Exploring Performance of Domain Decomposition Strategies for Monte Carlo Radiation Transport

Students: X. Chen[1,2], D. Huff[1,3], P. Karpov[1,4] • Advisors: R. Wollaeger[1], G. Rockefeller[1], B.K. Krueger[1]

[1]Los Alamos National Laboratory,[2]University of New Mexico,[3]New Mexico Institute of Mining and Technology,[4]University of California, Santa Cruz

Abstract

SuperNu[1] is a Monte Carlo (MC) radiation transport code for simulating light curves of explosive outflows from supernovae. The MC transport step is domain replicated. To enable scaling on next-generation HPC systems, we are implementing the recursive coordinate bisection approach of domain decomposition for the opacity calculation. Then, we plan to propagate the decomposition to other steps in the simulation and construct a communication infrastructure to support the decomposition. In this poster we demonstrate the results of two communication schemes: the Improved KULL and Improved Milagro algorithms[2].

What is SuperNu?

Supernovae are stellar explosions, resulting in a cloud of gas called a nebula (Fig.1). The intent of SuperNu is to efficiently produce light curves and spectra for such nebulae. For example, Fig.2 present a light curve of a core-collapse supernova with a jet along one direction, modeled in 3D.

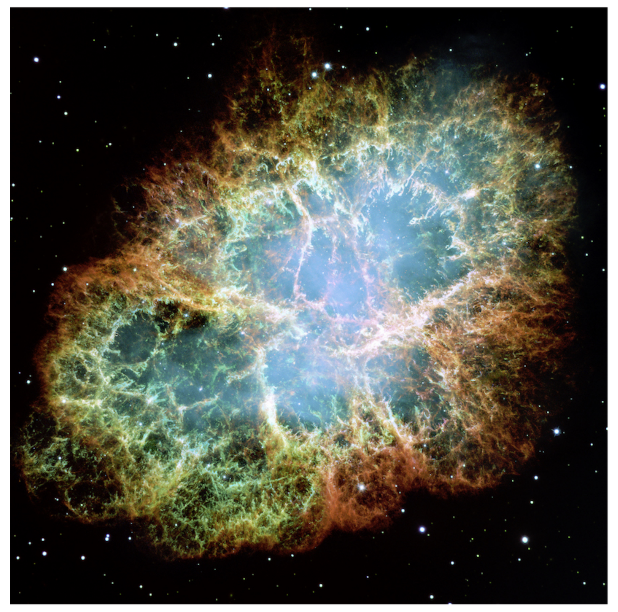


Figure 1: Crab Nebula, pictured above, was formed as result of a supernova[4]

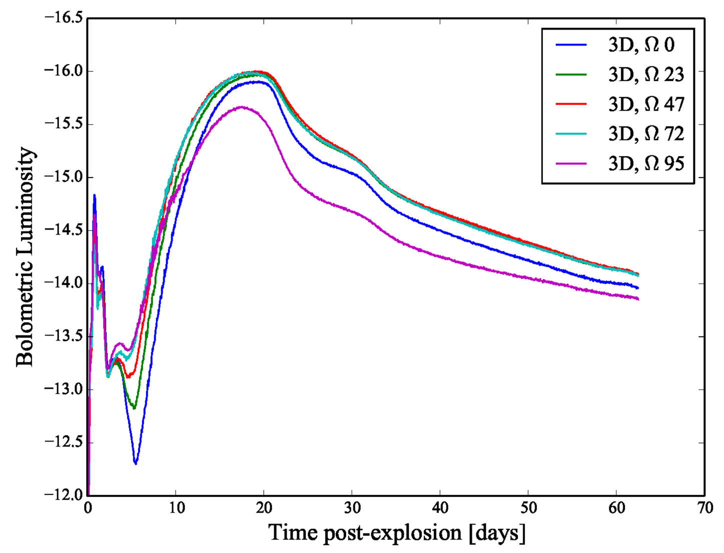


Figure 2: Light curve of a core-collapse supernova, where Ω is the viewing angle

Limitations

The previous particle transport approach involved replicating the whole domain on each rank, which limits problem size, memory wise.

The implementation also included an opacity calculation, where the domain was decomposed into strips (Fig.3). If this domain decomposition were applied to the transport step, there would likely be needlessly high communication between the ranks, as photons are less likely to stay in the part of the domain attributed to any one rank. This is due to the the low volume to surface area ratio of the strip decomposition.

