Seismic imaging on massively parallel computers

Curtis C. Ober, Ron A. Oldfield, David E. Womble, Sandia National Laboratory*
Charles C. Mosher, ARCO Exploration and Production Technology

Summary

A key to reducing the risks and costs of associated with oil and gas exploration is the fast, accurate imaging of complex geologies, such as salt domes in the Gulf of Mexico and overthrust regions in U.S. onshore regions. Pre-stack depth migration generally yields the most accurate images, and one approach to this is to solve the scalar-wave equation using finite differences. Current industry computational capabilities are insufficient for the application of finite-difference, 3-D, prestack, depth-migration algorithms. High performance computers and state-of-the-art algorithms and software are required to meet this need. As part of an ongoing ACTI project funded by the U.S. Department of Energy, we have developed a finite-difference, 3-D prestack, depth-migration code for massively parallel computer systems. The goal of this work is to demonstrate that massively parallel computers (thousands of processors) can be used efficiently for seismic imaging, and that sufficient computing power exists (or soon will exist) to make finite-difference, prestack, depth migration practical for oil and gas exploration.

Introduction

Oil and gas companies search continually for new reserves; however, most of the easy discoveries have been made. Many of the remaining possible sites for drilling are in regions of complex geologies, such as salt domes or overthrust regions. These complex geologies are characterized by large velocity variations, which tend to obscure deep images produced through seismic imaging. These images could indicate the presence of oil and gas.

To image beneath these complex geologies, more accurate imaging techniques are required. Current imaging techniques use ray-tracing schemes (i.e. Kirchhoff migration) or trace-averaging schemes (i.e., poststack migration), which can have difficulty with large, 3-D, velocity variations. To handle these velocity variations, we have developed a wave-equation, 3-D, prestack, depth-migration code, called Salvo. Although the wave-equation approach is not new, it requires substantial computational power and time to produce an image. These high costs have prevented its wide-spread use in the industry.

Massively Parallel Processor (MPP) computer systems can provide the needed computational power for these highly accurate imaging techniques. Several problems have been addressed in order to obtain an efficient code for MPP systems. These include efficient I/O, high single-node performance, and efficient parallel tridiagonal solves. Furthermore, Salvo has been restricted to high-level programming languages (C and Fortran) and standard interprocessor communications (MPI) to provide portable code.

In the following sections, the approaches used to obtain an efficient prestack migration code are presented, including I/O management for the trace reads, computations, parallel optimizations, and analysis of performance.

Input / Output Management

Efficient I/O is an important aspect of seismic imaging, because the datasets consisting of recorded pressure waves are often large. Even if the computations can be performed in-core, the time required to read the initial seismic data, read the velocity models and write the images can be substantial. In Salvo, the "I/O bottleneck" is mitigated by performing preliminary computations and data redistribution during the I/O intensive phases, and by assigning a small number of nodes to coordinate I/O during the compute intensive phases.

The dataset contains a sequence of traces, and is distributed across many disks to increase the total disk-to-memory bandwidth. A single node is assigned to handle the I/O for each file system, and is termed an I/O node. This is an effective arrangement for the 1000 processor Intel Paragon at Sandia National Laboratory, which uses a number of parallel disk arrays for I/O services.

The remaining nodes, termed compute nodes, can complete necessary computations and communications before the migration begins. Each compute node is assigned to an I/O node, and performs the pre-computations on the data read by its I/O node. Currently the pre-computation comprises fast Fourier transforms (FFTs), but other computations could also be performed. If a sufficient number of compute nodes are assigned to each I/O node, the time to read a block of seismic data will be greater than the time required to compute the FFTs and distribute the frequencies to the correct nodes for later computations. Thus, the computation time will be hidden behind the I/O time.

The I/O performance was measured by distributing a 150 MByte data file to 32 disks and then reading the data of the disks and measuring the time to read, Fourier transform and distribute frequencies for the traces stored on those disks. By using this method, we were able to accurately measure the scalability of Salvo as the number of disks increases. A run using 32 disks reads and writes data from disk at an aggregate bandwidth of approximately 48 MBytes/sec, which is about

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94AL85000.

75% of the peak bandwidth.

During the computations, the velocity data must be read from the disk, image data must be written to disk, pressure data must be written to disk for restarts, and if the problem is too large to fit in memory, frequency data must be swapped in an out. We perform these I/O operations in much the same way the trace data I/O is handled. A few nodes are allocated to perform I/O and do all of these operations in the background of the compute nodes. In addition to performing I/O, these nodes also interpolate the velocity model and stack and bin the image as it is being created. I/O for seismic processing can take as much as twenty percent of the overall time for a given run. If that twenty percent can be hidden, the savings can be tremendous.

