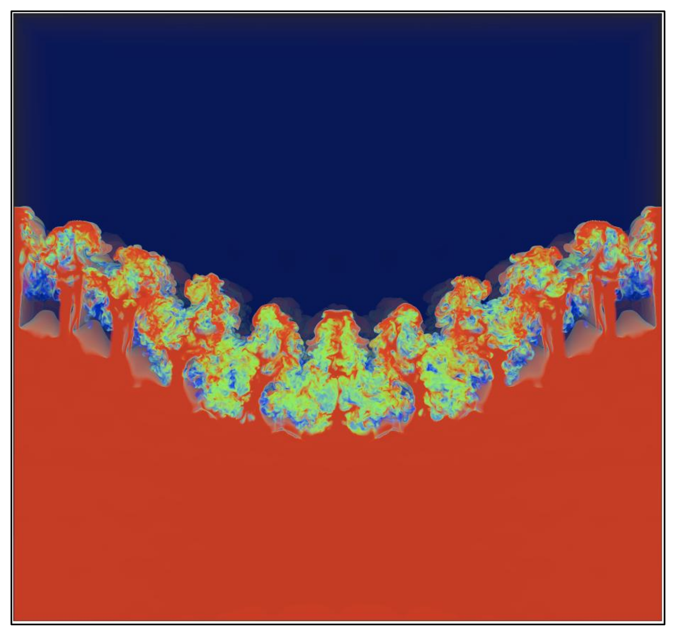
EX19701（大阪大学推薦課題）

Marcus Wallden（大阪大学）

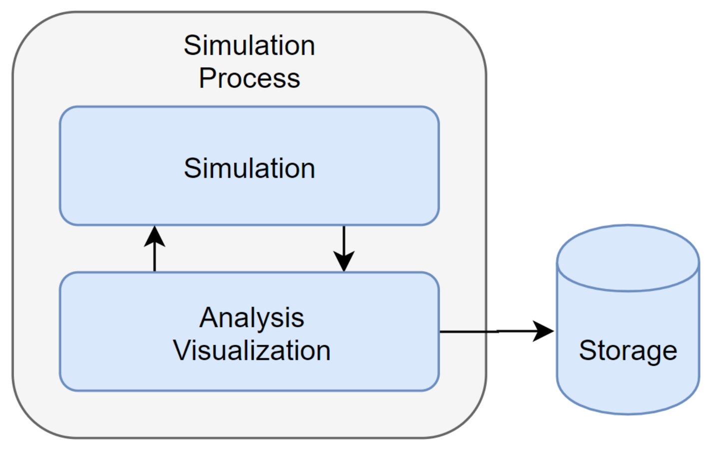
Effective Load Balancing for Distributed Large-Scale Volume Rendering Using a Two-layered Group Structure

Overview

Scientific simulations are used to analyze complex phenomena in many different areas of research. I/O bandwidth in modern supercomputers is advancing at a significantly slower rate than the compute capacity. Generated simulation data can as such not be saved to permanent storage without significant performance drawbacks. In many modern applications, less than 1% of the simulated data can be stored.

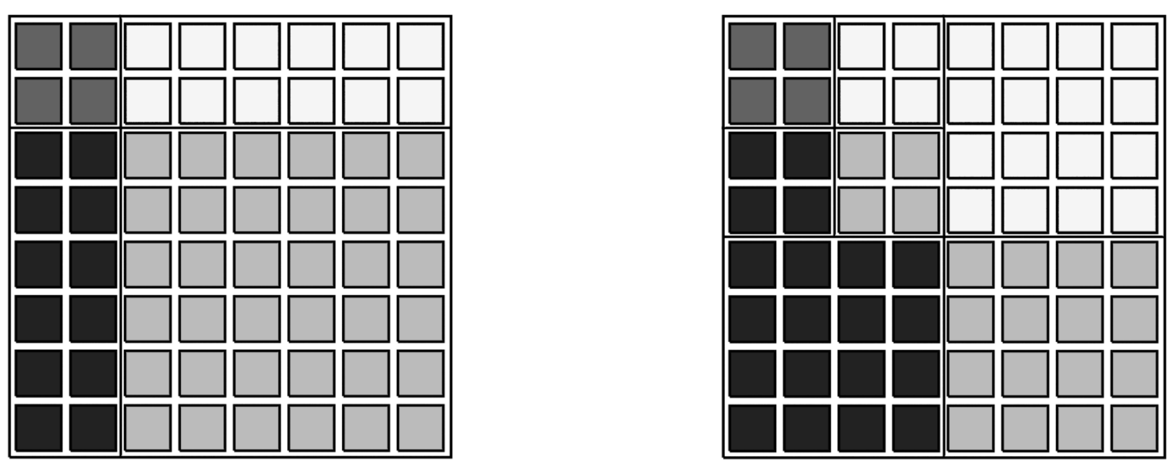


Researchers have tried to decrease the amount of stored data by lowering the resolution or by only saving certain parts of the data. However, simulations can generate terabytes of data each time step. Future simulations are bound to generate even larger data sets as they become more detailed and complex. As such, other methods are required to save the generated data. A common approach is to analyze and visualize the data as it is generated and then store the results, which is orders of magnitude smaller, to permanent storage. This approach is called *in-situ co-processing*. The purpose of our research is to develop new co-processing techniques to accelerate the visualization and analysis of scientific data sets. Specifically, we focus on two aspects: identifying what generated data is important and how to distribute analysis and visualization tasks across available compute nodes.



