Effective Performance of Large-Scale MHD simulation for Planetary Magnetosphere with Massively Parallel Computer

Keiichiro FUKAZAWA[1, 2], and Takeshi NANRI[1]

[1] Research Institute for Information Technology, Kyushu University, Japan
[2] International Center for Space Weather Science and Education, Kyushu University, Japan

1 Introduction

The global configuration of magnetosphere is very large while the scale of magnetospheric phenomena is varied as 10^8 order. For example, the aurora is small scale phenomenon however it is related to the global magnetospheric convection through the magnetic field lines. The magnetosphere is formed by the interaction between the space plasma from the Sun and the ionosphere, etc. and the planetary magnetic field. To describe the behaviour of space plasma, the Vlasov-Maxwell system equations are used for studies of electron-scale processes. For larger-scale processes such as the magnetosphere, however, electron-scale processes can be sometimes neglected then the MagnetoHydroDynamic (MHD) equations are used. The MHD equations are highly nonlinear and are very complex to solve by hand calculations. Thus in order to simulate the magnetosphere, it is needed to use the supercomputer.

Recently the scalar-type supercomputers are dominant in the world [1]. Thus it is necessary to optimize the simulation codes to the scalar-type supercomputers. These supercomputers have a large number of processors to achieve the high performance. To use such many processors, simulation codes are required the high parallelism. In addition, the execution efficiency becomes more important because the peak (ideal) performance greatly increases. Furthermore, post-processing including transferring and storing the simulation data is a serious problem.

The purpose of the present study is to effectively perform the MHD code for space plasma simulations on scalar-type massively parallel supercomputer systems toward peta and exa scale computing in a few years. In this paper some improvements to our simulation code and sequences to perform the large scale and high resolution simulation on the massively parallel computer are presented.

2 Simulation Model

We use a three-dimensional MHD code, which has been used for studies on global structures and dynamics of planetary magnetospheres [2, 3, 4]. The MHD code uses the "Modified Leapfrog" method, in which partial difference equations are solved by the two-step Lax-Wendroff method for one time step and then by the Leapfrog scheme for 7 time steps and the procedure is repeated. Thus the method has second-order accuracy in both space and time.

As a parallelization technique, the domain decomposition method for dividing three-dimensional space is adopted. Usually in parallel computing on a distributed-memory computer, the domain decomposition is an easy way to decompose three-dimensional Eularian variables. In the case of a three-dimensional model, the dimension of domain decomposition can be chosen as one dimension, two dimensions, or three dimensions [5]. To use large number of cores for peta and exa scale computing, we need to select the three-dimensional decomposition method. For example if we calculate the 3000×3000×3000 grids, we can use the 10^9 cores with three-dimensional decomposition method while 10^6 cores with two-dimensional decomposition method and 10^3 cores with one-dimensional decomposition.

Effective use of cache memory is important for obtaining better performance on a scalar processor. In this study we measure the performance of normal three-dimensional decomposition (Type A) and array replaced type three-dimensional decomposition (Type B) to consider the cache hit efficiency [5]. Here Type A array configuration is *f*(*nx*, *ny*, *nz*, *m*) and Type B is *f*(*m*, *nx*, *ny*, *nz*) where *nx*, *ny* and *nz*, are the number of *x*, *y*, and *z* directions, respectively. *m* is the variable of MHD (the plasma density, velocity vector, pressure, and magnetic field vector).

3 Performance Measurements

To understand the effect of cache, two supercomputer systems are used in this study. The Fujitsu PRIMERGY RX200 S6 at Kyushu University is the PC-cluster type system consists of Intel Xeon processor (Westmere) and the Fujitsu PRIMEHPC FX10 at the University of Tokyo is the successor computer system of 'K-computer' which consists of SPARC64 IXfx processor. We use the array of 64 MB/core for the computational domain and additionally 192 MB/core for workspaces for computing the MHD equations with the Modified Leapfrog method. To minimize the communication time, we use a buffer array which stores all the boundary data for inter-core and inter-node communications [5].

