Finite difference 3-D elastic and acoustic seismic modeling using Beowulf cluster

Bragança* , R.S.N. de; PETROBRAS S/A
Bulcão, A.; PEC - COPPE/UFRJ
Soares, Djalma M.F.; PETROBRAS S/A
Mansur, W.J.; PEC - COPPE/UFRJ

This paper was prepared for presentation at the 8th International Congress of The Brazilian Geophysical Society held in Rio de Janeiro, Brazil, 14-18 September 2003.

Abstract

In this work we present a complete performance analysis of the elastic and acoustic seismic modeling algorithms for 3-D seismic acquisition simulations and vectorial imaging (prestack elastic and acoustic depth migrations) based on a finite difference technique which divides the space domain in several parts. From our analysis it is possible to estimate the total amount of time a Beowulf cluster spent in 3-D modeling and reverse time migration accomplished in realistic situations.

Definitions

The following notation is used in this work:

L Number of machines in a parallel task

N Total Number of grid points in 3 dimensions

N Number of grid points in one dimension

$T1$ Total execution time using a serial code

T Total execution time using a parallel code

T^A Total execution time for an analysis of A seconds

T_L Total execution time using parallel code running in L machines

$Tproc$ Computation part of T

$Tcom$ Communication part of T

Ts Serial part of T (data I/O)

Tp Parallel part of T (computation + overhead)

From the definitions we can express the equations

$$T = Ts + Tp \quad (1)$$

$$Tp = Tproc + Tcom \quad (2)$$

Parallel processing basic concepts

- Speedup

Measures how much faster a parallel code can run compared to a serial one:

$$S = \frac{T1}{T_L} \quad (3)$$

$T1$ is the serial execution time and T_L the parallel execution time using L machines.

- Efficiency

The efficiency (E) of a parallel code can be expressed by the equation:

$$E = \frac{S}{L} = \frac{T1}{(L.T_L)} \quad (4)$$

The efficiency of a serial code is always 1 and a parallel code with efficiency 1 running in L machines is L times faster than serial.

- Granularity

The granularity (G) is an important property of parallel applications and is defined by:

$$G = \frac{T_{proc}}{T_{com}} \quad (5)$$

Low granularity means that you are spending much time in communications rather than in computation i.e. in the resolution of your task. Thus greater granularity is better in general but may difficult load balance.

Methodology

Finite difference was used to solve the differential equation of acoustic (Myczkowski,1991) and elastic (Ewing, 1994) wave propagation. Central operators of 2nd order in time and 4th order in space were used. Domain decomposition was used to split the problem physical domain in several partitions, which was sent and processed in different cluster nodes.

Such parallel structure was implemented in a master– slave fashion, using PVM for inter-node communications. Details of the adopted programming structure may be obtained in (Myczkowski, 1991) and (Levander, 1988).

All simulations was performed in the cluster SISMOS 3 at PETROBRAS Research Center, which characteristics are in table 1:

Table 1: Cluster Specifications

Number of nodes	72
Node interconnection	Fast Ethernet Switch
Node processor	Pentium III (Katmai) 550 MHz 512Kb cache
Node RAM memory	768 Mb
Hard disk system	SCSI RAID5 by NFS

The total execution time of modeling is proportional to the number of grid points and the time step used in analysis. In this work we tested cubic partitions so N = n3. It is a reasonable assumption that the processing time Tproc be proportional to N or Tproc. ∝ n3, besides, communication time Tcom is proportional to the interface areas between adjacent partitions so Tcom ∝ n2. Based in these assumptions we expect that the granularity may have a linear behavior with n.

To obtain the general behavior of the codes several tests were made considering homogeneous geologic models which partitions varying from 50 x 50 x 50 to 350 x 350 x 350 grid points for the acoustic case and from 100 x 100 x 100 to 240 x 240 x 240 grid points for the elastic one. The parameters used in all simulations can be found in table 2.

