HPC Parallel Programming Model for Gyrokinetic MHD Simulation[*)]

Hiroshi NAITOU, Yusuke YAMADA, Shinji TOKUDA[1,2)], Yasutomo ISHII[2)], Masatoshi YAGI[2,3)]

Yamaguchi University, 2-16-1 Tokiwadai, Ube 755-8611, Japan
[1)]*Research Organization for Information Science and Technology, 2-32-3 Kita-shinagawa, Shinagawa-ku, Tokyo 140-0001, Japan*
[2)]*Japan Atomic Energy Agency, 801-1 Mukoyama, Naka 311-0193, Japan*
[3)]*Kyushu University, 6-1 Kasuga, Fukuoka 816-8580, Japan*

(Received 22 December 2010 / Accepted 7 February 2011)

The 3-dimensional gyrokinetic PIC (particle-in-cell) code for MHD simulation, Gpic-MHD, was installed on SR16000 (“Plasma Simulator”), which is a scalar cluster system consisting of 8,192 logical cores. The Gpic-MHD code advances particle and field quantities in time. In order to distribute calculations over large number of logical cores, the total simulation domain in cylindrical geometry was broken up into $N_{DD\text{-}r} \times N_{DD\text{-}z}$ (number of radial decomposition times number of axial decomposition) small domains including approximately the same number of particles. The axial direction was uniformly decomposed, while the radial direction was non-uniformly decomposed. N_{RP} replicas (copies) of each decomposed domain were used (“particle decomposition”). The hybrid parallelization model of multi-threads and multi-processes was employed: threads were parallelized by the auto-parallelization and $N_{DD\text{-}r} \times N_{DD\text{-}z} \times N_{RP}$ processes were parallelized by MPI (message-passing interface). The parallelization performance of Gpic-MHD was investigated for the medium size system of $N_r \times N_\theta \times N_z = 1025 \times 128 \times 128$ mesh with 4.196 or 8.192 billion particles. The highest speed for the fixed number of logical cores was obtained for two threads, the maximum number of $N_{DD\text{-}z}$, and optimum combination of $N_{DD\text{-}r}$ and N_{RP}. The observed optimum speeds demonstrated good scaling up to 8,192 logical cores.

Keywords: gyrokinetic theory, particle-in-cell code, magnetohydrodynamics, tokamak, symmetric multiprocessing, message-passing interface, multi-threads, multi-processes

DOI: 10.1585/pfr.6.2401084

1. Inroduction

In order to explain and predict global magnetohydrodynamic (MHD) phenomena in present-day and future tokamaks, it is crucial to develop a model including kinetic effects because conventional MHD model usually fails to elucidate the strange phenomena observed in experiments. Here, the terminology of “kinetic” is used to designate that the velocity space dynamics of charged particles plays an important role. The electromagnetic gyrokinetic PIC (particle-in-cell) code is one of the candidates to simulate these kinetic MHD phenomena. It is based on the gyrokinetic theory [1, 2] in which gyro-motion of charged particles are averaged over gyro-orbits; hence, we can use larger time step and larger spatial mesh size compared to the conventional PIC code. Even with these advantages, the tokamak simulation with the gyrokinetic PIC code requires huge computer resources because it must follow extremely large number of charged particles (electrons and ions) in the whole tokamak plasma. It is inevitable to reduce drastically the computation time by fully utilizing the performance of the massive-parallel computers, which is a grand challenge to the high-performance computing (HPC).

We developed the gyrokinetic PIC code for MHD simulation, Gpic-MHD [3, 4], written in the cylindrical coordinates. The basic formulation is the same as the one used for the gyr3d code [5, 6], which was developed more than decade ago and written in the Cartesian coordinates. The standard version of Gpic-MHD as well as gyr3d uses delta-f scheme in which marker particles represents only the deviation from the equilibrium velocity distribution; the noise generated by the discreteness of marker particles reduces drastically.

There are 2-dimensional (2d) and 3-dimensional (3d) versions of Gpic-MHD. The 2d version assumes the single helicity and successfully simulated kinetic internal kink mode with $m/n = 1/1$ mode (m and n are poloidal and toroidal mode numbers, respectively). In the nonlinear phase, the collisionless magnetic reconnection was observed, which is closely related to the crash phase of the sawtooth oscillation. The 2d version is light compared to the 3d version; it is easy to test new ideas or new algorithms. We used 3d Gpic-MHD to study the parallelization

author’s e-mail: naitou@yamaguchi-u.ac.jp

[*)] This article is based on the presentation at the 20th International Toki Conference (ITC20).

performance on various computers [3,4].

