Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2020-323-AC2, 2021

Geoscientific
Model Development
Discussions

Interactive comment on "The GPU version of LICOM3 under HIP framework and its large-scale application" *by* Pengfei Wang et al.

Pengfei Wang et al.

lhl@lasg.iap.ac.cn

Received and published: 26 February 2021

Numerical climate modeling is a key method for scientists and researchers to better understand our planet, and one of the most popular applications that greatly challenges the most state-of-the-art high performance computing (HPC) systems. In this work, LICOM3, a standard ocean model is selected and scaled onto the GPU-based heterogeneous supercomputing system. The authors have done lots of porting and optimizing work to put almost all of the time-consuming computation processes into the GPU side, and greatly reduce the communication overhead. As a result, both the dynamic core and the physics part are ported and parallelized on GPUs. A speedup of 42x is achieved when using 284 AMD GPUs VS 384 CPU cores. Excellent scalability is also achieved. A test of 1/20 degree LICOM3-HIP is reached using 6550 nodes and

26200 GPUs, 2.72 SYPD in time-to-solution.

As a computer scientist who also focuses on porting and tuning climate models onto different HPC platforms, dealing with a complete model with lots of code legacies using a new accelerator is obviously not an easy work. Sometimes rewriting and redesigning are necessary to obtain a satisfactory performance. In this work, the optimizing techniques provided are sound and solid, and can be used as a good guidance for corresponding work. AMD GPU and HIP, though not as popular as Nvidia GPU and CUDA for now, are still very promising GPU accelerators for current generation supercomputers. Moreover, it is likely that some of the forthcoming Exa-scale supercomputers, will also be adopting AMD GPUs. So this work is also a good trials ahead of time. More specifically to the strategies: only the most time-consuming parts (seven subroutines) are translated into HIP C, deeply re-coded, ported onto the GPUs, and fully optimized (such as the usage of temporary arrays to avoid data dependency, the change of data structure of original Fortran arrays, etc.). Halos that contain partial communications are handled by CPU part. Therefore, a hybrid computing model is performed, to further improve the overall performance. This is also a very popular strategy when dealing with numerical problems with inter-node or inter-process communications. Besides, The IO part is also considered and tuned by rewriting the data reading strategies and doing parallel scattering.

Overall, the paper is well-structured, with sufficient figures and tables to help better illustrate the ideas where necessary. But there are grammar errors and misleading descriptions here and there. So I suggest the authors ask help from native speakers for further proofreading.

Response: Thank you very much for your comments and suggestions. We have revised the English of the manuscript as possible and will find a professional English editor to further improve the final manuscript.

Here are some other suggestions, The authors mentioned the dynamic and physics parts, but lacks further explanations to what they are. I understand that most communications exist within the dynamic part, but could the authors be more specific in pointing out the optimizing strategies for the dynamic part and the physics part, respectively? Are there any differences?

Response: Thanks. To explicitly separate the dynamic core and the physical package is an excellent ideal for further optimization. But so far, the optimizing strategies are mostly at the program level, not treat the dynamic or physics parts separately. We only ported all seven core subroutines within the time integration loops to GPU, including both the dynamic and physics parts.

Unlike the atmospheric models, there are no many time-consuming physical processes in the ocean model, such as the radiative transportation, cloud, precipitation, and convection processes. Therefore, the two kinds of parts are usually not clearly separated in the ocean model, particular in the early stage of model development. This is also the case of LICOM. We have added the discussion of this issue in the revised version, Lines 440-445.

In the end, the authors claimed that the 1/20_ LICOM3-HIP version can not only reproduce the observations, but also produce much smaller scale activities, such as submesoscale eddies and frontal scales structures. Could the authors explain how they obtain the observation version?

Response: Thanks. This is kind of misleading. So far, the horizontal resolution of most global-scale observation is commonly no more than 25km from merged remote sensing products, which cannot resolve the submesoscale eddies in most places of the ocean. Some products, such as sea surface temperature, indeed have higher resolution at several kilometers. But these products usually have either short period or limited region, and not suitable for the global-scale, long-term climate research. Here, we only would like to say that much finer scale processes can be captured in this 1/20 model and didn't intend to compare with the fine scale observation.

We have revised the sentences and tied to avoid misunderstanding, in Lines 424-425.

Porting a complete model is not an easy work. In this work, approximately 12000 lines Fortran code were rewrote from fortran to C. Could the authors estimate the cost? For example, the number and time cost of persons in the whole project.

Response: Thanks. This is a good question. To port the Fortran code to C costs us about five months (2018.11-2019.3), and five Ph.D. students and five part-time staff participated in this programming work. Then, it took about ten months to port these C codes to GPU in CUDA (2019.4-2020.1) and further four months (2020.2-2020.5) to optimize them on the HIP framework, such as introducing IO parallel and doing the large-scale test. Therefore, it totally took nineteen months, and five Ph.D. students and five part-time staff to finish this kind of porting work. We have added the discussion of this issue in the revised version, Lines 406-407.

The following work is suggested to be cited and comment as well, to enrich the related work part. Optimizing high-resolution Community Earth System Model on a heterogeneous many-core supercomputing platform.