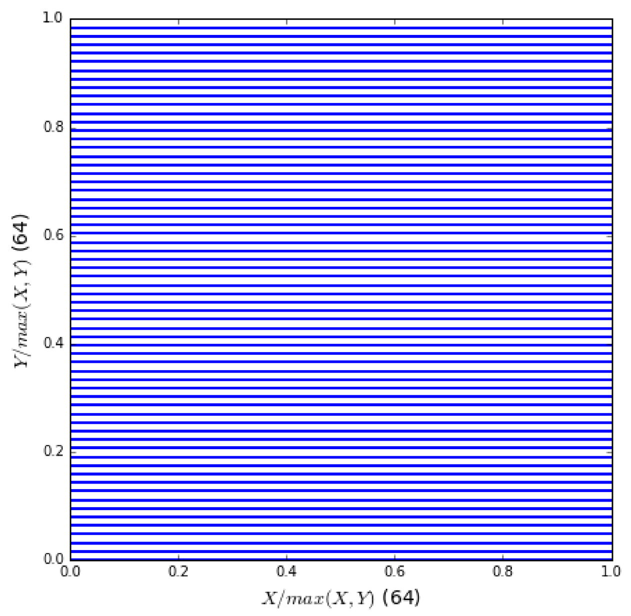
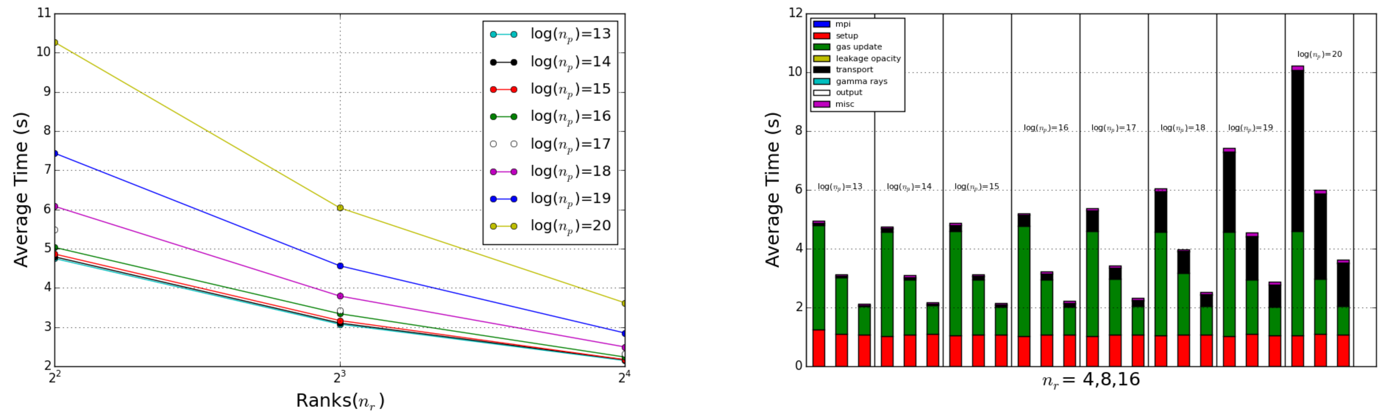


Figure 3: Particle sets in a strip configuration, with one strip of particles assigned to each of the 64 MPI rank.

Previous Performance



(a) total run time (b) time per simulation step

Figure 4: Benchmarking SuperNu on Wolf at LANL, with n_p=particle number

The graphs above show how SuperNu behaves with different run conditions, modeling W7 Type Ia supernova. The parameter varied was the number of particles within the simulation. Fig.4a represents the overall time it takes to finish a given simulation. Fig.4b looks further into the steps, and identifies the time it takes to complete each step. This allows us to identify the most time-consuming phases, such as the transport step discussed previously.

