Parallel Computing on PC clusters: a prestack time migration experiment

Tianfei Zhu*, Daoliu Wang and Scott Cheadle
Veritas DGC Inc, 715 Fifth Avenue SW, Suite 2200, Calgary, AB, T2P 5A2
Tianfei_Zhu@veritasdgc.com

ABSTRACT

The phenomenal growth in performance and capacity of micro processors and network technologies over the past decade, coupled with the recent development in distributed-memory parallel computing, has made PC clusters a powerful tool for high-performance scientific computing. Compared to other computing systems, PC clusters are more cost effective and less limited in their scalability. As a result, PC clusters have increasingly become the choice for computationally intensive data processing techniques such as prestack migration. Typical of current distributed-memory parallel computing techniques, however, the performance of PC clusters is also highly dependent on the implementation of a particular application, with data decomposition and inter-node communication being the two crucial components for parallel computing efficiency.

As part of efforts to move our land prestack migration programs to PC clusters, we have implemented a prestack time migration with a Message-Passing Interface (MPI) system, and experimented with different data decomposition and communication schemes to enhance the efficiency and robustness of our implementation. As a result, we have developed a data-centered approach for optimal data decomposition and communication. Experiments with synthetic and field data show that this approach has not only greatly increased computing efficiency, but also enhanced the program portability among heterogeneous clusters.

Introduction

Parallel computing has long been a focus of seismic research as algorithms for seismic data processing and imaging are overwhelmingly parallel and most of these algorithms can be decomposed over the spatial, time, frequency, and/or velocity domain(s) for parallelization. Furthermore, in some cases, parallel computing represents the only practical means for providing the enormous speed and computing power required by some computationally intensive algorithms. One of the recent advances in this long effort has been closely related to the advent of PC clusters, which combine fast, inexpensive personal computers with distributed-memory parallel computing techniques such as recently introduced Message-Passing Interface (MPI) standard (e.g., Gropp, 1999). Cost-effective and powerful, the PC cluster technique has made it possible to implement seismic algorithms once considered impractical because of their enormous demand for computer speed and capacity. This technique, for example, has been successfully employed in implementing prestack time and depth migrations for large-scale data processing (Dai, 2002).

Compared to the single-computer environment, on the other hand, parallel computing with PC clusters involves more sophisticated programming efforts, as it must now include additional components for dealing with data decomposition and communication among the computer nodes with MPI. As these components are crucial for the performance of PC clusters, considerable effort may be needed to design an optimal data decomposition and communication scheme by experimenting different configurations.

A Prestatck Time Migration Experiment

As part of the effort to deploy our land prestack imaging programs on PC clusters, we have implemented a prestack time migration with MPI, and experimented with different data decomposition and communication schemes to investigate the effects of these schemes on the robustness and efficiency of the migration program and to optimize the performance of PC clusters. As a result, we have developed a data-centered approach for data decomposition and communication. Compared to the frequently used node-centered schemes, which communicate with signals sent and received between master and slave nodes, our approach compacts all necessary information into data blocks. It thus reduces the communication between the master and slave nodes to a minimum. This not only reduces the communication to computation ratio, but also makes the entire calculation process less vulnerable to the disruption caused by faulty nodes.

Experiments with synthetic and field data further show that our approach achieves greater load balancing among the computational nodes, resulting in greater efficiency than node-centered schemes. Moreover, as the size of data blocks used in our approach is adaptive to node memory, our implementation is also expected to be more portable among heterogeneous clusters.

Acknowledgements

The authors would like to express our gratitude to Ernie Pasveer, Wayne Davis, Rick Macdonald, and David Wilkinson for their invaluable help during the completion of this study.

References

Dai, Hengchang, 2002, Parallel processing of prestack time migration on a Beowulf cluster; vol. 9, Edinburgh Anisotropy Project Research Report

Gropp, W., Lusk, E., and Skjellum, A., 1999, Using MPI, The MIT press.