§7. Calculation for Interactions of Ultra Intense Laser with Real Scale Implosion Pplasma using Load Sharing Parallelized PIC Code

Habara, H., Murakami, R., Kawazu, S. (Osaka Univ.), Sakagami, H.

Investigation of ultra intense laser pulse propagation into a-long scale imploded plasma is one of the important subjects for fast ignition laser fusion research. Especially, "super-penetration" fast ignition, in which the ultra intense laser pulse is directly irradiated into the imploded plasma, requires understanding of the propagation from underdense to overdense plasmas, and of the energy and momentum distributions of fast electrons created during the propagation. For this purpose, a particle-in-cell (PIC) plasma simulation is widely used because it can treat the dynamics of energetic particles with relativistic energies. However, significantly huge computer resources are required for the real scale simulation because the imploded plasma has a very long plasma scale and very high plasma density in 3D space. In order to overcome this problem, we applied a new method for the parallelization, Oh!Help[1)] into our PIC code. Until last year's collaborative research, we have already demonstrated the plasma channel formation in near critical density plasma with several 100μm-length up to critical density by injecting an ultra intense laser light. However, due to time limit for one run in the older plasma simulator, we could only calculate up to 5 ps, in which the laser cannot propagate into the whole plasma region (~70%) even more than 8000 CPU was used.

From this reason, we conducted a further optimization of our code. Especially, we reconstructed the code on the new "plasma simulator" system at NIFS. In order to evaluate the modification, we prepare 2541 x 1261 grid size simulation box. Also we put a plasma slab as shown in Fig. 1. The size of plasma is 100 x 80 μm and has a density ramp from 10^{20}cm^{-3} to 10^{21}cm^{-3} with exponential increase. An ultra intense laser light, whose laser intensity is 4×10^{19}W/cm^2 in 1μm wavelength with 20fs pulse duration in Gaussian profile, is injected from the left boundary of the simulation box. However, although the simulation required 5-ps for 10 hours in the previous system as mentioned above, the run in the new system ends only at 1-ps even the same input parameter was used. With the discussions with Fujitsu technicians, we totally checked the code to find the temporal bottleneck. As the results, significant time loss was found at the system call for time stamp, where there was no delay in the previous system. We replaced this function as a similar function, and also optimized the process of reading or writing of arrays. Using these revisions, the revised program shows near 21% faster than the code without optimization when a small simulation setup is tested.

In addition, we added a function enabling restart from the end-point of the previous run. We use two different algorisms; the first one is redistributing the particles at the restarting by Oh!Help itself. One of the unique feathers of Oh!Help is that the each node has a two particle regions, the primary and the secondary particles in order to share with busy nodes to help. At the end of the calculation, all particle information is gathered in the primary region. In the first case this gathered particle and field information is simply outputted. Then Oh!Help redistribute the particles as it does at the beginning. Although this is very simple and easy to modify the code, but we found that the processes of reading and writing the data take too long time. The second way is saving the primary and secondary particle data separately, and reading each data at the restarting. For this method, we modified the communication process group of MPI as it was saved in the previous run.

Using these modifications on the new system, we can conduct a full-scale laser-imploded plasma simulation with reasonable time scale.

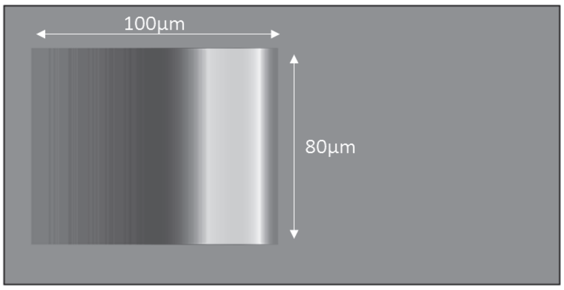

Fig. 1. Simulation setup. The pink region indicate the vacuum in the simulation box.

1) Nakashima, H. et al., ICS'09 (2009) 90-99.