During the computations, velocity data must be read from the disk, image data must be written to the disk, pressure data must be written to disk for restarts, and if the problem is too large to fit in memory, frequency data must be swapped in and out. We currently perform this I/O from within the compute processors, but we are rewriting the code to use a subset of the compute nodes to coordinate I/O in a manner similar to how the trace data I/O is handled.

Imaging Algorithm

After the input and preprocessing phase, migration and imaging are performed. The approach used in Salvo is based on industry-standard approaches for shot record migration utilizing frequency domain implicit solutions to the scalar wave equation (Claerbout 1985, Yilmaz 1987, Li 1991), summarized here for reference. The equation used in Salvo to model the propagation of pressure waves through the earth is based on a paraxial approximation to the scalar wave equation:

$$\frac{\partial P}{\partial z} = \pm\frac{i\omega}{v(x, y, z)}\left(\frac{\alpha S_x}{(1+\beta S_x)}+\frac{\alpha S_y}{(1+\beta S_y)}\right)P \qquad (1)$$

where ω is angular frequency, P is pressure as a function of (x,y,z,ω), $v(x,y,z)$ is the acoustic velocity of the media, α and β are expansion coefficients (Lee and Suh, 1985), and $S_{x,y}$ are partial derivative operators

$$S_{x, y} = \frac{v^2}{\omega^2}\left(\frac{\partial^2}{\partial(x, y)^2}\right) \qquad (2)$$

The positive and negative signs correspond to upcoming and downgoing wave fields.

Filters (i.e. Graves and Clayton, 1990) are also incorporated to correct for errors introduced by the square-root operator and operator-splitting approximations. Absorbing boundary conditions similar to those of Clayton and Engquist (1980) are used. Note that both Equation 1 and the correction filters involve solutions of large numbers of tridiagonal systems, which is addressed in a later section on performance.

In the shot record imaging procedure, source and receiver wavefields are downward continued using Equation 1. The source wavefield is produced by introducing an estimate of the source wavelet into a uniform computational grid at the surface (x,y) location of the source. The receiver wavefield is interpolated onto a second grid of the same size. At each extrapolation step, downgoing waves are selected for the source wavefield, upgoing waves for the receiver wavefield. An image is produced at each depth level by selecting the zero time cross-correlation of the source and receiver wavefields. Alternatively, the receiver wavefield can be deconvolved with the source wavefield, which requires more accurate estimates of the source wavelet.

The impulse response of the Salvo solution is shown in Fig. 1.

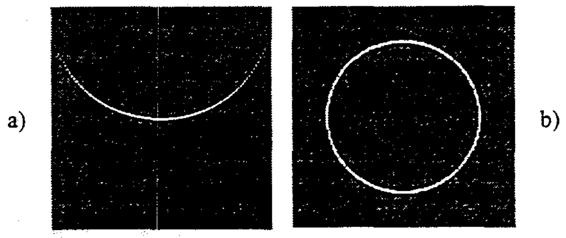

Fig. 1: Salvo impulse response test

The inputs for this test are a source trace with an impulse at some time, a receiver trace with an impulse at some later time, and a constant velocity field. The shape of the impulse response is close to circular up to a propagation angle of 65 degrees. Beyond 65 degrees, the image falls off gradually, amounting to a high cut filter for steeply dipping events. The cross-section of the impulse response is nearly circular, which should be the case since this is within the 65 degree approximation limits.

As an example of application to a pre-stack 3D seismic dataset, a small region of the synthetic SEG/EAEG Overthrust Model (Aminzadeh et al, 1994) was used. This model contains moderate variations in velocity, both in depth and in the horizontal directions. The velocity model for the entire Overthrust Model has 801 x 801 x 187 grid points with 25 m spacing in each direction. The selected subvolume has 100 x 100 x 150 grid points. A 2-D slice through the velocity model is shown in Fig.2 along with the generated image from Salvo. The image layers closely agree with the velocity model layers, indicating an acceptable solution.

