MPI Performance Analysis in Vlasiator: Unraveling Communication Bottlenecks

SC23 Denver, CO | i am hpc.

Jennifer Faj Department of Computer Science, KTH Royal Institute of Technology Stockholm, Sweden faj@kth.se

Jeremy J. Williams Department of Computer Science, KTH Royal Institute of Technology Stockholm, Sweden jjwil@kth.se

Ivy B. Peng Department of Computer Science, KTH Royal Institute of Technology Stockholm, Sweden bopeng@kth.se

Urs Ganse Department of Physics, University of Helsinki, Helsinki, Finland urs.ganse@helsinki.fi

Markus Battarbee Department of Physics, University of Helsinki, Helsinki, Finland markus.battarbee@helsinki.fi

Yann Pfau-Kempf Department of Physics, University of Helsinki, Helsinki, Finland yann.kempf@helsinki.fi

Leo Kotipalo Department of Physics, University of Helsinki, Helsinki, Finland leo.kotipalo@helsinki.fi

Minna Palmroth Department of Physics, University of Helsinki, Helsinki, Finland minna.palmroth@helsinki.fi

Stefano Markidis Department of Computer Science, KTH Royal Institute of Technology Stockholm, Sweden markidis@kth.se

Funded by the European Union

MOTIVATION

Large-scale plasma simulations are critical for understanding plasma dynamics in various plasma environments, from fusion devices and accelerators to space and astrophysical systems. In order to efficiently perform such simulations on upcoming Exascale machines, challenges with regard to parallelization, data distribution and communication need to be addressed. This work aims to provide insights into communication and the impact of load imbalance of the Vlasiator simulation code, and to indentify potential optimization techniques.

VLASIATOR

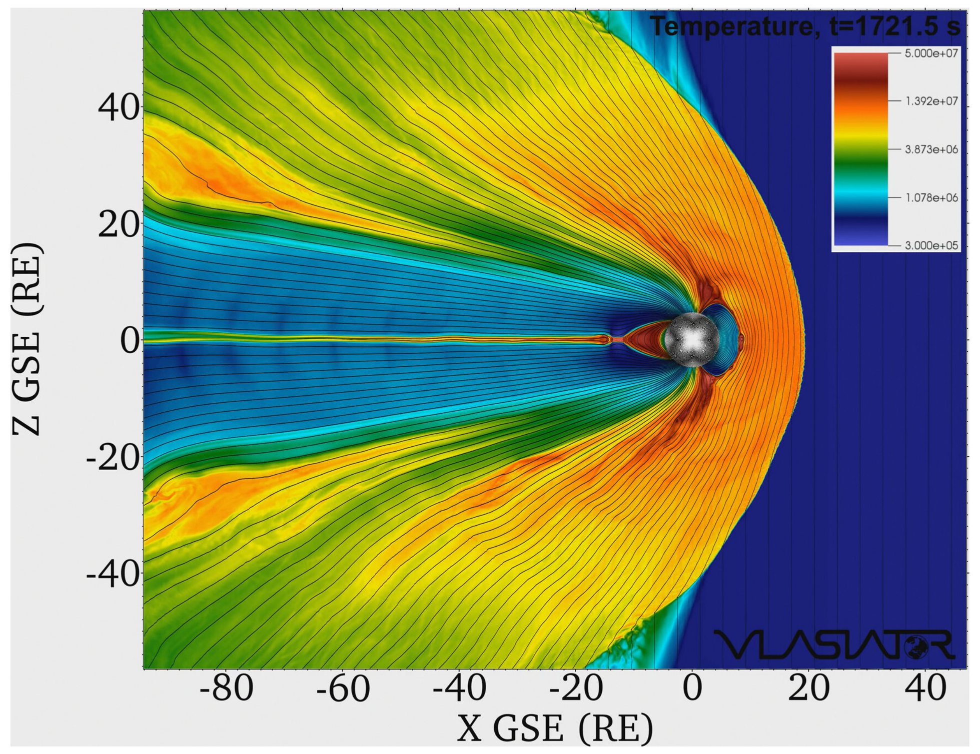

Fig 1: Vlasiator simulation

Vlasiator [1,2,3] is one of most powerful and established parallel C++ codes for the highly accurate simulation of magnetospheric and solar wind plasmas with application to space weather. The code uses a hybrid Vlasov solver, operating on a 6-dimensional simulation grid. The code is parallelized using both MPI and OpenMP. Additionally, libraries are utilized to improve the load balance across MPI nodes (Zoltan) as well as implement adaptive mesh refinements (DCCRG).