To simulate larger scale and higher beta plasma, the split-weight scheme [7, 8] was proposed as the improvement of the conventional delta-f scheme. We proposed alternative algorithm, which uses the vortex equation and generalized Ohm's law along the magnetic field to calculate field quantities [9, 10]. We verified that the 2d Gpic-MHD with the advanced scheme could successfully simulate the collisionless and kinetic internal kink mode in larger scale and higher beta tokamaks.

The standard version of 3d Gpic-MHD with conventional delta-f scheme has been used as the benchmark code to study parallelization performance of various massive-parallel computers. The 1d domain decomposed version of Gpic-MHD uniformly breaks up the total simulation domain in the axial direction. The parallelization performance on Altix3700Bx2 (now replaced to new one) of JAEA (Japan Atomic Energy Agency) was studied in an article [3]. Altix3700Bx2 is based on the rather conservative parallelization architecture; 16 nodes with 128 single-core CPUs per node. The 1d domain decomposed version of Gpic-MHD with replicas ("particle decomposition") showed no saturation up to 1024 cores.

The performance of 3d Gpic-MHD on SR16000 ["Plasma Simulator" in NIFS (National Institute for Fusion Science)] was studied in the article [4]. SR16000 is a state-of-the-art scalar SMP (symmetric multiprocessing) cluster system consisting of 8,192 logical cores (128 nodes, each node includes 32 physical cores with SMP architecture, and one physical core is equivalent to two logical cores with multithreading technology). The parallelization performance was studied for the small size system of $N_r \times N_\theta \times N_z = 129 \times 128 \times 128$ mesh. The 1d domain decomposed version with replicas showed good performance. The 2d domain decomposed version was developed to simulate the larger system. The 2d domain decomposed version breaks up the total domain in the radial direction in addition to the axial direction. It was found that 2d domain decomposed version with replicas showed slightly less performance compared with 1d domain decomposed version for this small size system.

Because the wavelengths of modes in tokamaks are long along the magnetic field and short across the magnetic field, the resolution of the mesh should be low along θ and z, and high in r-direction. In this article we investigated the parallelization performance of Gpic-MHD on SR16000 for the medium size system of $1025 \times 128 \times 128$. We studied mainly 2d domain decomposed version including 1d domain decomposition as a special limit.

We will execute the jobs with the large system size of about $10000 \times 128 \times 128$ for the future large-scale tokamak simulation by using 10^5-10^6 logical cores.

2. Parallelization Model

The axial direction is broken up into $N_{DD\text{-}z}$ equal domains, while the radial direction is divided into $N_{DD\text{-}r}$ non-equal domains. The example for $N_{DD\text{-}r} = 8$ and $N_{DD\text{-}z} = 8$ is illustrated in Figure 1. This example assumes that the distribution of marker particles is uniform in space. Note that each decomposed domain includes almost equal number of particles in order to make computational load uniform on each logical core. The decomposed domain includes different number of radial meshes, so the load for the field calculation is not uniform for different radial domains. Usually the particle calculation is dominant for PIC code over field calculation. As the ratio of the field calculation to the particle calculation increases for the highly parallelized case, more sophisticated load balance may be needed, but it is left for the future study. We can use N_{RP} replicas for each decomposed domains to use large number of logical cores.

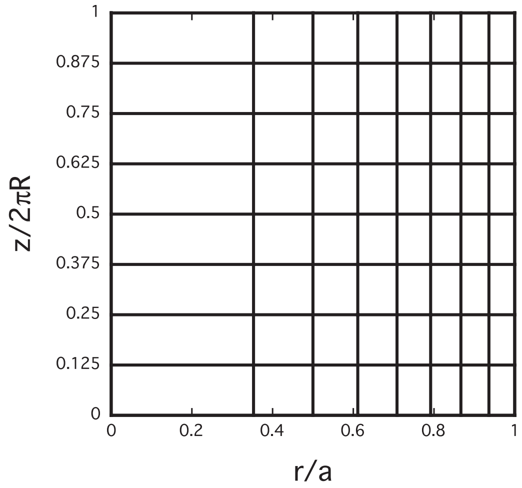

Fig. 1 Schematic view of 2-d domain decomposition.