Single-Node Optimization

To obtain good performance for the overall code, high single-node performance is required. Because our code is expected to be portable, we have restricted ourselves to the use of FORTRAN and C, which removes the possibility of assembly cod-

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, make any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

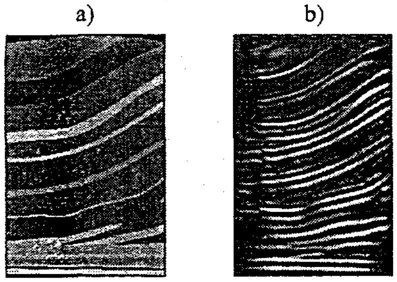

Fig. 2: Salvo image from SEG/EAG Overthrust Model

ing portions of the computational kernel. However, the modular design of the code allows vendor-optimized routines to be incorporated at a later date.

The complex exponentiation required for a solution to Eq. 1 is very expensive. To reduce the computation, Euler's formula is used to replace the complex exponential with sines and cosines. These sines and cosines are represented by a table of discrete sine values. Using the first two terms from a Taylor series, both the real and imaginary parts of the complex-exponential solution are approximated within desired accuracy. Sample runs have shown a 10% reduction in computation time by using the approximate complex exponential solution.

For the tridiagonal solves, several computational points have been addressed. First, tridiagonal solves are vector operations. Their performance is limited by the memory-bus bandwidth, because the expected size of our tridiagonal systems limits the effectiveness of cache usage. Thus, we have approached this problem by combining the coefficient-generation and tridiagonal-solve loops to reduce the demands on the memory bus. This code has achieved 30% or more of peak performance on many of the microprocessors used in modern workstations and parallel computers.

Parallel Optimization

Another problem that must be addressed is the efficient use of thousands of processors. There are several levels of parallelism available in a finite-difference solution of the wave equation for seismic imaging. At a coarse-grained level, many shots can be processed in parallel. Within a shot, each of the frequencies can be processed independently. Computing the final image requires only a global sum. This frequency parallelism is highly efficient. One problem is that only 100 to 500 frequencies are usually processed, and we cannot use more processors than frequency levels. Another problem is that each processor must store (or have access to) the entire velocity model. Some MPP systems have as few as 16 MBytes of memory per node, which limits the size of problem that can be solved.

A solution to this size limit has been to introduce spatial parallelism for each frequency plane, which means that tridiagonal solves must be paralellized. Parallelizing individual tridiagonal solves is difficult, so our approach has been to take advantage of the fact that each frequency plane represents many tridiagonal systems to be solved. This fact can be used to set up a pipeline.

In the first stage of the pipeline, processor one starts a tridiagonal solve. In the second stage of the pipeline, processor two continues the first tridiagonal solve, while processor one starts a second tridiagonal solve. This process continues until all processors are busy.

In the implementation of a pipeline, there are two sources of parallel inefficiency. The first is communication between processors. This communication time is dominated by the message latency since very small amounts of data must be transferred. This can be offset by grouping several tridiagonal solves into each stage of the pipeline.

The second source of parallel inefficiency is processor idle time associated with the pipeline being filled or emptied. This is dominated by the computation time of each pipeline stage. It can be reduced by reducing the computation time, but it is increased by grouping several tridiagonal solves in each stage of the pipeline. The total parallel overhead can be minimized by choosing how many tridiagonal solves are grouped into each stage of the pipeline.

Performance

To test the computational performance of Salvo, an impulse problem was used. The spatial size of the impulse problem has been adjusted so that each processor has approximately a 101 x 101 spatial grid. Therefore if four processors were used in the x-direction, and four processors were used in the y-direction, the total domain was 401 x 401. Sixty-four frequencies were retained for the solution independent of how many frequency processors were used. Thus if four processors were used in the ω-direction, sixteen frequencies would be migrated per processor.

Timings for a sample impulse run are shown in Table 1 and in Table 2. These results show that spatial parallelism is

Table 1: Spatial Parallelism

Size	Time, sec	Efficiency,%
1x1x1	84.1	100.0
2x1x1	92.4	91.0
2x2x1	103.2	81.5
3x3x1	108.7	77.4
4x4x1	108.9	77.2
5x5x1	112.2	75.0
6x6x1	114.8	73.3
7x7x1	115.6	72.8
8x8x1	116.2	72.4

very efficient as soon as the pipeline includes three or more processors. However there is a penalty for introducing the pipeline in each direction, which is about 10% for each dimension.

Table 2: Frequency Parallelism

Size	Time, sec	Efficiency, %
1x1x1	84.1	100.0
1x1x2	42.21	99.6
1x1x4	21.19	99.2
1x1x8	10.63	98.9
1x1x16	5.35	98.2
1x1x32	2.71	97.0
1x1x64	1.40	93.8

Second, the frequency parallelism is very efficient, staying in the upper 90's for most of the problems. This is expected, since frequency parallelism requires little communication during the solve. The primary communications are a broadcast of velocity data at the beginning of each depth step and a summation to produce an image at the end of each depth step.