METHODOLOGY

Testcase: Magnetosphere 3D
Input size: ~4*10^7
Simulated time: 20 seconds (~800 time steps)
Experiments: scaling from 8 to 32 nodes

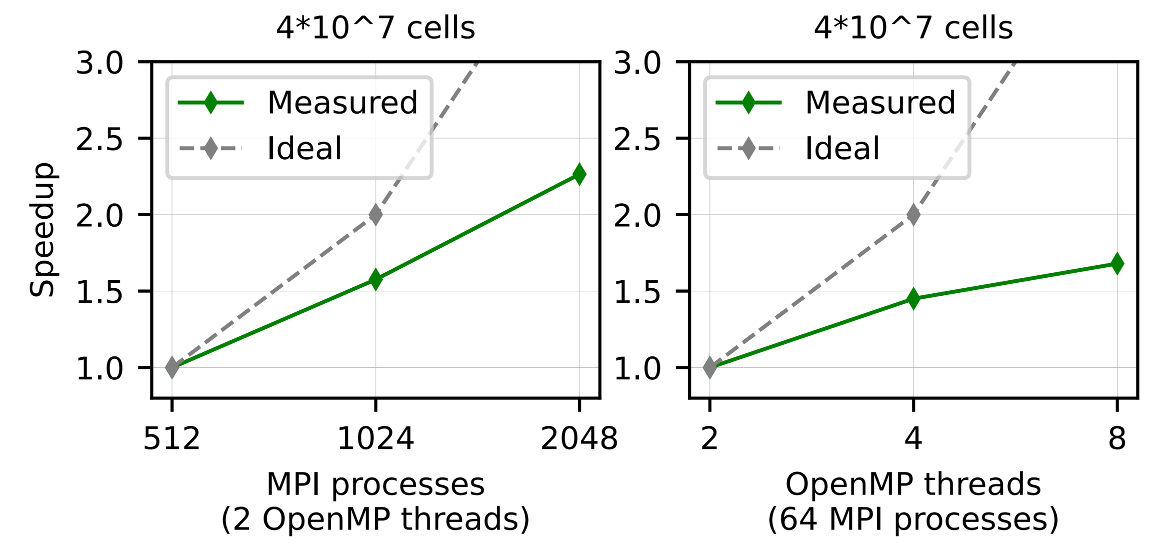

Fig 2: Multi-node scaling with OpenMP and MPI

Profiling:

