Project ID: Jh160014-ISH

Data Locality Optimization Strategies for AMR Applications on GPU-accelerated Supercomputers

Mohamed Wahib (RIKEN AICS)

Abstract

As the memory hierarchies of supercomputers get more complex, improving the performance of applications bounded by the data movement throughput becomes challenging. For example, TokyoTech's TSUBAME, the machine we intend to use, includes different data transfer bottlenecks that complicate the domain decomposition and load balancing: inter-node connection, intra-node multi-GPU connections, and GPU memory hierarchy. Our main research target is to study the impact of hierarchical memory on the optimization strategies used in specific classes of memory-bound applications. Further, we are currently expanding our study to investigate the data movement challenge for Adaptive Mesh Refinement (AMR) applications. The resources we apply for in this proposal are to be specifically used for AMR applications to analyze the scalability of data locality optimizations that specialize the computation on the nearest memory level. Applying those optimizations can improve the performance of AMR applications in systems having complex memory hierarchies as in TSUBAME.

The AMR method is widely used by a diversity of scientific applications. AMR methods are complex, and suffer from bottlenecks at different levels of data movement. We are motivated by challenges to AMR applications imposed by changes in the memory hierarchy. In our experiments, we demonstrated the potential of a data-centric approach for AMR applications. Further, were able to demonstrate that our data locality approach in AMR can scale to 3,640 GPUs for three real-world applications.

1. Basic Information

(1) Collaborating JHPCN Centers

Tokyo Tech (TSUBAME2.5)

(2) Research Areas

- ☐ Very large-scale numerical computation
- ☐ Very large-scale data processing
- ☐ Very large capacity network technology
- ☐ Very large-scale information systems

(3) Roles of Project Members

(4) Mohamed Wahib, RIKEN AICS (PI), Naoya Maruyama, RIKEN AICS (Participant), Takayuiki Aoki TokyoTech (Participant)

2. Purpose and Significance of the Research

The main purpose of this work is to identify and optimize for data locality in AMR applications running in GPU-accelerated supercomputers. First, we devised a performance model for the data-centric AMR method. This provides bases for guiding the problem decomposition and load balancing, if needed. Numerous performance models exist in HPC. However, to the author's knowledge, analytical optimization for data locality in AMR, which influences problem decomposition is not covered in literature. Second, optimized

data-centric AMR implementations for real world applications, including the phase-field simulation. Finally, we tested and evaluated the execution of real world applications at the full scale of TSUBAME. This did not only improve the performance of the tested applications; it also provided a realistic measure of the applicability of the framework to other applications implemented by other researchers.

3. Significance as a JHPCN Joint Research Project

Two factors motivate the application for the proposed work as a JHPCN joint research project: human factor and machine factor. For the human factor, the group under Prof. Aoki at TokyoTech has leading expertise in AMR simulations for scientific applications. Moreover, the same group did highly valued work on fixed mesh phase-field simulations for 3D dendritic growth (2012 Gordon bell award). Introducing the data-centric AMR method to AMR applications, including the phase-filed simulation, is an opportunity for the proposed study.

For the machine factor, the data-centric AMR approach is designed mainly for GPUs. Japan's largest GPU accelerated supercomputer computer, TSUBAME, is hosted at TokyoTech. TSUBAME is an ideal test bed for scalability and performance studies in GPU applications.

4. Outline of the Research Achievements up to FY 2015

We conducted experiments on a single node of multi-GPUs for two different AMR applications. The experiments involved testing different data locality approaches within our AMR framework. Promising results were achieved when a technique for data-centric computation eliminated the CPU-GPU data transfer bottle-neck. More specifically, we utilized a specialization technique by which the CPU specializes on the operations touching the data structures used to manage the mesh (an octree in our framework), while the GPUs specialize on operations that touch the data arrays of the blocks. While this approach requires writing additional GPU CUDA kernels, it eliminates that need to transfer the blocks between the CPU and GPU every time the mesh is evaluated for changes. In comparison to baseline implementations that require the blocks to be transferred to the CPU, we reported speedups of up to 2.21x and 2.83x in data-centric AMR implementations of a hydrodynamics simulation and shallow-waters simulation, respectively [1][2]. Details of the results can be found in the publications listed in section 7.