A more complete test of the overall machinery and accuracy of Salvo was conducted using a subset of the SEG/EAEG Salt model (Aminzadeh et al, 1994). 45 shots were selected, each covering a 200 by 200 grid point subvolume, with a grid spacing of 20 m. The 45 shots were summed to produce a 600 by 600 grid point image, with a maximum fold of 15. A 2D slice through the computed image with the velocity model overlain is shown in Figure 3a, a 3D chair-cut display showing the imaged salt structure is shown in Figure 3b. For this calculation, the 45 shots were processed in 9 hours on 256 nodes of Sandia's Intel Paragon. Even with 45 shots, Salvo produces an excellent image of the salt structure, and accurately positions reflectors below the salt. Based on initial tests, performance projections for Salvo performance on Sandia's next-generation TeraFlop computing system exceed 500 shots per hour. At this rate, large 3D surveys could be processed with reasonable turnaround times, including multiple iterations for velocity analysis.

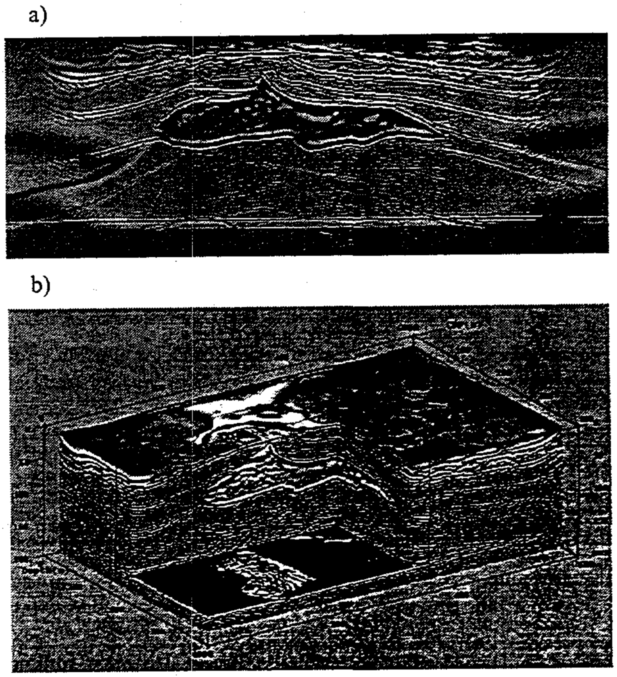

Figure 3: Salvo Images from SEG/EAEG Salt Model

Conclusions

We have presented approaches for efficient implementaion of wave equation shot record migration on massively parallel computers. These approaches include I/O partitioning to perform trace data reads in an asynchronous manner, optimizations of single-node performance, and multiple levels of parallelism. Frequency parallelism has high parallel efficiencies, while spatial parallelism has slightly lower efficiencies. The complete code provides high performance and accurate results for imaging standard industry 3D seismic models.

Acknowledgments

We would like to thank the Department of Energy and the participants in the Advanced Computational Technology Initiative Project "3-D Seismic Imaging of Complex Geologies" for their support of this work: Sandia National Laboratory, the University of Texas at Dallas, ARCO, Conoco, Oryx, Providence Technologies, Golden Geophysical, PGS Tensor, TGS Caliber, IBM, Intel, and SGI/Cray.

References

Aminzadeh, F., Burkhard, N., Nicoletis, L., Rocca, F., Wyatt, K., 1994, SEG/EAEG 3-D modeling project: 2nd update, Leading Edge, September, 949-952.

Claerbout, J.F., 1985, *Imaging the Earth's Interior*, Blackwell Scientific Publications, Boston.

Clayton, R., and Engquist, B., 1980, Absorbing boundary conditions for wave-equation migration, Geophysics, **45**, 895-904.

Graves, R., and Clayton, R., 1990, Modeling acoustic waves with paraxial extrapolators, Geophysics **55**, 1650-1660.

Lee, M.W., and Suh, S.Y., 1985, Optimization of one-way wave equations, Geohpysics **50**, 1634-1637.

Li, Z., 1991, Compensating finite-difference errors in 3-D migration and modeling, Geopysics **56**, 1650-1660

Yilmaz, O., 1987, *Seismic Data Processing*, Investigations in Geophysics No. 2., Society of Exploration Geophysicists, Tulsa.