Response: Thanks for your suggestion and for providing a new reference. We have cited it in the revised paper in Lines 55-57.

Line 45, I suggest to update the TOP 500 list using the latest one (Nov. 2020);

Response: Thanks. We updated the list in the revised manuscript, Lines 45-47.

Line 49, I don't think the energy result is provided in the work of Xu et al. (2015). Please double check.

Response: Thanks. We have checked the work of Xu et al. (2015). They did provide the energy result in Sub-section 5.3.4 of their paper, shown in the following image (Figure R1).

Line 71, and the conclusions is in Section 6 –> and the conclusions are in Section 6

Line 74: which started to develop –> which has been developed Line 83: That makes the coupler is suitable to apply to high resolution modelling. –> It makes the coupler suitable to be applied to high resolution modeling. Line 85: improve –> improves Line 103: remove totally

Response: Thanks for your careful reading. We have corrected all five typos in the revised manuscript.

Part 2.3: add some citations or links with more detailed introductions to the supercomputer used in this work.

Response: Thanks. Because there is no publication about this supercomputer, we have added some information about this machine, including InfiniBand network speed and structure, the speed of the storage file system, etc. We have added this information in the revised manuscript, please see Section 2.3.

Line 140: place –> replacing Line 143: Some... the others... –> Some... some...

Response: Thanks for your careful reading. We have corrected these two typos in the revised manuscript.

Lots of professional words are used in this article, such as theses syntax and macros used in HIP or CUDA. Please use a different syntax (e.g. italic) for these professional word, to help better identify them. For example, I suspect that 'for tracer', 'baroclinic' and 'barotropic' may refer to professional processes or subroutines in LICOM3, and 'including', 'cuda_', 'hip_', may refer to professional designations of CUDA or HIP framework.

Response: Thanks for your suggestion. We have revised the manuscript to avoid misunderstanding. Now "barotr", "HipMemcpy" and etc., which inside "" are the function name in .cpp source file. The "cuda" and "hip" in Lines 154 is the prefix for the conversion of function call (or header files) from CUDA style to HIP style. For example "CudaMemcpy" need to be changed to "HipMemcpy", and "cuda_runtime.h" to

"hip_runtime.h".

Line 244, an –> a

Response: Thanks. Corrected.

Figure 5, I suspect the IO time is the result after the IO optimization (part 3.4) being applied. Is that right?

Response: Thanks. Yes, it is correct.

Line 263, times –> time

Response: Thanks. Corrected.

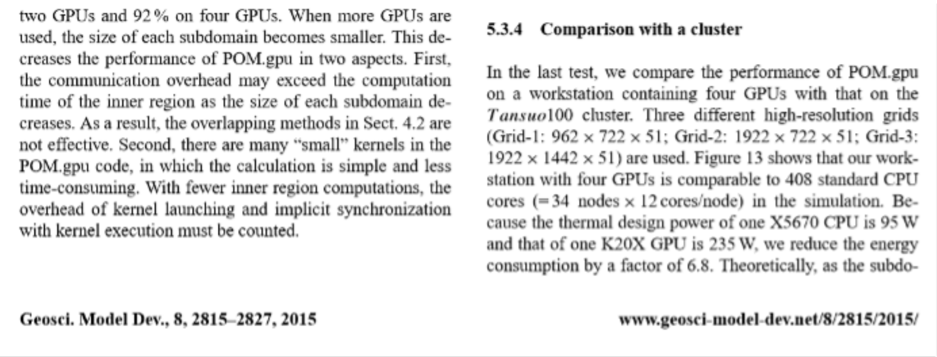
Please provide more details about the hardware configurations, e.g., the version of CPU and GPUs, the version of compilers, OS, etc.

Response: Thanks. We have added all the information in the revised manuscript following your suggestions in Section 2.3.

Please replace Flops/s with Flops.

Response: Thanks. Replaced.

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2020-323, 2020.



two GPUs and 92 % on four GPUs. When more GPUs are used, the size of each subdomain becomes smaller. This decreases the performance of POM.gpu in two aspects. First, the communication overhead may exceed the computation time of the inner region as the size of each subdomain decreases. As a result, the overlapping methods in Sect. 4.2 are not effective. Second, there are many "small" kernels in the POM.gpu code, in which the calculation is simple and less time-consuming. With fewer inner region computations, the overhead of kernel launching and implicit synchronization with kernel execution must be counted.

5.3.4 Comparison with a cluster

In the last test, we compare the performance of POM.gpu on a workstation containing four GPUs with that on the *Tansuo*100 cluster. Three different high-resolution grids (Grid-1: 962 × 722 × 51; Grid-2: 1922 × 722 × 51; Grid-3: 1922 × 1442 × 51) are used. Figure 13 shows that our workstation with four GPUs is comparable to 408 standard CPU cores (= 34 nodes × 12 cores/node) in the simulation. Because the thermal design power of one X5670 CPU is 95 W and that of one K20X GPU is 235 W, we reduce the energy consumption by a factor of 6.8. Theoretically, as the subdo-

Fig. 1. Image from Xu et al. (2015).