figure 5: The left panel is the result of split-step migration. The right panel is three gathers from the same migration. Note how the image is slightly different from Figures 1 and 5, but the moveout information is nearly identical. bob3-mig-faster [CR]

I would expect at most only a $10-20\%$ speed up. Another method is to do a better job choosing appropriate velocities, thereby reducing the number of velocities needed for accurate downward continuation. Clapp (2002) discussed one method to accomplish this.

CONCLUSIONS

Wave equation migration can be significantly sped up by making depth steps a function of frequency and velocity while the offset domain can be reduced as we go down in depth. Wave equation migration can be sped up by a factor of three, and acceptable migration gathers for velocity analysis can be generated five times quicker than conventional migration methods.

ACKNOWLEDGMENTS

I would like to thank Ecopetrol for the data used in this paper.

REFERENCES

Biondi, B., and Palacharla, G., 1996, 3-d prestack migration of common-azimuth data: 3-d prestack migration of common-azimuth data:, Soc. of Expl. Geophys., Geophysics, 1822–1832.

Biondi, B., and Sava, P., 1999, Wave-equation migration velocity analysis:, *in* 69th Ann. Internat. Mtg Soc. of Expl. Geophys., 1723–1726.

Claerbout, J. F., 1995, Basic Earth Imaging: Stanford Exploration Project.

Clapp, R. G., 2001, Geologically constrained migration velocity analysis: Ph.D. thesis, Stanford University.

Clapp, R. G., 2002, Reference velocity selection by a generalized Lloyd method: SEP–**111**, 215–225.

Frigo, M., and Johnson, S. G., 1999, FFTW: `http://www.fftw.org/`.

Kjartansson, E., 1979, Attenuation of seismic waves in rocks and applications in energy exploration: Ph.D. thesis, Stanford University.

Prucha, M. L., Clapp, R. G., and Biondi, B., 2000, Seismic image regularization in the reflection angle domain: SEP–**103**, 109–119.

Sava, P., and Fomel, S., 2000, Angle-gathers by Fourier Transform: SEP–**103**, 119–130.