Domain Decomposition

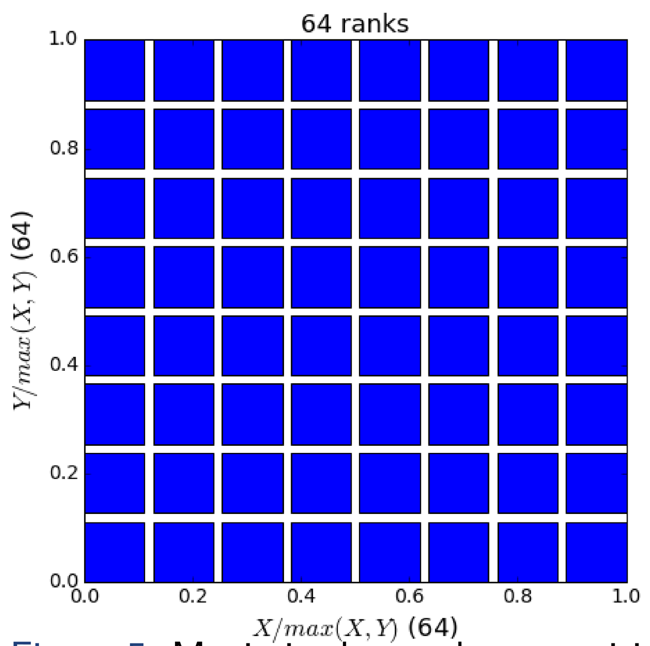


Figure 5: Maximized-area decomposition

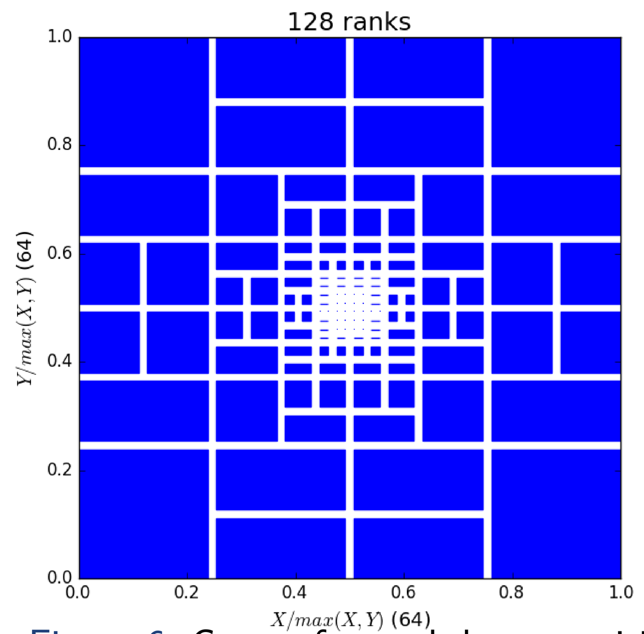


Figure 6: Center-focused decomposition

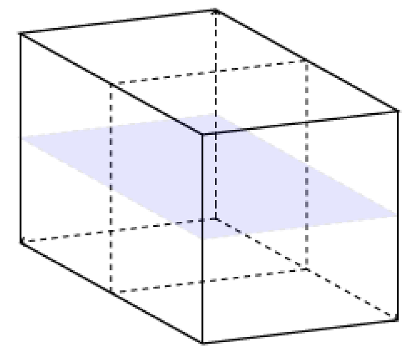


Figure 7: RCB in 3D

The spatial domain is decomposed into geometrically consecutive regions, which maximize area(2D) for a given perimeter(2D), following the recursive coordinate bisection algorithm (RCB) (Fig.5). Such a decomposition minimizes the frequency with which particles cross sub-domain boundaries, reducing the need for communication between the ranks. For supernova problems, the source is typically at the center. Hence, in order to achieve better load balance, a center-focused domain decomposition was implemented, as in Fig.6.

Communication Methods

We had to address both particle-passing between rank-assigned sub-domains and completion messages in a domain-decomposed scheme.

Improved KULL

The Improved KULL algorithm uses non-blocking communications to exchange particles with neighbors which are also ready to communicate. 'At the last stage, all of the worker-ranks communicate back to the head-rank indicating they completed the step, and only then the head-rank broadcasts the global 'complete' message.[1]