The hybrid parallelization model of multi-threads and multi-processes is used in which threads are parallelized by the auto-parallelization and processes are parallelized by the message-passing interface (MPI). Usually the best performance was obtained for two threads (SMP = 2). The number of processes is $N_{DD\text{-}r} \times N_{DD\text{-}z} \times N_{RP}$.

The field quantities are represented by Fourier mode expansions both in axial and azimuthal directions by using FFT (Fast Fourier Transformation). Finite difference method is used in radial direction. When we Fourier transform in the axial direction, the domain decomposition in z is transposed to the domain decomposition in θ. After the "particle pushing", the particles getting out of the designated local domain are moved to the adjacent local domains by the communication between processes.

3. Parallelization Performance

We used the medium size system of $1025 \times 128 \times 128$. Usually best performance was obtained for two threads (SMP = 2) and maximum number of axial domain decomposition, $N_{DD\text{-}z} = N_z/2 = 64$. It is not practical to use $N_{DD\text{-}z} = 128$ because the real and imaginary parts of

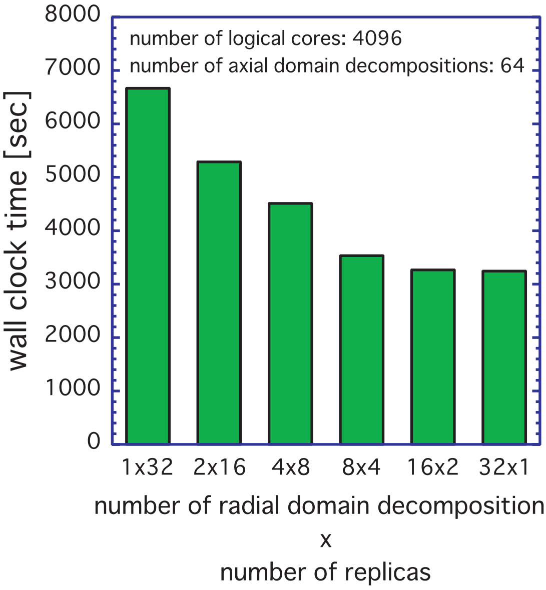

Fig. 2 Wall clock times versus various decompositions.

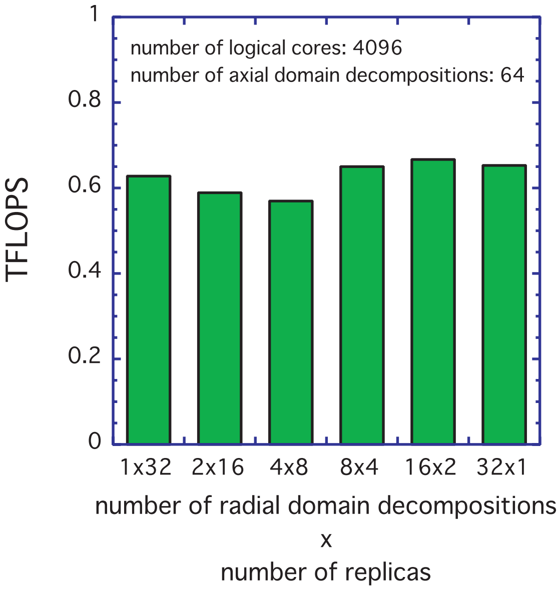

Fig. 3 Flops versus various decompositions.

Fourier components reside in different domains. All the data in this article were obtained with SMP = 2. Figure 2 shows the wall clock times for various combinations of $N_{DD\text{-}r}$ and N_{RP} for 4,096 logical cores. The number of particles for each logical core is approximately one million. The total number of particles is 4.092 billion. The time steps of 1000 are followed. It is clear that the wall clock time decreases by half when $N_{DD\text{-}r}$ is increased from 1 to 32. The case with $N_{DD\text{-}r}$ = 32 and N_{RP} = 1 is optimum, but the result of the case with $N_{DD\text{-}r}$ = 16 and N_{RP} = 2 shows the almost same performance. Fig. 2 demonstrates that 2d domain decomposition is preferable compared with 1d domain decomposition with replicas if the mesh size is large enough. Note that we used the small size system of 129 × 128 × 128 in the previous article [4] and found no apparent difference between 1d and 2d decompositions.