3.1 Fujitsu PRIMEGY RX200 S6

Figure 1 shows the performance of MHD code with RX200 S6. Using 4704 cores, we achieved the computational peak performance 17 TFlops and computational efficiency 32% with three dimensional decomposition Type A. The efficiency of Type B (cache hit considered type) is about 10% less than Type A (12 TFlops). Thus Type B is not suitable for this system. This trend is the same as our previous results [5]. Here the difference of 1% of computational efficiency becomes the difference of 500 GFlops in the performance so that we should add performance tuning to simulation code using the massively parallel computer.

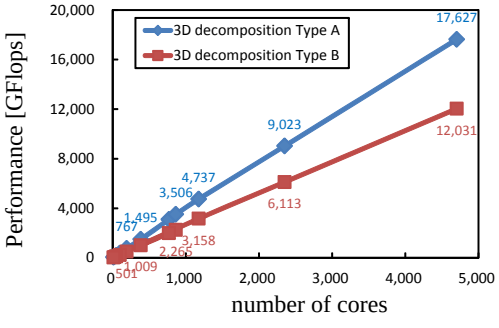

Fig. 1. Performance of MHD code with RX200 S6.

3.2 Fujitsu PRIMEHPC FX10

Figure 2 shows the results of performance measurements with FX10. In this case we use up to 1024 cores during a trial run and we obtained 2.5 TFlops (16% of efficiency) in the Type B case. This is adverse result of RX200 S6. The cache hit type is suitable for FX10. There is about 5% difference of computational efficiency between the Type A and Type B.

From both results, it becomes clear that there are different optimizations for scalar type computer and the optimization is very important to perform the large scale computing.

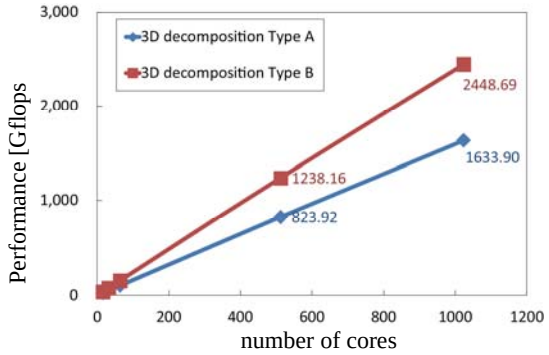

Fig.2. Performance of MHD code with FX10.

4 Data I/O and Post Processing

We start running the large scale MHD simulation of Saturn's magnetosphere using the results of performance measurements then overcome three problems. First problem is about data I/O. In the simulation we write the large simulation data (about 400 GB at maximum) with 864 cores. In the simulation code, the distributed data divided by the number of rank due to the parallel computing were gathered into one data in one rank and it was written as an output data. However, the speed of writing the large size of data is very slow and the size of gathered data became larger than the size of memory in one node (i.e. RX200 S6 has 64GB memory per node) thus it is impossible to use the gather technique in I/O part. Thus it is necessary to write the simulation data in each rank for the massively parallel computing.

Then we have the problem of data storing. The output data size is 400 GB/sampling time and 300 data are required to see the time evolution of the magnetosphere thus the total size of data becomes over 120 TB. To store these data, the Gfarm [6] storage system operated by the National Institute of Information and Communications Technology (NICT) [7] is used. The Gfarm is the distributed storage system using the grid computing and has the advantage of data mirroring and data access speed. The Gfarm storage system of NICT consists of over 50 storage servers at several locations in Japan and its size reaches almost 3PB currently. The transfer speed of simulation data from the supercomputer system at the University of Tokyo to the NICT is 27 MB/s on average. In the exa scale computing era, due to the limitation of I/O bandwidth, the simulation data on memory will be transferred to the memory on the storage server directly.

Finally the post processing of distributed data is the problem. It is impossible to analyse the large size data on one computational node because there is not enough memory to treat the data. Thus we develop the programs to analyse and visualize the data [8]. These programs are based on CUI (Character User Interface) so that the user cannot control the functions viscerally, however, the user can use up to the limit of hardware (CPU and memory), and parallelize programs easily.