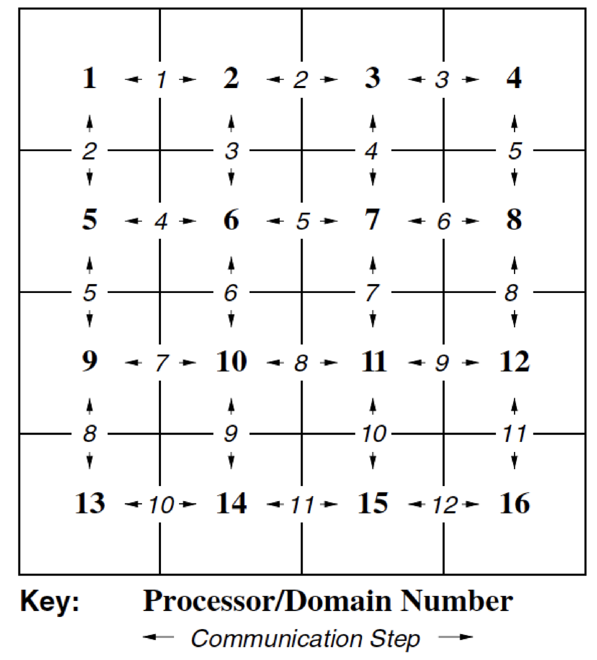


Figure 8: Non-blocking exchange

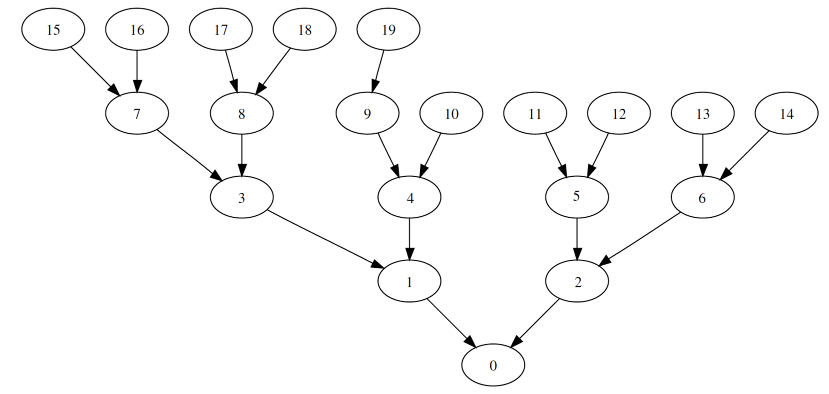
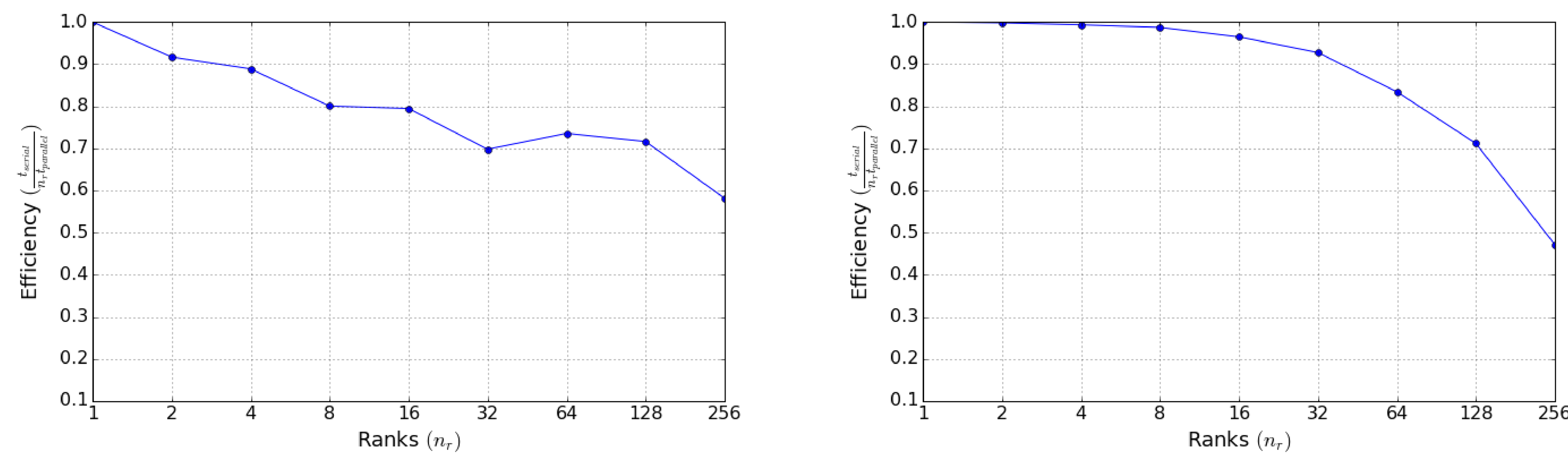


Figure 9: Binary tree pattern of Improved Milagro[2]

Improved Milagro

Unlike KULL's 'send to all'-'gather from all' scheme, the Improved Milagro uses a binary tree communication pattern (Fig.9). Short messages passing like the asynchronous *particles completed messages* perform better in this pattern.[2]

Testing the Methods



(a) Improved KULL (b) Improved Milagro

Figure 10: Efficiency results of the mini-applications

Fig.10 presents the scaling study of two simplistic load-balanced particle transport applications, exercising the Improved KULL and Improved Milagro methods. Both applications were run on Blue Waters, with a 2D 1024x1024 resolution and one particle per cell. However, the two results are not meant to be directly compared due to preliminary implementation differences.

Conclusion

The domain decomposition method, based on the RCB algorithm, will aid SuperNu to simulate on larger domains. To pass Monte Carlo particles between the resulting sub-domains, we explored the Improved KULL and Improved Milagro algorithms. These algorithms employ non-blocking message passing to allow computation during MPI communication. We tested the new domain-decomposition, and communication methods on LANL's local clusters and Blue Waters supercomputer, which will next be implemented within SuperNu.

References

- 1) R.T. Wollaeger et al. 2013, ApJS 209, 37
- 2) T.A. Brunner et al, 2006, JCoPH .212..527B
- 3) R.T. Wollaeger, D.R. van Rossum. ApJ. 214 (2014) 28
- 4) J. Hester, A. Loll, NASA, ESA, (Arizona State University)

Acknowledgments

The authors would like to thank:

- Parallel Computing Summer Research Internship at ISTI, LANL
- LANL Institutional Computing Resources
- Blue Waters sustained-petascale computing of UI at Urbana-Champaign and NCSA.