

GPU acceleration of a non-hydrostatic ocean model using a mixed precision multigrid preconditioned conjugate gradient method

Takateru Yamagishi

Research Organization for Information Science
and Technology

1-5-2, Minatojimaminami-machi,
Chuo-ku, Kobe-shi, Hyogo-ken,
650-0047, JAPAN.
+81-78-599-9511

yamagishi@rist.jp

Yoshimasa Matsumura

Institute of Low Temperature Science,
Hokkaido University

Kita-19, Nishi-8, Kita-ku, Sapporo,
060-0819, JAPAN
+81-11-706-5465

ymatsu@lowtem.hokudai.ac.jp

ABSTRACT

To meet the demand for fast and detailed calculations in numerical ocean simulation, we implemented a non-hydrostatic ocean model on a graphics processing unit (GPU) using fundamental methods. We also evaluated the application of mixed precision calculation to the preconditioning of a Poisson solver. The GPU-implemented model was 4.3 times faster than the execution on a CPU. The application of mixed precision achieves a 16% acceleration of a Poisson solver, without introducing specific errors in the outputs.

1. INTRODUCTION

Oceanic numerical simulations use fast and detailed calculations with a large number of grids to model small oceanic processes. GPU is expected to be able to handle the task, but few studies have been conducted on the role of GPU in ocean models and the experimental research available is insufficient to support the study of oceanic processes [1, 2].

This study aims to study the execution of non-hydrostatic ocean models on GPU and to create an experimental model for the study of small oceanic processes. We implemented a numerical, non-hydrostatic ocean model called “kinaco” [3] on GPU, following basic but essential methods. In addition, we introduced a mixed precision calculation to the multigrid preconditioned conjugate gradient (MGCG), which kinaco uses as a Poisson/Helmholtz solver.

2. METHODS

At the core of kinaco is the three-dimensional non-hydrostatic Navier–Stokes equation in the orthogonal curvilinear coordinate system, which simulates ocean dynamics. After making certain assumptions, this equation reduces to the 3D Poisson equation and the 2D Helmholtz equation for the pressure field and the free-surface elevation, respectively. Kinaco also simulates a number of values such as potential temperature, salinity, and other passive properties of sea water using advection-diffusion equations. All the equations are discretized on structured grids, and each grid systematically accesses the adjacent grids. The systematic character of the discretization and the introduction of MGCG give kinaco high performance and linear scalability in executions on the CPU with a large number of grids.

Kinaco was originally written in Fortran 90; therefore, the GPU version was implemented using CUDA Fortran. We followed

basic but essential methods, such as exploitation of large-scale parallelism in threads and instructions, coalesced access to the global memory, and minimization of memory transfer between the CPU host and GPU device. We did not change the basic structure of the programs in kinaco, except for a switch of the array of structures (AoS) to usual arrays, and the elimination of recursive description in the multigrid kernel. The switch of the AoS to usual arrays enabled coalesced access to the global memory, and the elimination of recursive description removed the overhead for launching the multigrid kernel.

We introduced a mixed precision calculation to the multigrid preconditioning, which deteriorates in performance due to the small number of GPU threads in the multigrid method. All arrays for multigrid preconditioning such as smoother matrices, residuals, and temporal arrays were set as four byte single precision floating point numbers. The precision of the conjugate gradient method was kept at double precision.

The GPU implementation was run and evaluated on a workstation with an Intel Core i7 3930K and NVIDIA K20C. We also evaluated the basic performance on a single node of a K computer incorporating a Fujitsu SPARC64 VIIIfx.

3. RESULTS

In comparison with the execution on a Fujitsu SPARC64 VIIIfx, the GPU-implemented kinaco, the Poisson/Helmholtz solver, and the other calculations including the diffusion-advection equations, respectively, ran 4.3, 2.8, and 5.1 times faster on the NVIDIA K20C. The application of mixed precision to the preconditioning contributed toward a 16% acceleration of the Poisson/Helmholtz solver, and most of the preconditioning kernels were also accelerated. For example, one of the matrix multiplication kernels in the preconditioning, namely apply3d3, ran 2.1 times faster with the application of mixed precision on the NVIDIA K20C, compared with that without. This acceleration is consistent with the decrease of global memory transfer due to the switch of precision. The outputs did not show any specific errors that would be critical in the study of oceanic processes.

4. DISCUSSION AND CONCLUSIONS

This study suggests that an ocean model is suitable for GPU implementation. There is potential for further improvement using techniques such as utilization of shared memory, storing temporal values to registers, kernel fusion, and loop unrolling. The performance of the Poisson/Helmholtz solver was improved less

than that of other calculations. However, the application of mixed precision to the preconditioning of the MGCG further improved the performance of the Poisson/Helmholtz solver. This study shows the application of mixed precision to be an effective method, suggesting further research to identify other applicable kernels and to verify them within both the computational science and geophysical disciplines.

5. REFERENCES

- [1] Milakov, M. 2013. Accelerating NEMO with OpenACC. NVIDIA GTC 2013.
- [2] van Werkhoven, B., Maassen, J., Kliphuis, M., Dijkstra, H. A., Brunnabend, S. E., van Meersbergen, M., Seinstra, F. J., Bal, H. E. 2013. A distributed computing approach to improve the performance of the Parallel Ocean Program (v2.1). *Geoscientific Model Development Discussions*. 6, 3, 4705-4744. DOI= <http://dx.doi.org/10.5194/gmdd-6-4705-2013>
- [3] Matsumura, Y., Hasumi, H. 2008. A non-hydrostatic ocean model with a scalable multigrid Poisson solver. *Ocean Model*. 24, 15-28. DOI= <http://dx.doi.org/10.1016/j.ocemod.2008.05.001>