5 Summary

In this study we have made performance measurements and tuning of MHD code for the massively parallel scalar supercomputer which is the PRIMEGY RX200 S6 at the Kyushu University and the PRIMEHPC FX10 supercomputer system at the University of Tokyo. For the MHD code, we evaluated two types of decomposition methods: regular three-dimensional domain decompositions and a cache hit considered type of three-dimensional decomposition. We found that the regular three-dimensional decomposition methods are suitable for RX200 S6 and the cache hit considered type of three-dimensional decomposition is suitable for FX10. As the results, we understand the importance of optimization to perform the large scale computing.

To run the large scale MHD simulation, there are three problems which are data I/O, storage to store the data and post processing. Then we introduce the distributed I/O due to the shortage of memory to gather the divided data on each core to one node. In addition we use the distributed storage system Gfarm at NICT to store the sum of large scale time evolution data. This system has an advantage of data transfer. Finally we have developed analysis and visualization programs to treat the large size of simulation results.

Thanks to these results we can simulate the planetary magnetosphere with high resolution. For example we can calculate the Saturn's magnetosphere with good spatial resolution and obtained the relation of small scale vortex and patchy auroral emission [9].

Acknowledgements

This work was supported by CREST, the Japan Science and Technology Agency (JST). The computational resource was provided by Information technology center in the University of Tokyo, Research Institute for Information Technology in the Kyushu University and the OneSpaceNet in the NICT Science Cloud. This work was also conducted as a Joint Usage / Research Center for Interdisciplinary Large-Scale Information Infrastructures in Japan and Advanced Computational Scientific Program 2011, Research Institute for Information Technology, Kyushu University.

References

[1] Top500 Supercomputing Sites. (http://www.top500.org/)

[2] Ogino. T, R. J. Walker, M. Ashour-Abdalla, "A global magnetohydrodynamic simulation of the magnetopause when the interplanetary magnetic field is northward", *IEEE Trans. Plasma Sci.* vol. 20, 1992, 817-828.

[3] Fukazawa, K., T. Ogino, and R.J. Walker, The Configuration and Dynamics of the Jovian Magnetosphere, J. Geophys. Res., 111, A10207, doi:10.1029/2006JA011874, 2006.

[4] Fukazawa, K., S. Ogi, T. Ogino, and R.J. Walker, Magnetospheric Convection at Saturn as a Function of IMF Bz, Geophys. Res. Lett., 34, L01105, doi:10.1029/2006GL028373, 2007.

[5] Fukazawa, K., T. Umeda, T. Miyoshi, N. Terada, Y. Matsumoto, and T. Ogino, Performance measurement of magneto-hydro-dynamic code for space plasma on the various scalar type supercomputer systems, IEEE Transactions on Plasma Science, Vol. 38, No. 9, pp2254, 2010.

[6] Tatebe, O., Y. Morita, S. Matsuoka, N. Soda, H. Sato, Y. Tanaka, S. Sekiguchi, Y. Watase, M. Imori, T. Kobayashi, "Grid Data Farm for Petascale Data Intensive Computing", Technical Report, *Electrotechnical Laboratory, ETL-TR2001-4*, 2001.

[7] Ken T. Murata, S. Watari, T. Nagatsuma, M. Kunitake, H. Watanabe, K. Yamamoto, Y. Kubota, H.Kato, T. Tsugawa, K. Ukawa, K. Muranaga, E. Kimura, O. Tatebe, K. Fukazawa and Y. Murayama, A Science Cloud for Data Intensive Sciences, *CODATA Data Science Journal*, submitted, 2012.

[8] Three dimensional visualization codes with VRML. (http://center.stelab.nagoya-u.ac.jp/web1/simulation/jst2k/hpf02.html)

[9] Fukazawa, K., T. Ogino, and R. J. Walker (2012), "A Magnetohydrodynamic Simulation Study of Kronian Field-Aligned Currents and Aurora", *J. Geophys. Res.*, 117, A02214, doi:10.1029/2011JA016945.