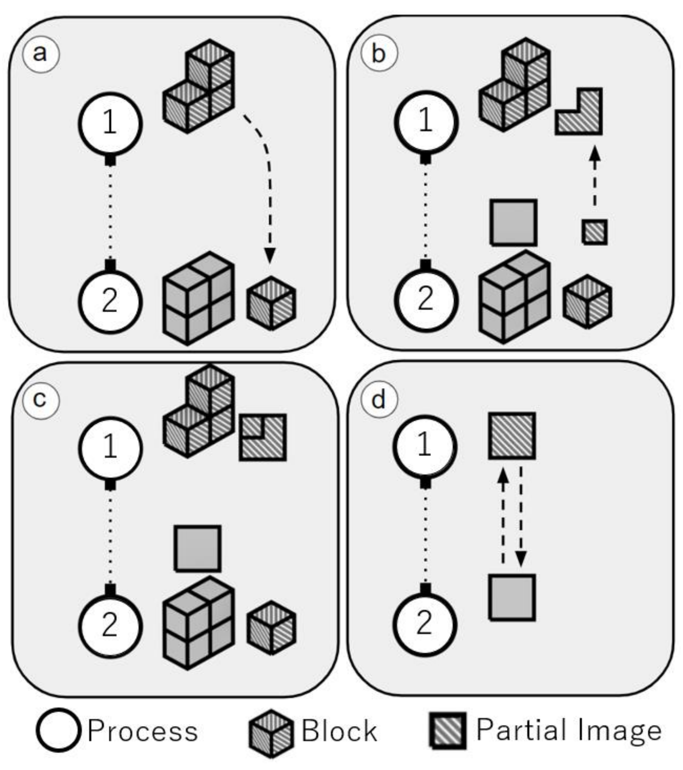
Load Balancing

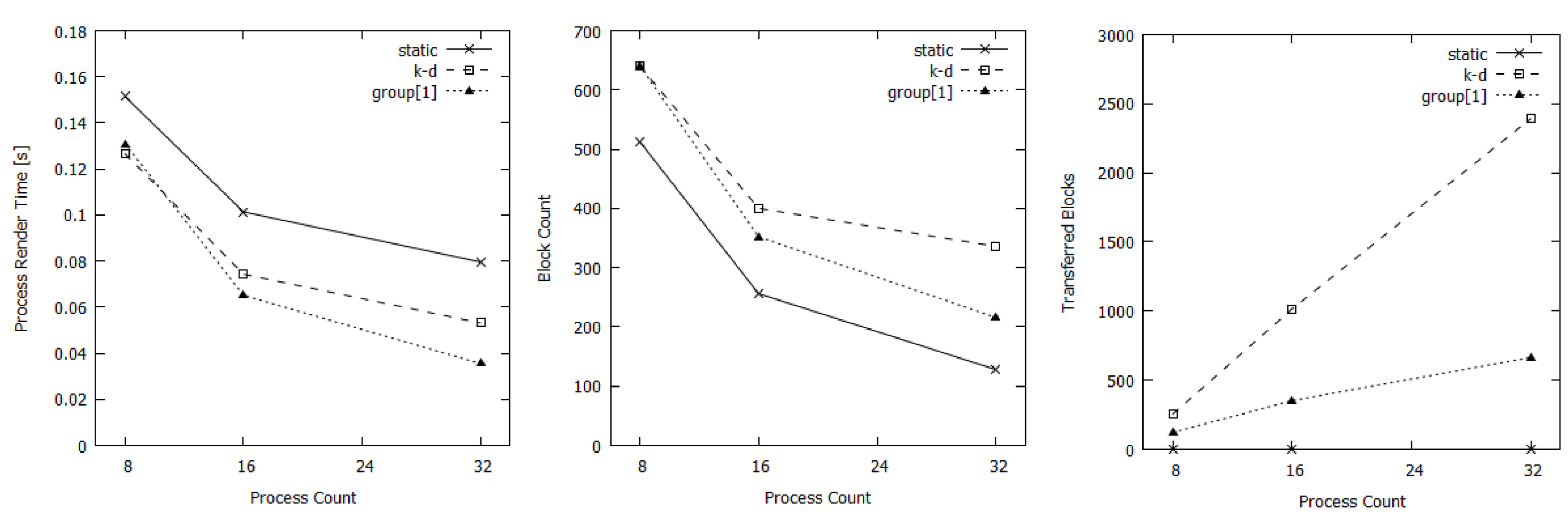
The time required to visualize or analyze various regions of a data set can vary between processes. As such, using load balancing techniques to redistribute generated simulation data among available processes can in many cases effectively minimize the co-processing computation time. On the other hand, redistributing the load could cause a high memory imbalance. Processes can easily exceed the available amount of memory when handling terabyte-sized data sets.



