Research on Acceleration of Particle Simulation

Report Number: R22ETET30
Subject Category: Skills Acquisition System
URL: https://www.jss.jaxa.jp/en/ar/e2022/20886/

Responsible Representative

Yasuhiro Mizobuchi, Aviation Technology Directorate, Aircraft Lifecycle Innovation Hub

Contact Information

Takaaki Miyajima, Department of Computer Science, School of Science and Technology, Meiji University(takaaki.miyajima@cs.meiji.ac.jp)

Members

Aoto Abe, Kazumi Kayajima, Dai Wada

Abstract

Acquiring advanced numerical analysis methods such as particle methods

Reasons and benefits of using JAXA Supercomputer System

Scalability is a major issue for the MPS method as it grows in scale, and the JAXA supercomputer was used to understand the current state of P-Flow and to optimise the computation and communication.

Achievements of the Year

With the large-scale simulation of the MPS method in mind, the following computational and communication optimisations were carried out.

- An optimisation of computation: comprehend the current status of the neighbouring particle list and performance prediction.

One of the bottlenecks in the processing time of the MPS method is a process called neighbour particle search. This process requires random memory access and accounts for 60-75% of the total simulation time at most. The current P-Flow solves this problem by using a neighbouring particle list. The specific implementation was in the form of a list with numbers assigned to particles and indirect access to each physical quantity based on these numbers. The speed-up rate when using the ideal list for CPUs was estimated to be 16 times faster for neighbourhood particle search and just under 3.5 times faster overall. The list structure will be improved aiming for these values in the future.

- An optimisation of communication: use of one-sided communication

At present, P-Flow uses MPI_alltoall to share the number of communicating particles between nodes. Therefore, it was obvious that this part would become a bottleneck in processing time as the number of nodes increased. To address this problem, we proposed a method in which MPI_alltoall is replaced by MPI_put, so that only the necessary communication is performed. To effectively validate the proposal, a sample programme was run to compare the execution times of MPI_alltoall and MPI_put using the information communicated between the processes used to execute MPI_alltoall in P-Flow. The results showed that in most cases MPI_put tended to be faster than the execution time of MPI_alltoall. This is thought to be due to the fact that the total number of communications was reduced by using MPI_put rather than MPI_alltoall. As future tasks, it is necessary to conduct similar experiments on other multi-node parallel computer systems and to conduct similar experiments using other MPI implementations to evaluate the generality of this implementation. It is also necessary to examine whether the speed-up can be achieved by replacing not only MPI_alltoall but also the entire communication with other MPI implementations.

Publications

N/A

Usage of JSS

- **Computational Information**

Process Parallelization Methods	MPI
Thread Parallelization Methods	N/A
Number of Processes	1 - 72
Elapsed Time per Case	1500 Second(s)

- **JSS3 Resources Used**

Fraction of Usage in Total Resources*1(%): 0.00

Details

Computational Resources		
System Name	CPU Resources Used (core x hours)	Fraction of Usage*2(%)
TOKI-SORA	225.52	0.00
TOKI-ST	193.31	0.00
TOKI-GP	0.00	0.00
TOKI-XM	0.00	0.00
TOKI-LM	2,418.52	0.16
TOKI-TST	0.00	0.00
TOKI-TGP	0.00	0.00
TOKI-TLM	0.00	0.00

File System Resources		
File System Name	Storage Assigned (GiB)	Fraction of Usage[*2] (%)
/home	30.00	0.03
/data and /data2	300.00	0.00
/ssd	300.00	0.04

Archiver Resources		
Archiver Name	Storage Used (TiB)	Fraction of Usage[*2] (%)
J-SPACE	0.00	0.00

[*1]: Fraction of Usage in Total Resources: Weighted average of three resource types (Computing, File System, and Archiver).

[*2]: Fraction of Usage：Percentage of usage relative to each resource used in one year.

- **ISV Software Licenses Used**

ISV Software Licenses Resources		
	ISV Software Licenses Used (Hours)	Fraction of Usage[*2] (%)
ISV Software Licenses (Total)	0.00	0.00

[*2]: Fraction of Usage：Percentage of usage relative to each resource used in one year.