5. Details of FY 2016 Research Achievements

We introduced a high-level programming framework, named Daino, that provides a highly productive programming environment for AMR. The framework is transparent and requires minimal involvement from the

programmer, while generating efficient and scalable AMR code. The framework consists of a compiler and runtime components. A set of directives allows the programmer to identify stencils of a uniform mesh in an architecture- neutral way. The uniform mesh code is then translated to GPU-optimized parallel AMR code, which is then compiled to an executable. The runtime component encapsulates the AMR hierarchy and provides an interface for the mesh management operations. The framework is publically available at github: https://github.com/wahibium/Daino

We demonstrate the scalability of auto-generated AMR code using three production applications. We compare the speedup and scalability with hand-written AMR of all three applications:

- *Phase-field Simulation*: This application simulates 3D dendritic growth during binary ally solidification.

- *Hydrodynamics Solver*: This solver models a 2nd order directionally split hyperbolic schemes to solve Euler equations.

- *Shallow-water*: Modelling shallow water by depth-averaging Navier-Stokes equations.

In a weak scaling experiment, shown in Figure 1, the run-time for uniform mesh, hand-written AMR, and auto-generated AMR are compared. The following points are important to note. First, more than 1.7x speedup is achieved using Daino suing the full TSUBAME machine, 3,640 GPUs, for the phase-field simulation. This is a considerable improvement considering that the uniform mesh implementation is a Gordon Bell prize winner for time-to-solution. Second, Daino achieves good scaling that comparable to the scalability of the hand-written AMR code.

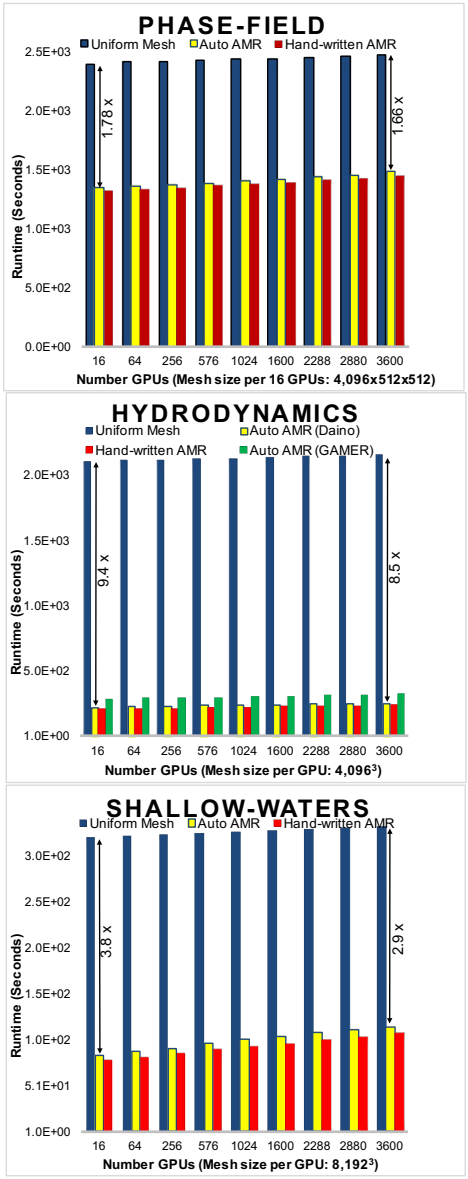

Figure 1: Weak scaling of uniform mesh, hand-written and automated AMR

Figure 2 shows a strong scaling comparison for hand-written and auto-generated AMR against uniform mesh implementation. The code generated by Daino achieves speedups and scalability comparable to optimized hand-written implementations. However, when using more GPUs, reduction in speedup starts to occur as the management of AMR starts to dominate the simulation runtime.