CrayPat: Determine critical functions

IPM: Analyse MPI communication

CRAYPAT ANALYSIS

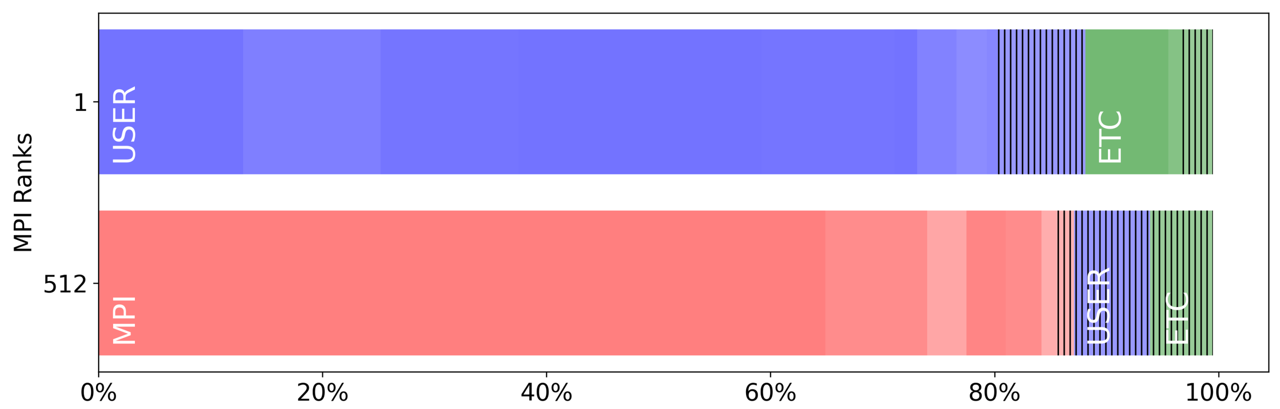

Fig 3: CrayPat analysis

Grid Size	AMR Cells	Nodes	MPI	OpenMP	%Comm
~85k	~13k	8	512	2	82%
			64	16	51%
~830k	~137k	16	128	16	48%

Tab 1: Communication ratios

IPM ANALYSIS

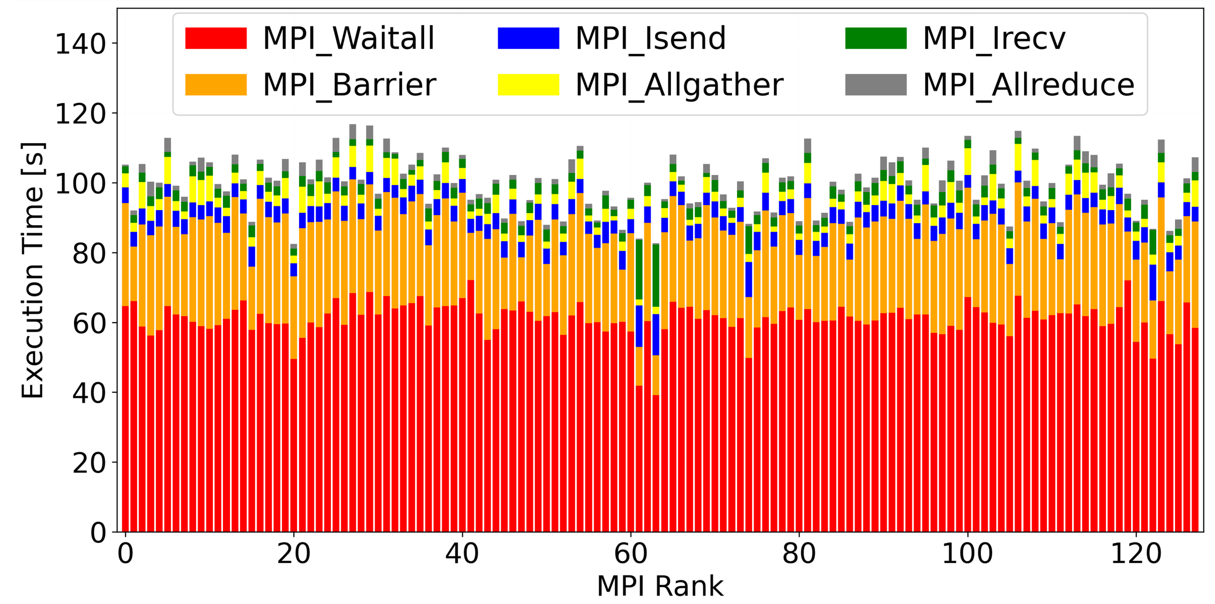

Fig 5: Breakdown of MPI calls.

The breakdown of MPI function calls shows that 86% of MPI time is spent in two functions: MPI_Waitall and MPI_Barrier.

Furthermore, it can be observed that tremendous amounts of data are communicated between ranks.

A breakdown of the message size distribution reveals messages of up to 256MB and a significant amount of time being spent in sending messages between 16-64MB.