The speed of Gpic-MHD can be estimated by the inverse of the wall clock time. Figure 3 shows similar results as Fig. 2 but for FLOPS (floating point operations per second). The reason that FLOPS for small $N_{DD\text{-}r}$ is not so small compared with the inverse of the wall clock time is as follows. The 1d domain decomposition and 2d domain decomposition with small number of $N_{DD\text{-}r}$ includes large redundant field calculations. Hence the number of floating point operations is large without communications between logical cores; the FLOPS becomes large. For this case FLOPS is not a good index to represent the performance of the code.

Figure 4 shows "strong scaling" of FLOPS depending on the number of logical cores. The total number of particles is 8.192 billion. For this case, the dependence of FLOPS is almost identical to the dependence of the inverse of the wall clock time (hence, the figure is not shown). The solid circles connected with solid lines show FLOPS for different $N_{DD\text{-}r}$ with fixed $N_{DD\text{-}z}$ = 64. It is clear that FLOPS increases as $N_{DD\text{-}r}$ increases. The solid circles connected with broken lines, show FLOPS for different $N_{DD\text{-}z}$ with fixed $N_{DD\text{-}r}$ = 64. Slight degradation of the scaling is observed for $N_{DD\text{-}r}$ = 64. This degradation occurs because the ratio of the field calculation increases up to 40 percents for $N_{DD\text{-}r}$ = 64 mainly by the increase of the communication between processes. The discussion about this is stated in Sec. 4. We can reduce $N_{DD\text{-}r}$ and use replicas. The solid square in Fig. 4 with $N_{DD\text{-}r}$ = 32 and N_{RP} = 2 shows the best performance of 1.25 TFLOPS.

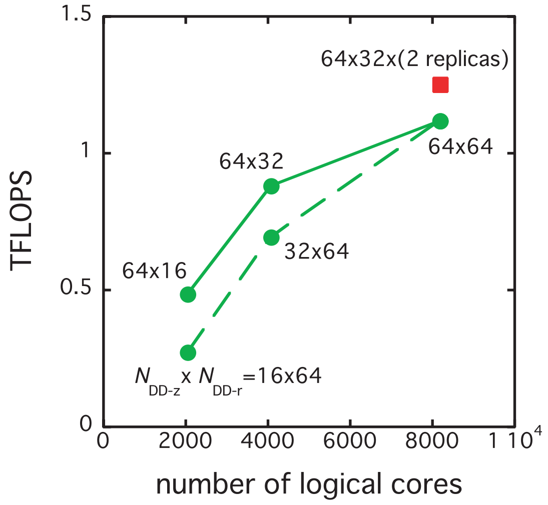

Fig. 4 Flops versus number of logical cores.

Figure 5 shows the wall clock time for the various combination of $N_{DD\text{-}r}$ and N_{RP} with fixed $N_{DD\text{-}z}$ = 64. The number of logical cores is 8,192 and the number of particles 8.192 billion. The best result is obtained for $N_{DD\text{-}r} \times N_{RP}$ = 16 × 4 and 32 × 2. There is a tendency that speed of the code increases with $N_{DD\text{-}r}$ but saturates at some number and the speed slows down as $N_{DD\text{-}r}$ increases

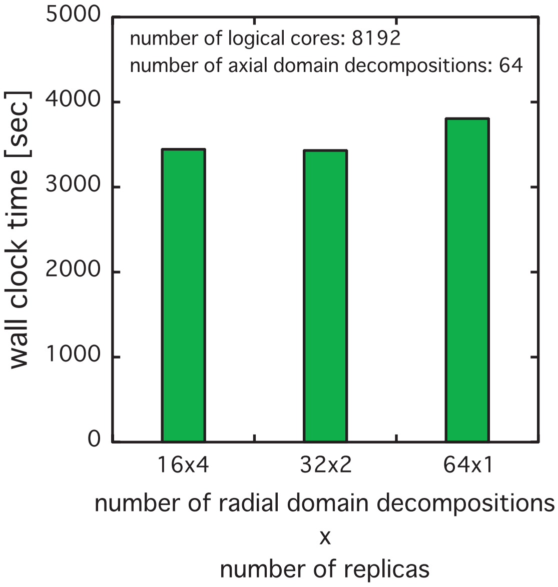

Fig. 5 Wall clock times versus various decompositions.

further. For the good performance, there is an optimum choice of $N_{DD\text{-}r}$ and N_{RP}.