Table 2: Finite Difference modeling parameters

	Acoustic	Elastic
Grid spacing	12.5 m	6.25 m
Time step	0.0004 s	0.0002 s
Total analysis time	0.02 s	
Cut frequency	28 Hz	
Model	Homogeneous,V=3.0 Km/s	

The tests were performed considering a cubic partitioning scheme. Figure 1 shows the four partition groups: $(1)^3= 1$, $(2)^3= 8$, $(3)^3= 27$ and $(4)^3= 64$ partitions. Dimensions varying from 50 x 50 x 50 (1 partition) to 1400 x 1400 x 1400 (64 partitions) were acoustic modeled and dimensions varying from 100 x 100 x 100 (1 partition) to 960 x 960 x 960 (64 partitions) were elastic modeled.

Since in this particular work we are interested only in computation time, we used a little trick to speed up the analysis execution. The computations were performed for a reduced analysis time of 0.02s but making separate measurements of the model loading time Ts (which is independent of the analysis time) and the model computation Time Tp. The execution time T for a typical analysis of 5s may be expressed by:

$$T = T^{5.0} = Ts + 250 * T_P^{0.02} \quad (6)$$

Every time T appears in the text means $T^{5.0}$ or the total computer time spent in the modeling of a 5 seconds analysis.

To minimize some little variations in the obtained execution time, 3 analysis were performed for each model size and the computed average was considered. These variations are mainly due to the use of the CPU in other system tasks. We did not considered the time spent in seismogram and/or snapshots output in our analysis.

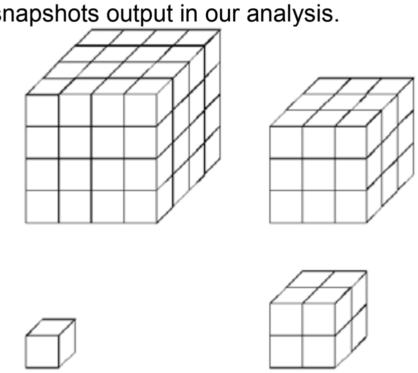

Figure 1: 1, 8, 27 and 64 partition models.

Results

Figure 2 presents a graph of the total execution time T function of the total calculated grid points N for each partition scheme (1, 8, 27 and 64 machines) for the acoustic modeling. The T x N relationship is linear and the corresponding fitting equations are in the same picture.

The T x N behavior for the elastic case is not linear as we can see in figure 3. The data was fitted using power curves and the adjusted equations are also within figure 3.

It is possible to note the absence of some points in the curves of figure 2. These points were considered “anomalous” because did not fit the data trend and were excluded from the curve fit. The criteria used to elect an anomalous point is the postulate that the execution time Tp always grows with N. In this way if Tp decrease with increasing N implies that the last point is an anomalous one and is not considered for the curve fitting. The presence of such anomalies is due to cache effects and is analyzed in more detail in (Braganca, 2001). Acoustic modeling speedup and efficiency do not depend on N and are listed in table 3.

