A Dynamic Data Partition Algorithm Oriented to MPI and OpenMP

Zhou Chen, Chen Zhenjie, Li Feixue

School of Geographic and Oceanographic Sciences, Nanjing University, 210046
Telephone: 0-13182997228
Email: chenzj@nju.edu.cn

1. Introduction

With the rapid increase of massive multi-dimensional spatial-temporal data, geocomputation has been increasingly data-intensive and computation-intensive. But nowadays, the existing geographical information system and traditional hardware have been unable to meet the demand of massive geocomputation. Fortunately, emergence of new hardware architecture, such as multi-core processors, has broken through the bottleneck of computation and data processing to some extent. And it opens up new avenues of research for geocomputation issues. So using parallel computing in complex geocomputation with massive data becomes an irreversible trend.

As is well-known, MPI (Message Passing Interface) and OpenMP are the most popular programming models of parallel computing. MPI is the implementation standard based on message passing programming model, while OpenMP is an industry standard sharing the common storage. In a multiprocessor environment, MPI+OpenMP hybrid programming model can realize the design thought of modularization and improve effectively the performance of system.

Runtime is an important indicator to evaluate the performance of parallel programs. Runtime of parallel programs are usually determined by the time that the slowest process takes. Generally, runtime of different processes will differ from each other because of different data complexity. In a multiprocessor system, the load unbalance phenomenon probably occurs, which will directly affect the performance, so it is necessary to implement load balancing in multiprocessor systems. In theory, load balancing means that the calculative processes will be distributed loads evenly so that nearly all processes tend to accomplish tasks at the same time, decreasing the longest runtime of the process, to improve overall performance of parallel programs. Load balancing algorithms can be classified into two types according to the basic principle: static load balancing algorithm and dynamic load balancing algorithm. Static algorithm has some drawbacks — poor real-time capability, blindness and low efficiency. By contrast, dynamic algorithm can balance load on each node according to the current status of the system.

Foundation item: National High Technology Research and Development Program of China (Granted NO.2011AA120301)
Biographies: Zhou Chen (1990-), male, master student; Chen Zhenjie (1974- , corresponding author), male, doctor, associate professor, chenzj@nju.edu.cn; Li Feixue (1983-), female, doctor, associate professor
Address: School of Geographic and Oceanographic Sciences, Nanjing University, No.168 Xianlin Avenue, Nanjing, Jiangsu, P.R.China

2. Methods and Conclusions

As one of the crucial technologies of parallel computing, dynamic load balancing has significant implications for the full utilization of geographical computing resources and reasonable scheduling of tasks. An optimized data partition strategy can cost less to achieve load balancing. Up to now, the existing researches on data partition strategy have already been very in-depth but more focused on partitioning data statically rather than dynamic data partition. The strategy of dynamic data partition is able to obtain better effect because of its outstanding real-time property and practicality. This thesis extends discussion surrounding the implementation of the dynamic data partition algorithm from the following several aspects:

Firstly, in this thesis, the technology of runtime prediction has been studied on basis of the methods and characteristics of the MPI+OpenMP hybrid programming model. And then a model of task runtime prediction is built to set load threshold dynamically to schedule tasks reasonably. The main factors of task runtime of parallel geocomputation oriented to MPI+OpenMP hybrid programming model include data complexity, number of processes, initial distributed data granularity, error range, number of threads and so on. During building the model, a variety of experimental performance data are collected to adopt some methods of spatial data analysis to fit out the prediction model of data complexity, number of processes, initial distributed data granularity, error range, number of threads and the task runtime. Not only that, in the course of actual task scheduling operation, the parameters of the model will be revised by self-tuning in line with the real-time measurement. Also, it can improve the accuracy of predictions and lay the foundation for a reasonable task scheduling.

Secondly, a dynamic data partition algorithm oriented to MPI+OpenMP hybrid programming model is proposed. The algorithm employs two-level parallel mode which means that MPI is adopted in task allocation and scheduling while OpenMP is used in specific task calculation. While the program is running, the administrative process generated by the system takes charge of collecting load information for partitioning tasks to calculative processes, choosing operation velocity as the load indicator. Then, proportion relationships are set up between the load indicator and tasks, with the purpose of the consistency between the quantity of assigned tasks and the calculation velocity of the tasks, to achieve load balancing.

Finally, polygon scanline rasterization algorithm is used as a case to verify the efficiency of the result of the experiments. The dynamic data partition strategy proposed above is applied into the scanline rasterization algorithm. Some contrastive and comparative analyses have been launched between the algorithm proposed and the algorithm without the strategy of dynamic load balancing in aspects of parallel efficiency and load degree. It is proved that the dynamic data partition algorithm oriented to MPI+OpenMP hybrid programming model can achieve favorable acceleration efficiency and load balancing.

Key Words: Geocomputation; MPI+OpenMP hybrid programming model; Task runtime prediction model; Dynamic data partition; Load balancing

3. Reference

M.M. Hayat, S. Dhakal, C.T. Abdallah et al, 2004. Dynamic Time Delay Models for Load Balancing, Part II: Stochastic Analysis of the Effect of Delay Uncertainty. *Advances in Time Delay Systems*, 38:355-368.

S. Dhakal, M.M. Hayat, JE.Pezoa, C.Yang et al, 2007. Dynamic Load Balancing in Distributed Systems in the Presence of Delays: A Regeneration-Theory Approach. *IEEE Transactions on Parallel and Distributed Systems*, 18(4):485-497.

Chi-chung Hui and Samuel T. Chanson, 1999. Hydrodynamic Load Balancing. *IEEE Transactions on Parallel and Distributed System*, 10(11):1118-1137.

Yang Guowei, Zhou Xianpu and Tu Xuyan, 2003. Tensibility of the Matter Element System and its Applications. *Advances in Modeling &Analysis A*, 40 (1):23-39.

Tonguz O.K and Yanmaz E, 2008. The Mathematical Theory of Dynamic Load Balancing in Cellular Networks. *IEEE Transactions on Mobile Computing*, 7(12):1504-1518.

G.L.Reijns and A.J.C.v. Gemund, 2005. Predicting the Execution Times of Parallel-independent Programs Using Pearson Distributions. *Parallel Computing*, 31(8-9):877-899.

D.R.Liang and S.K.Tripathi, 2000. On Performance Prediction of Parallel Computations with Precedent Constraints. *IEEE Transactions on Parallel and Distributed Systems*, 11(5):491-508.

Lu Zhao, Zhang Jinjuan, Shi Jun et al, 2010. Dynamic Load Balancing Strategies in MPI Parallel Environment. *Computer Technology and Development*, 20(5):132-135,149.

Guo Jing, Zhu Yongzhi and Wang Yanling, 2009. Research on Dynamic Load Balancing Algorithm Based on MPI. *Computer Technology and Development*, 19(5): 150-153,191.

Yu Dunfu, Li Hongjian, Tang Hong et al, 2012. Dynamic Load Balancing Algorithm Design and Application Based on Feedback. *Application Research of Computers*, 29(2): 527-529.

Liu Bin and Shi Feng, 2007. Research on Dynamic Load Balancing Algorithm Based on Message Passing Mechanism. *Computer Engineering*, 33(10): 58-60.