The state of the art load balancing technique for 3D volumetric data sets is the *k*-d tree technique. However, the *k*-d tree technique has a strict load balancing scheme which can result in a high memory imbalance, making it unsuitable for large-scale load balancing. We propose a two-layered group load balancing structure and a novel image compositing pipeline to allieve this problem. The novel pipeline enables processes to handle data from arbitrary regions of the volume, which is not possible in existing techniques.

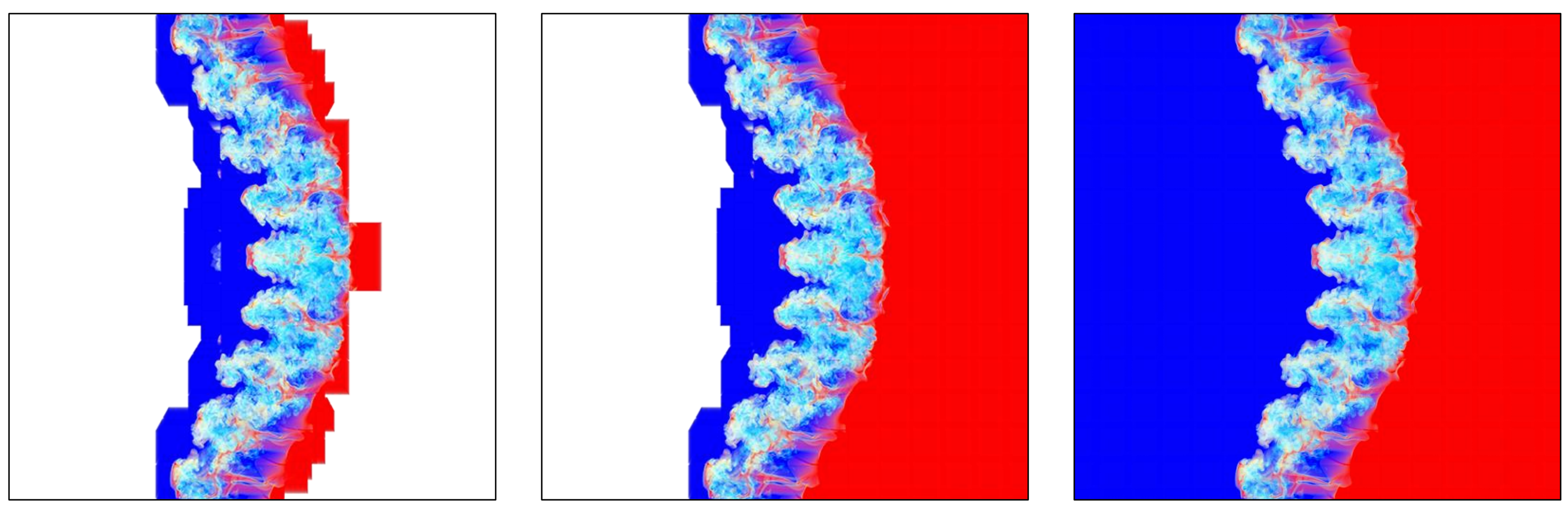
Our evaluation shows that the proposed two-layered technique achieves similar computation times as the *k*-d tree, whilst significantly reducing the worst-case memory usage and the amount of transferred data. As such, the proposed technique can be used to lower the memory imbalance during large-scale co-processing, without negatively affecting the computation time.





ATCO: Adaptive In-Transit Co-Processing

In-transit is a specific case of in-situ co-processing where co-processing is performed on a separate group of compute nodes. In-transit approaches allow co-processing to be performed asynchronously whilst the next time step is generated. However, in-transit approaches generally also require all generated simulation data to be transferred to the used co-processing nodes. The simulation can consist of many terabytes of data each time step, making this process very time consuming. We are developing a framework called ATCO (short for adaptive in-transit co-processing) to solve this problem. We note that a simulated phenomenon is often concentrated in a subset of the generated data set. As such, unimportant data can be compressed, reduced or even removed without affecting the detail of the studied phenomenon.



An important consideration in the development of ATCO is that it has to be usable in different types of applications and in any type of analysis pipeline. We have developed a *reduction pipepile*, which consists of multiple filters and data probes. All elements of the pipeline can easily be extended by simulation researchers based on the underlying use case. We are currently collaborating with Professor Dimitris Drikakis of the University of Nicosia. He is the creator of the state-of-the-art fluid simulation software CNS3D. Together we are investigating the advantages of the ATCO framework and are investigating how it can be further improved. Currently we are working towards running ATCO coupled with a large-scale simulation of the Richtmyer Meshkov instability. We are very grateful to be able to use the Octopus cluster system in the Cybermedia Center at Osaka University. This experiment will result in over 10 terabytes of data per time step, spread over four billion unique points of data, running on up to 6144 threads.