§30. Development of PASMO Code and In-situ Visualization

Ohtani, H., Ohno, N. (Univ. Hyogo)

Magnetic reconnection is widely considered to play an important role in energetically active phenomena in high-temperature plasmas. In spite of intensive research, many basic questions on the details of mechanisms of reconnection still remain poorly understood. To clarify the relationship between particle kinetic effects and anomalous resistivity due to plasma instabilities in the reconnection phenomena, we have been developing a three-dimensional particle simulation code for an open system, called "PASMO[1,2,3]".

In the 2012 fiscal year, we developed the code PASMO and researched "in-situ visualization" in order to perform large-scale particle simulations and to visualize such big data which is produced by the simulation codes.

Development of PASMO

PASMO is the code for a large-scale simulation because it deals with many particles. In order to perform such large-scale simulation on a distributed memory parallel computer system, we have parallelized the code on the domain decomposition method[4]. For performing larger-scale simulation, we improved it as follows:

- The Maxwellians for particle velocity are generated at the upstream boundary every time step. The quiet-start method[5] is one of the generation schemes of the Maxwellian but needs large memory for work arrays. On the other hand, the Box-Muller method[6] generates random deviates with a normal distribution and does not need such work arrays. We adopted Box-Muller method for generation of Maxwellian, and succeeded the reduction of used memory size and simplified the algorithm of the Maxwellian generator at the upstream boundary in the domain decomposition method.
- We changed the type declaration statements about the variables of the number of particles. Since a total number of particles exceeds a billion, the length of the variables should be 8 bytes. By using the compile option, it is easy to change the length of all integer variables, but the excessive memory is needed. We changed the type declaration statements only about the variables of particles whose length is 8 bytes.

In-situ visualization

Since the simulation data have been getting larger and larger, in-situ visualization would be one of the indispensable techniques in the near future. We developed an in-situ visualization tool for PIC simulation codes which are parallelized by MPI. Because graphics hardware is not required to draw pictures, it runs on high-performance computers. The tool visualizes charged particles as spheres. It also supports color slice, isosurface, volume rendering, arrows and streamlines for visualizing scalar and vector fields such as electric and magnetic fields. We designed the tool in the form of a Fortran90's module so that simulation researchers can use it by adding some Fortran statements for calling the subroutines to proper points in their PIC simulation codes. Thus they might have to change their codes but only a little. This tool visualizes data by accessing it directly, and outputs images after the code finishes the calculation on a certain time step. Therefore the integrated PIC codes need more CPU times than the original one but do not require much more memory. We are now planning to parallelize this tool by OpenMP, integrate it into PASMO and execute the integrated PASMO on the Plasma Simulator.

1) Horiuchi, R. et al: Phys. Plasmas **6** (1999) 4565.
2) Ohtani, H. et al: LNCL **4759** (2008) 329.
3) Ohtani, H. and R. Horiuchi: PFR **4** (2009) 024.
4) Liewer, P.C. and Decyk, V.K.: J. Comp. Phys. **85** (1989) 302.
5) Birdsall, C.K. and Langdon, A.B.: Plasma Physics via Computer Simulation, New York, 1985.
6) Press, W.H. et al: Numerical Recipes in FORTRAN, New York, 1986.