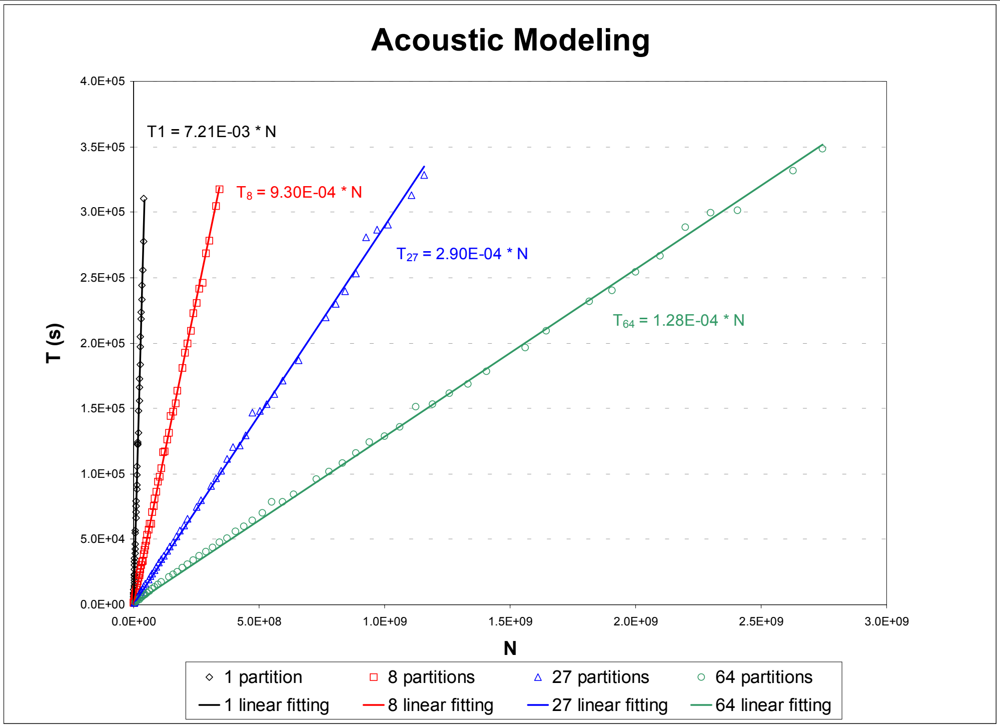

Figure 2: Acoustic T x N graph of the cubic partition schemes with 1, 8, 27 and 64 machines.

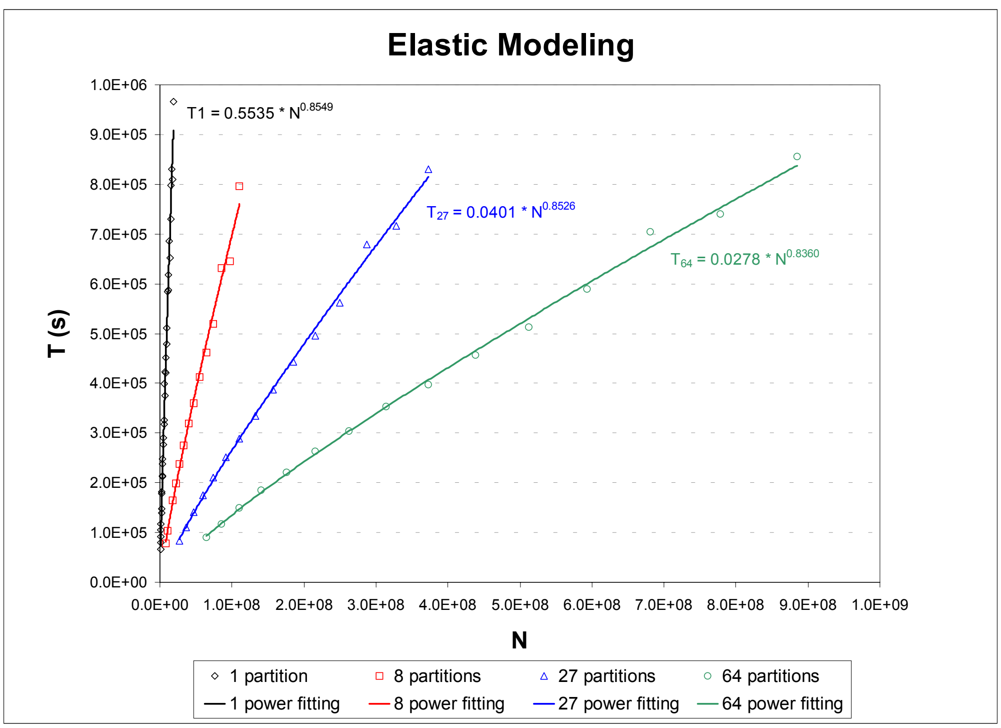

Figure 3: Elastic T x N graph of the cubic partition schemes with 1, 8, 27 and 64 machines.

Table 3: Acoustic modeling efficiency and Speedup obtained from the curve fitting.

Coefficient	Efficiency	Speedup
7,2069E-3	100,00 %	1,00
9,3013E-4	96,85%	7,75
2,8954E-4	92,19%	24,89
1,2820E4	87,84%	55,22

Elastic modeling speedup depends on N and is shown in figure 4.

