Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2019-257-AC3, 2020

Geoscientific Model Development Discussions

Interactive comment on "Parallel I/O in FMS and MOM5" *by* Rui Yang et al.

Rui Yang et al.

rui.yang@anu.edu.au

Received and published: 15 March 2020

Referee's Comments: The authors present a detailed study of implementing parallel I/O using NetCDF in the Modular Ocean Model version 5 via the Flexible Modelling System. Even though the implementation is quite specific to MOM5, the paper can serve as a useful experience for developers aiming to implement parallel I/O within other scientific software packages. Overall, I believe the paper is worth publishing, especially since I/O aspects are often neglected. There are still some points for improvement, though.

Specific comments:

Referee's Comments:

- Lines 36-38: Where is the number of 350 MB/s for disk throughput coming from? The HDDs I know about typically max out at roughly 200 MB/s. While I understand the point

you are trying to make with these sentences, I believe some more details would make them easier to follow.

Response:

The 350 MB/s performance was based on measurements of direct disk writes (using 'dd') for idealized output on the machine (Raijin), and was the performance typically reported in most technical specifications of the machine. Although this machine has since been decommissioned, 350 MB/s is the cited performance of the "gdata1" filesystem; see the following presentation by Daniel Rodwell from 2016, slide 27 (https://www.eofs.eu/_media/events/lad16/05_petascale_data_migration_rodwell.pdf).

We also note that consumer SATA SSD speeds of 500 MB/s are not uncommon, and the presentation above cites Lustre OST write speeds as high as 800 MB/s. So our estimate of 350 GB/s seems to be reasonable for the purpose of discussion at this early stage of the paper. Given the large variation in write speed performance, we do not have a good reference but are welcome to suggestions from the reviewer.

Referee's Comments:

How long does a one-year simulation typically take? Is writing out one terabyte of data even relevant in this case?

Response:

We believe that the sentences preceding the discussion of terabyte-per-year output justify this output rate. A typical 0.1° grid (3600 x 2700) with 75 levels at double precision will require approx. 5.8 GB per step. At a 5-day output rate, this will require over 400 GB, and the output could have many such fields.

There is no simple way to characterize a typical model output rate, but we believe that the information above justifies that even a minimal high-resolution experiment will produce output on the order of terabytes per year.

As for whether a year represents a typical climate simulation time, we felt that this needed no citation.

A typical runtime of a high resolution model would be on the order of 10 hours per year. But the compute runtime or the ratio of compute to I/O time does not change the fact that terabytes of data must be written, and the preceding sentences establish that it is a reasonable workload for a high resolution ocean model. It must be done, and it would require hours of time to complete if it were done serially.

Our purpose was to demonstrate that serial I/O of a high resolution model is a prohibitive task, and we feel that the leading statements justify this statement. But if the reviewer disagrees with any of the statements above, we are happy to address them.

Referee's Comments:

- Lines 87-96: Please elaborate why you have selected NetCDF for your parallelization efforts. There are also other approaches such as SIONlib or ADIOS. While NetCDF probably makes the most sense for geoscientific applications, this should at least be discussed briefly.

Response:

In a later revision of the paper as a response to the first referee, we explain that I/O domain of FMS provides most of the functionality of these libraries, and therefore opted to directly implement parallel I/O based on the existing I/O domain structure. The following discussions have been added in Section 2 of the revised manuscript:

"Because FMS provides access to distributed datasets as well as a mechanism for collecting the data into larger I/O domains for writing to disk, we concluded that FMS already contained much of the functionality provided by existing parallel I/O libraries, and that it would be more efficient to generalize the I/O domain for both writing to files and passing data to a general-purpose IO libraries such as netCDF. In this sense, there is no need to set up the dedicated I/O server with extra PEs as other popular parallel

I/O solution like XIOS does."

Referee's Comments:

- Lines 191-195: Have you considered the alignment of chunks? We have shown in "A Best Practice Analysis of HDF5 and NetCDF-4 Using Lustre (Bartz, Chasapis, Kuhn, Nerge, Ludwig)" that chunk alignment can have very significant impact on parallel I/O performance. Sadly, NetCDF did not (and apparently still does not) expose this functionality while HDF5 does. It is therefore necessary to patch NetCDF to enable HDF5's chunk alignment. Missing alignment could be the cause of contention you describe when increasing the number of I/O PEs per I/O domain.

- Lines 295-299: See previous comment, this could also be caused by missing alignment.

Response:

In this paper we focus on the configurable parameters associated with MOM5 I/O domain layout, the netCDF library (based on standard HDF5 installation), MPI-IO, and the Lustre file system. The impact of chunk alignment configurable by HDF5 is an interesting idea worthy of further exploration, and it may help to explain some of the performance differences between PnetCDF and HDF5, but we feel that it is perhaps beyond the scope of this paper, i.e. it is not tuneable via netCDF library.