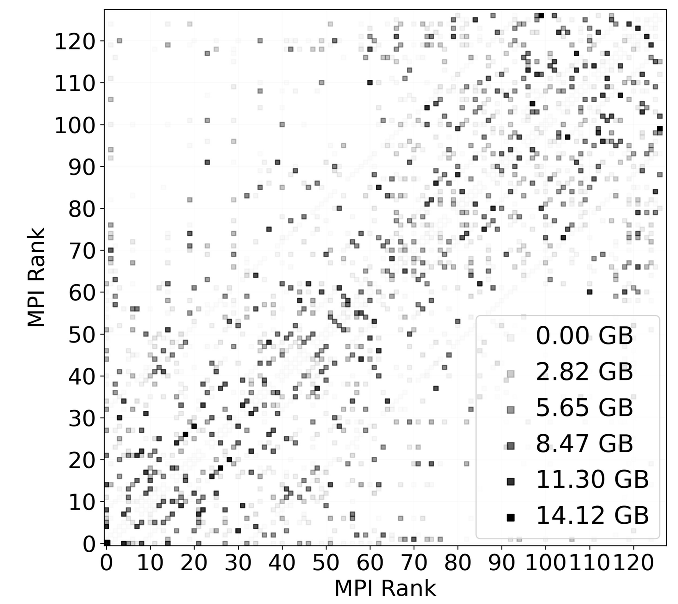

Fig 6: Topology

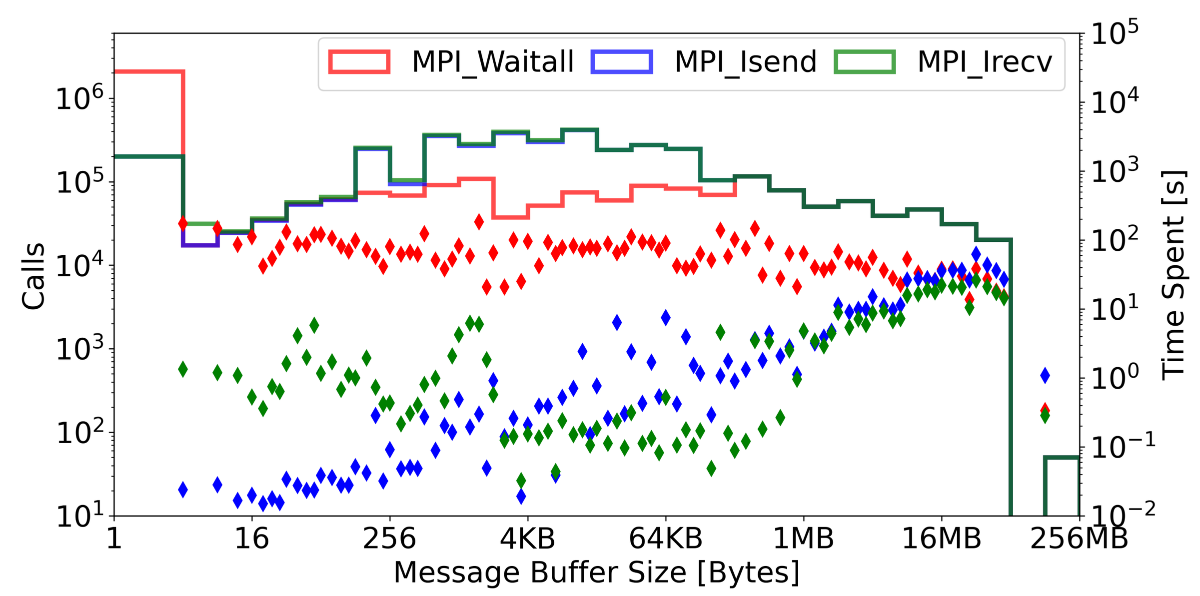

Fig 7: Message size distribution

CONCLUSIONS

In this work, we analyzed the MPI performance of the plasma simulation code Vlasiator, using CrayPat and IPM. We showed that Vlasiator is a communication-bound application, with communication percentages varying from 48% to 82% of the total time. We note that the highest costs within this MPI communication are associated with synchronization and process imbalances [4]. To eliminate intra-node communication and improve the overall performance of the code, we found that an increased integration of OpenMP is a promising approach. To elaborate further specific optimization techniques, an in-depth analysis of load imbalance sources in the complex spatial domain (Fig. 8) of the simulation is required.