4. Conclusions and Discussion

The 3d Gpic-MHD code was parallelized by the 2d domain decomposition and "particle decomposition". The cylindrical domain was broken up into $N_{DD\text{-}z}$ axial domains and $N_{DD\text{-}r}$ radial domains. The N_{RP} replicas of each decomposed domain were used. The axial direction is uniformly decomposed, while the radial direction is non-uniformly decomposed. Each local domain includes approximately the same number of particles. The parallelized Gpic-MHD was installed on SR16000, which is the state-of-the art scalar SMP cluster system consisting of 8,192 logical cores. The hybrid parallelization model of multi-threads and multi-processes was employed: threads are parallelized by the auto-parallelization and $N_{DD\text{-}r} \times N_{DD\text{-}z} \times N_{RP}$ processes were parallelized by MPI. The performance of Gpic-MHD was investigated for the medium size system of $N_r \times N_\theta \times N_z = 1025 \times 128 \times 128$ and 4.096 or 8.192 billion particles. The highest speed (or FLOPS) for the fixed number of logical cores was obtained for two threads, the maximum number of axial domain decomposition $N_{DD\text{-}z}$ = 64, and optimum combination of $N_{DD\text{-}r}$ and N_{RP}. The optimum speeds and FLOPS scaled very well with the increasing number of logical cores ("strong scaling").

The maximum FLOPS obtained, is 1.3 TFLOPS for 8,192 logical cores, which is 1.6 percents of the theoretical maximum speed of 77 TFLOPS of SR16000. We still have margin for the further optimization. The present version is not optimized well because each radially localized domain has the data of global radial domains, because the Poisson solver uses global data distributed over radial directions. The "gather" calculation, in which each logical core (radially decomposed domain) gathers the field data of the different radial domains, uses about 40 percents of the wall clock time. We are testing the version with different algorithm, such as BiCGstab-P method [11], which uses mainly localized data included in the separate domain. The results will be reported in the future article.

We will execute the productive runs with the large system size of about $10000 \times 128 \times 128$ for the future large-scale tokamak simulation by using 10^5-10^6 logical cores. Even with the present version of Gpic-MHD we can expect the good parallelization scaling depending on the increasing number of logical cores.

Acknowledgments

This research is partially supported by Grant-in-Aid for Scientific Research (C) (20560766), collaboration programs between universities and JAEA (22K450), collaboration programs between universities and NIFS (NIFS09KTBL009).

[1] W.W. Lee, J. Comput. Phys. **72**, 243 (1987).
[2] T.S. Hahm, W.W. Lee and A. Brizard, Phys. Fluids **31**, 1940 (1988).
[3] H. Naitou, H. Hashimoto, Y. Yamada, S. Tokuda and M. Yagi, J. Plasma Fusion Res. SERIES **8**, 1158 (2009).
[4] H. Naitou, H. Hashimoto, Y. Yamada, S. Tokuda and M. Yagi, accepted in Progress in Nuclear Science and Technology.
[5] H. Naitou, K. Tsuda, W.W. Lee and R.D. Sydora, Phys. Plasmas **2**, 4257 (1995).
[6] H. Naitou, T. Sonoda, S. Tokuda and V.K. Decyk, J. Plasma Fusion Res. **72**, 259 (1996).
[7] I. Manuilskiy and W.W. Lee, Phys. Plasmas **7**, 1381 (2000).
[8] W.W. Lee, J.L.V. Lewandowski, T.S. Hahm and Z. Lin, Phys. Plasmas **8**, 4435 (2001).
[9] H. Naitou, K. Kobayashi, H. Hashimoto, T. Andachi, W.W. Lee, S. Tokuda and M. Yagi, "Proposal of a brand-new gyrokinetic algorithm for global MHD simulation", Bull. APS, 51st Annual Meeting of the Division of Plasma Physics, Nov.2-6, 2009, Atlanta, Georgia (2009).
[10] H. Naitou, Y. Yamada, K. Kajiwara, W.W. Lee, S. Tokuda and M. Yagi, submitted to Plasma Science and Technology.
[11] H.A. van del Vorst, SIAM Journal on Scientific and Statistical Computing **13**, 631 (1992).