Referee's Comments:

- Lines 451-453: The serial I/O versions with 720 PEs ran for 6 hours while the ones with 1440 PEs were killed after 5 hours. Did the 720 PE version run on a different partition? If so, is it still possible to compare the two?

Response:

Both 720 PE and 1440 PE jobs run on the same partition, but the latter has a shorter time limit, i.e. 5 hours, set by the PBS queue system. At this stage, it is not possible

to compare the two as the machine has been decommissioned. We can only present it as an incompletable task on our platform, which we believe is sufficient for the more detailed analysis of the parallel I/O performance. However, as per request of the first referee, we added a new table (Table 6 in the revised manuscript) to compare simulations with typical I/O loads. In that table, both serial I/O and parallel I/O are compared between 720 PEs and 1440 PEs but with much less I/O loads than those in Table 5.

Referee's Comments:

- Lines 508-512: Why did you develop your own I/O profiling tool? There are existing options such as Score-P or Darshan. Please state why the existing tools did not meet your requirements.

Response:

We would like to analyse costs of each site in major I/O call paths by collecting the elapsed time and sizes for all MPI ranks at multiple I/O layers such as NetCDF, MPI-IO and POSIX I/O calls. The existing I/O profilers, however, cannot fully approach this goal. Score-P is good at profiling user code and MPI-IO functions, but it is hard to measure the time spent within the netCDF library and POSIX calls. Also, it cannot measure the elapsed time and size per I/O operation for each individual input and output file. Darshan, on the other hand, is good at profiling the time and size of I/O operations of different files. However, it cannot provide the rank distribution of time which is necessary to analyse the load balance issue.

By recognizing the above deficiencies of existing I/O profilers, we decided to develop our own profile tool which can address above issues with negligible overheads. The tool can provide all details we need to evaluate the cost of each I/O layer, rank distribution of time per file, access size per I/O function or operation and so on.

Referee's Comments:

- Line 526: I gave the GitHub repository a quick look but could only find the source

code. According to GMD's code and data policy, the data must also be provided. You have also not mentioned in the paper which commit you were using to perform the model runs.

Response:

These issues were also raised by the executive editor. We have updated more details about the code and data availability as below:

The source code of parallel I/O enabled FMS is available from doi:org/10.5281/zenodo.3700099. The MOM5 code used in the work is available at https://github.com/mom-ocean/MOM5.git. The core dataset is available as doi:10.1007/s00382-008-0441-3. Build script, configure files and job scripts are available from dio:org/10.5281/zenodo.3710732.

Referee's Comments:

Technical corrections: - Line 28: The acronym OS has been introduced before in line 23 and does not need to be repeated here.

Response: Removed acronym OS.

Referee's Comments:

- Lines 62-70: Since you talk about "single file I/O" in the paragraph before, it might be worth mentioning explicitly that one file is created per I/O domain in this case.

Response: Explicitly cite 4 write patterns of Table 1 in the context.

Referee's Comments:

- Line 73: "A typical 0.25âŮę global simulations ..." - It should be "simulation".

Response: Fixed.

Referee's Comments:

- Line 183: "... in Table 3, ..." - This should be "Table 2".

Response: Fixed.

Referee's Comments:

- Line 227: "... of the I/O parameters in Table 3." - Should be "Table 2".

Response: Fixed.

Referee's Comments:

- Line 238: "... grids are disturbed over ..." - This should probably be "distributed".

Response: Fixed.

Referee's Comments:

- Line 375: "... in the charts below for each library." - This should rather reference the figures directly since they are placed in the appendix.

Response: Reference the figures directly.

Referee's Comments:

- Line 429: "... in Figure 14." - Figure 14 seems to be rather blurry while the others are fine. Please provide a high-resolution version if possible.

Response: Figure 14 is reproduced with the higher resolution.

Referee's Comments:

- Lines 581-622: Are the reported values averages? If so, you should mention this somewhere and also give deviations. Figure 14 already includes them but the others do not.

Response: Clarified in figure caption that the reported values are maximum among all PEs.

Referee's Comments:

- Lines 625-639: Bright orange is hard to read on white, so it might make sense to change the color for the profiling graphs.

Response: The light color has been replaced by the deeper one.

- Line 665: "Number of Output File" - This should be "Files".

Response: Fixed.

Referee's Comments:

- Lines 680-685: To better assess the scaling behavior, please also mention the number of nodes in addition to the number of PEs.

Response: Added node counts in Table 5.

Please also note the supplement to this comment:
https://www.geosci-model-dev-discuss.net/gmd-2019-257/gmd-2019-257-AC3-supplement.pdf

---

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2019-257, 2019.