Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2020-91-RC3, 2020

Geoscientific Model Development Discussions

Interactive comment on "A new distributed algorithm for routing network generation in model coupling and its evaluation based on C-Coupler2" *by* Hao Yu et al.

VIJAY MAHADEVAN (Referee)

mahadevan@anl.gov

Received and published: 13 May 2020

1 Summary

The manuscript under review presents an improved and scalable routing network generation algorithm for model coupling between component in climate solvers implemented within the C-Coupler2 infrastructure.

The authors present motivations on why the existing routing network generation algorithms do not scale well, and showcase the performance degradation in terms of both time and memory on increasing core counts due to a $O(N^2/K)$ complexity, where N

is grid size and K is number of processes. The DaRong algorithm introduces changes to the workflow and corresponding datastructure modifications to maintain the computational cost bounded by $O(Nlog(N)/K)$.

Results demonstrating the superior parallel efficiency of the modified algorithm are also presented in comparison to the global algorithm that involves collective gather/broadcast operations. The figures provided in the manuscript are helpful, and the tables explain the construction of the datastructures needed for the algorithmic workflow. However, the language is confusing in certain sections, and should be rephrased to better improve the overall thesis presented.

In summary, the authors prove that the existing algorithms can be improved through the introduction of an intermediate distribution to eliminate collectives to make this step scalable. However, I fail to see enough conclusive evidence that improvement of just this initialization phase of the solver would lead to a significant impact on the total time to compute the overall coupled climate solution. Additionally, the proposed modifications to the global routing table generation scheme is incremental in nature, and does not aim to minimize communication times between source and destination processes. I do not recommend the publication of the submitted manuscript in the journal of Geoscientific Model Development, as significant new additions are needed to provide better motivations, and results to highlight the overall impact due to the modification in the routing network generation algorithms. Detailed comments are provided below.

2 General Comments

The algorithmic modifications proposed in the current manuscript is an incremental update to an existing algorithm, and does not provide a significant enough impact on the overall runtime of the climate solver. It does provide a lower bound on the overall parallel setup cost of the routing network generation, which is used for efficient data-

transfer at runtime during temporal integration of the coupled components. However, since the algorithm does not aim to provide a better partitioning strategy, or make use of architecture layout to minimize communication latency (using say task mapping algorithms), the resulting communication graph between processes still remain the same as the one generated from global routing network algorithm.

More specific comments detailing areas that should be addressed are listed here. I am happy to engage in a conversation if the authors require more clarifications on my comments.

1. First, the authors claim that "existing couplers employ an inefficient and unscalable global implementation for routing network generation that relies on collective communications. That's a main reason why the initialization cost of a coupler increases rapidly when using more processor cores.".

   Can you provide some actual timings from the fully coupled climate solver runs to put the actual setup costs in perspective ? The scalability shown in Fig. (3) still indicate about 3.5s of compute time (since speedup for global routing network case is $\approx$1) for the 16M grid case on 1600 processes. If this accounts for say over 5% of the actual runtime of the solver, or a non-trivial percentage of total time to simulate a year (or days for high-res) of climate interactions, then 20x improvements in the setup cost could be quite impactful. However, such one time costs get amortized with physics setup costs for high-res runs, in addition to long-term temporal integration of the actual coupled simulation. Hence, I think the manuscript lacks a strong motivation, and provides only an incremental update to avoid the collective algorithms in the coupled simulation invoked during the initial setup phase.

2. Secondly, the global ID based partitioning strategy used in the distributed sort with DaRong to determine the communication pattern is not an innovative concept. There have been several algorithmic ideas based on graph partitioning

   strategies used in the parallel Sparse Matrix-Vector (SpMV) linear algebra context [1].

   In a simplified sense, without a constraint on the message volume, data locality or latency of communication (such strategies may require repartitioning and/or task mapping), the globally unique ID space can be used in a round-robin type partitioning scheme. For instance, if the source component data are distributed on M processes, and destination on N processes, an implicit decomposition can be determined a-priori based on the global ID numbering that leads to MxN data redistribution. Such an implicit ID decomposition establishes a direct point-to-point communication pattern after which the CLGMT table can be created on both source and destination processes for further send/receive of DoF data at runtime. There may be a need for multiple rounds of rendezvous communication to establish message size for buffer allocation etc, but such an algorithm can eliminate collectives like broadcast and allreduce operations as necessary for better scalability.

3 Specific Comments

1. *Line 29-30: "Although most existing couplers achieve scalable data transfer and data interpolation, i.e., the data transfer and data interpolation generally can be faster when using more processor cores, there is almost no evidence of scalable initialization of a coupler."*

   Total cost of a coupled solver includes both the setup/initialization cost and the runtime remap operator application and data-transfer at every time step per coupled component pair/field. Hence, cost of initialization often gets amortized in a climate simulation run. As mentioned previously, please cite or provide data to substantiate such strong claims, preferably with real results using MCT and

C-Coupler2 runs.

2. *Paragraph starting at line 79* is confusing. Please rephrase the sentence better to clearly describe the particular step and its time complexity that leads to the inefficient and non-scalable implementation of routing network generation.

3. *Line 104: "Specifically, we employ a regular intermediate distribution that evenly distributes the CLGMT entries among processes based on the ascending order of the global cell index. Such an intermediate distribution is not only simple, but also enables to easily achieve the rearrangement to the intermediate distribution via a sorting procedure similar to distributed sort. "*

   As noted previously, the GSMap and Router infrastructure in MCT already has such options to redistribute data based on Global ID numbering. This is inherently what has been described here as the intermediate distribution of the CLGMT. The primary difference seems to be that GSMap is a O(P) datastructure that grows with core counts, and is accumulated on all process through a gather on root and a subsequent broadcast. The authors of the current paper are trying to avoid this one-time collective operation, which could be an over optimization considering the total runtime of the climate solver.

4. *The paragraph starting at Line 171* can be rewritten in the context of the distributed sort workflow shown in Fig. (2).

5. It will be helpful to explicitly mention the time and memory complexity for each stage in a table format, for both the global and the DaRong algorithm so that the reader can immediately get a sense of the actual improvement.

6. The weak scalability results shown in Table. (7) are not uniform since the grid sizes are doubled, but the core counts do not, going from 250 to 450 and 900 to 1600. Please rerun these calculations with P=[200,400,800,1600] instead.

4 Technical Corrections

1. Section (3.2) title: "Rearranging CLGMT entries intra a component model". Please rephrase. Do you mean to say "between component models" ? Same comment applies to Section (3.5) title as well.

2. Line 175, "SPT" should be "SRT" ?

3. Consistently use "Fig." vs "Figure" to reference figures

References

Hendrickson, Bruce, Robert Leland, and Steve Plimpton. "An efficient parallel algorithm for matrix-vector multiplication." International Journal of High Speed Computing 7, no. 01 (1995): 73-88.

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2020-91, 2020.