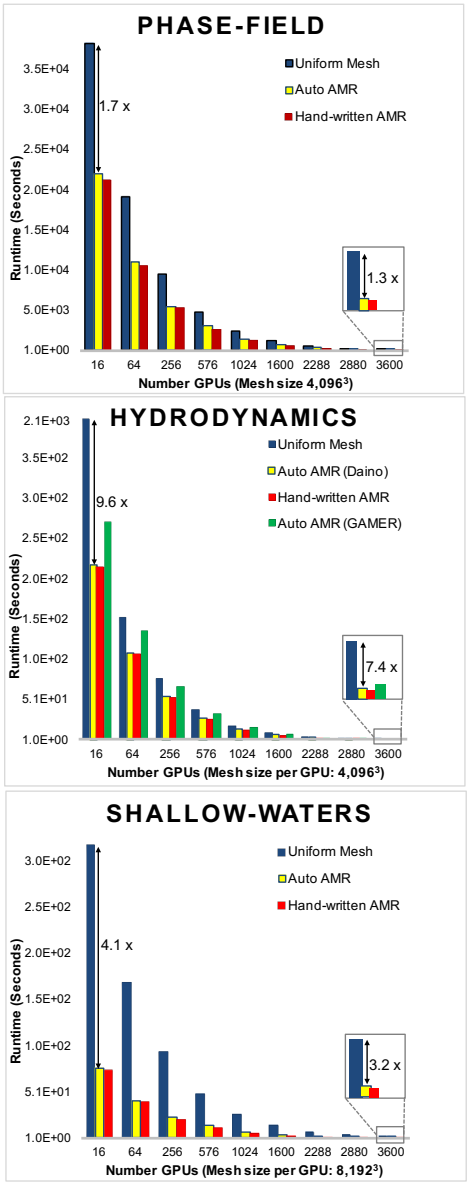

Figure 2: Strong scaling of uniform mesh, hand-written and automated AMR

In the following paragraph, we highlight a specific case when the performance model we derived for our framework was able to help in identifying non-optimal data exchange of ghost layers for blocks residing on different nodes. The Z-order Morton curve used in our octree representation of the mesh is known to produce partitions with high spatial locality. However, protruding sub-domains can cause an MPI rank to communicate with neighbor blocks scattered over a large set of ranks (See illustration in Figure 3).

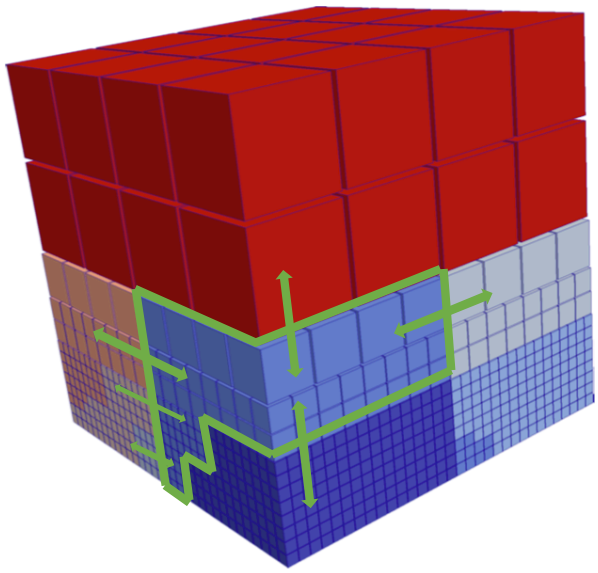

Figure 3: Illustration of inefficient neighbor communication when using Morton Z-curve

We resolved this problem by using an approach that increases the data locality; namely and Morton-level ordering that considers the levels of the octree. While this solution helped reduce the communication bottleneck, it is by no means ideal and we are considering other geometric domain decomposition methods.

6. Progress of FY 2016 and Future Prospects

So far, we have tested the scalability of the auto-generated codes. As for future work, we intend to start a study on the potential of abstracting data-locality optimizations in AMR runtime systems. More specifically, we will conduct a set of experiments to quantify the effect of our data-locality approach on the performance when considering different structured-AMR approaches when applied for different domain decompositions and work load scenarios.

7. List of Publications and Presentations

(1) Journal Papers

N/A

(2) Conference Papers

[1] Mohamed Wahib, Naoya Maruyama, Takayuki Aoki, Automated GPU Kernel Transformations in Large-Scale Production Stencil Applications, SC'16, ACM/IEEE Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, Salt Lake City, US [Acceptance rate: 18%][**Best Paper**]

[2] Mohamed Wahib, Naoya Maruyama, Data-centric GPU-based Adaptive Mesh Refinement, 5th Workshop on Irregular Applications: Architectures and Algorithms, co-located with SC'15 (IA3'15) [Acceptance rate: 25%]

(3) Conference Presentations (Oral, Poster, etc.)

- PADAL16 workshop, Third Workshop on Programming Abstractions for Data Locality Oct 24-26, Kobe, Japan [Invited Talk]
- GTC16 Tokyo, GPU Technology conference Oct 5th, Tokyo, Japan [Invited Talk]
- GTC17 San Jose, GPU Technology conference May 11th, San Jose, CA, USA [Invited Talk]

(4) Others (Patents, Press releases, books, etc.)

- Mohamed Wahib, Naoya Maruyama, Takayuki Aoki, A High-level Framework for Parallel and Efficient AMR on GPUs, Tsubame E-science Journal, April 2017, Vol. 15, pp. 56-64