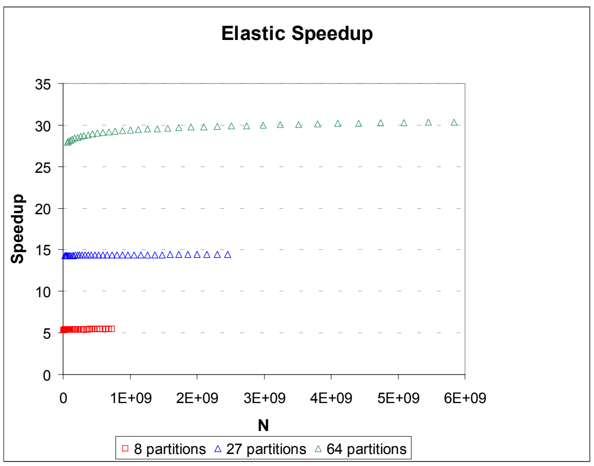

Figure 4: Speedup curves for the elastic modeling

Result analysis and conclusions

Figure 2 shows that the acoustic modeling T1 x N behavior is linear and a quasi-linear behavior occurred for T8, T27 and T64. This result lead us to believe that the communication overhead in the cluster is still very low compared to the amount of computation performed. Even huge models with N = 2.74 billions grid points (which allocate 44 Gbytes of RAM memory spread over 64 machines) did not affect the cluster performance. The high granularity of those huge simulations granted the good performance. For small N we can observe a little shift in the data points, especially on curves corresponding to 27 and 64 partitions, indicating that the low granularity wields to a little delay in execution time.

The elastic modeling T1 x N behavior of figure 3 was very surprising. It is very hard to explain why a serial code has a non-linear trend. We believe that this may be some kind of memory effect that is not yet clearly understood. The fact that this effect did not appeared in the acoustic code lead us to suspect that memory I/O delays may be the key since the elastic code requires and manipulates much more memory than the acoustic for the same model size.

For the acoustic modeling the cluster accommodate the network traffic very well. It did not even destroyed the quasi-linear behavior of T x N for 64 machines.

Elastic modeling did not get the same extraordinary efficiency and speedup as the acoustic. At the present stage we could not explain this behavior.

The use of Beowulf clusters for seismic 3-D modeling and wave equation migration is a reality due to the very low cost and performance. Despite the apparently low speedup (speedup 29.4 with 64 nodes) a huge elastic modeling (~ 1 Billion grid points) is still feasible in the cluster. Such job requires 40,2 Gb of memory and it is impossible to run in a single machine.

The graphs of figures 2 and 3 are of extreme importance because they allow us to estimate the processing time for a given modeling and number of processing nodes. This can be used to estimate processing time of RTM depth migration codes which uses the modeling algorithms analyzed here in their cores. It is also very useful to determine the best partition scheme for a given problem.

References

Myczkowski, J.; McCwan, D.; Mufti, I.; (1991); Finite-Difference Seismic Modeling in Real Time, Geophysics, The Leading Edge. 6, 49-52.

Ewing, R.; Sharpley, R.; Mitchum, D.; Oleary, P., Sochacki, J.; (1994), Distributed Computation of Wave Propagation Models Using PVM. IEEE Parallel & Distributed Technology, 2, 26-31.

Levander A.R.; (1988); Forth-order Finite-difference P-SV seismograms, Geophysics,53,1425-1436

Braganca, R.S.N.de; Bulcao, A.; Soares, Djalma M.F.; Mansur, W.J. (2001);3-D Seismic Modeling in Clusters – Performance Analysis, (Portuguese) 7th International Congress of the Brazilian Geophysical Society

Acknowledgments

The authors would like to thanks PETROBRAS for the permission of publication of this work and to the PROFEX program for financing the SISMOS 3 cluster construction.

An special thanks must be given to Joao Claudio Conceicao who was a defender of the Beowulf cluster construction at PETROBRAS Research Center from the very beginning, and to Alexandre Gamma who helped in all construction phases.

We also would like to thanks the Brazilian National Research Council - CNPq project 462304/00-0 (NV) for partially support this work via the Federal University of Rio de Janeiro - UFRJ.