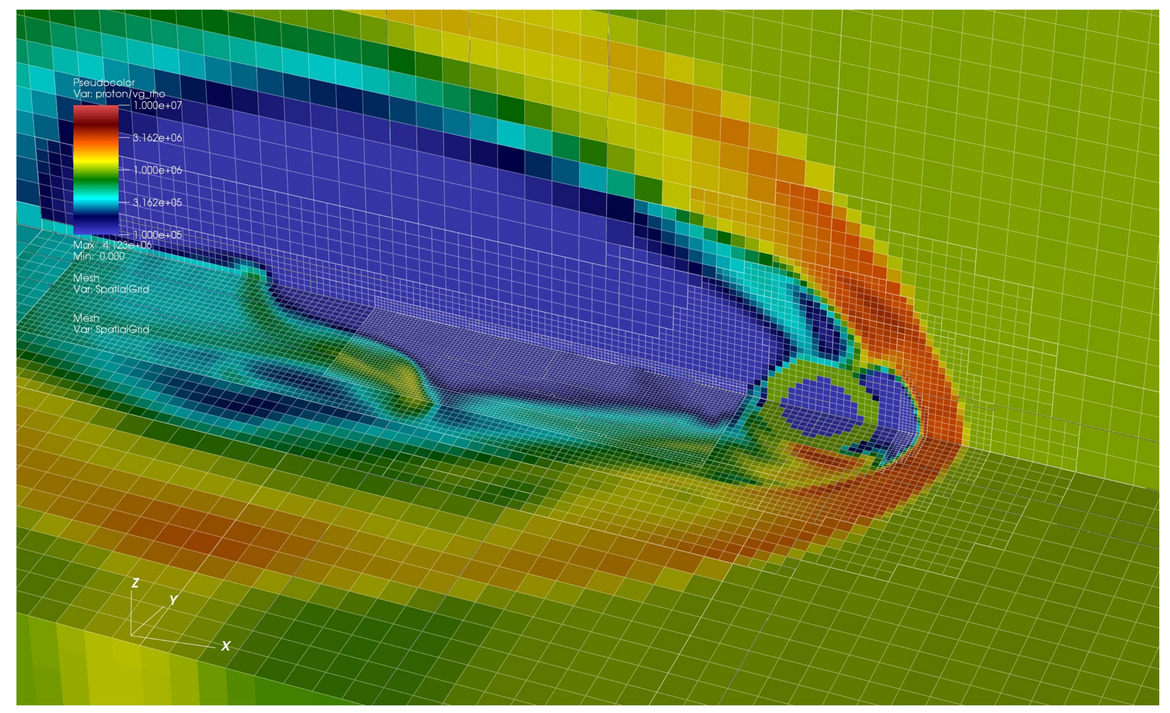

Fig 8: Vlasiator domain space decomposition

References

[1] Minna Palmroth. 2022. Daring to think of the impossible: The story of vlasiator. Frontiers in Astronomy and Space Sciences 9 (2022), 210.
[2] S. von Alfthan, D. Pokhotelov, Y. Kempf, S. Hoilijoki, I. Honkonen, A. Sandroos, and M. Palmroth, “Vlasiator: First global hybrid-vlasov simulations of earth's foreshock and magnetosheath,” Journal of Atmospheric and Solar-Terrestrial Physics, vol. 120, pp. 24–35, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364682614001916
[3]: M. Palmroth, U. Ganse, Y. Pfau-Kempf, M. Battarbee, L. Turc, T. Brito, M. Grandin, S. Hoilijoki, A. Sandroos, and S. von Alfthan, “Vlasov methods in space physics and astrophysics,” Living Reviews in Computational Astrophysics, vol. 4, no. 1, p. 1, 2018.
[4] Ivy Peng et al. 2015. The cost of synchronizing imbalanced processes in message passing systems. In 2015 IEEE Cluster. IEEE, 408

KTH VETENSKAP OCH KONST
HELSINGIN YLIOPISTO HELSINGFORS UNIVERSITET UNIVERSITY OF HELSINKI
PLASMA PEPSC

This work has received funding from the European High Performance Computing Joint Undertaking (JU) and Sweden, Finland, Germany, Greece, France, Slovenia, Spain, and Czech Republic under grant agreement No 101093261

Plasma-PEPSC
UoH-Research
Vlasiator