Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2018-280-AC3, 2019

Geoscientific Model Development Discussions

Open Access

Interactive comment on "Improving climate model coupling through a complete mesh representation: a case study with E3SM (v1) and MOAB (v5.x)" *by* Vijay S. Mahadevan et al.

Vijay S. Mahadevan et al.

mahadevan@anl.gov

Received and published: 24 January 2019

Dear Reviewer,

We sincerely appreciate the detailed comments and suggestions for modifications to make the manuscript clearer. We have replied specifically to some of the question in the review below, and we are in the process of including the suggested modifications in the final manuscript.

1. Reviewer: The outstanding questions that are not answered in the paper are

Printer-friendly version

Discussion paper

(a) can the weights generated online be counted on to produce error free interpolation (conservation, monotonicity, etc) without first being reviewed and validated offline?

(b) is the weights generation capability robust and reliable enough to run on different platforms and expect the same results to at least roundoff?

(c) is it faster to generate weights online vs reading them in?

(d) Is there some benefit to generating the weights online and then being able to reuse them as compared to regenerating them each time the model is run with regard to performance or reproducibility?

It would be helpful if the paper addressed these issues if possible. These issues are partly raised in a few places in the paper, at least Page 6, Lines 25-27 and Page 18, Lines 9-10. Some addiitonal discussion/results might be interesting.

Author: Thank you for the detailed comments. Some comments below to address concerns. We will also add specific information on having a reproducible workflow using the online remapping implementation.

(a) The online remapping weights use the exact same input grids, discretization specifications and even most of the same routines as the offline method with TempestRemap. These are also exposed directly through a MOAB-TempestRemap tool (mbtempest) that can be run offline, and in parallel, for verification and validation along with ability to write out the weights to file. MOAB handles the mesh decomposition, the parallel intersection computation, DoF management, and offloads the actual remapping weight computation to TempestRemap, which has been verified and validated independently.

(b) Yes the online weight generation workflow is robust and the MOABTR workflow works consistently in serial (OSX, Linux) and in parallel on clusters and large-scale machines (Blues, Cori, Theta).

(c) We have not yet explicitly performed production-case comparison tests against serial I/O read of the weights file along with broadcast to tasks in pes from a single process vs the fully online remapping weight computation with MBTR.

(d) At large scale, we expect the online remapper to be much faster, since I/O is expected to be the slowest component in next generation architectures. Hence, it may be advantageous to run the onine remapper all the time for production runs, after sufficient verification and validation. We can write out the generated weights for provenance and reproducibility.

2. Reviewer: Page 8, line 1, what does mesh aware entail? You discuss the potential all-to-all nature of weights generation in the prior paragraph. What information does MOAB carry around that help this problem and how much memory does it require? Does each task have access to the global grid information without requiring communication? Or is there just neighbor connectivity stored?

   Author: Mesh aware indicates that the data-structure has the knowledge to traverse the underlying discrete grid used in the component model. The all-to-all communication may happen if the source and target grids on the coupler PEs have no "geometrically coincident" elements. This means that in order to compute the intersection mesh on a particular task, we need to bring the grid elements and vertices that fully cover the target regions. We explain this task as the source coverage mesh computation in Algorithm 1.

   MOAB does not explicitly store datastructures that have a O(P) dependency, where P is the number of processes. Only neighbor connectivity information and "ownership" information for shared entities across process domains is stored.

3. Reviewer: If the grid description is not global on all tasks, how much is communication reduced vs having only local information? The MCT gsmap has global information on each task related to ID, and pe. I assume the MOAB mesh has

the same plus coordinate information? The MCT gsmap is generally compressed significantly because the information can be defined via a single start and end ID for certain kinds of decompositions. Since the MOAB mesh carries more info, I assume that compression is not possible and that the mesh consists of "n" fields of data for each gridpoint/corner/edge/etc? Is that a lot of data? Does the memory scale at all at higher resolutions and higher pe counts?

Author: MOAB is a fully distributed datastructure. MOAB does not have any global storage resembling or similar to the MCT gsmap. We use a transient datastructure with bounding boxes (that encompasses the local elements/vertices) to determine the communication pattern between arbitrary decompositions during the setup phase. All communications from thereon are point-to-point. The key differentiating factor, as you pointed out is that MCT has no notion of the global mesh/vertex distributions and hence optimizations based on topology are unavailable. However, since all meshes are treated as truly unstructured, MOAB does store the 'local' vertex coordinates and the connectivity information without any compression, although in a contiguous array-based datastruture.

The fields are stored contiguously per element and the user manages the DoF layout definition on each local element. So for a SE discretization, DoFs may have canonical numbering with p=4, which results in 16 DoFs. For FV, there may only be 1 data per element. This memory requirement is directly proportional to the solution data vector in each component. MOAB being a general mesh library, does allow users to define data on edges/faces etc but we do not use this functionality in our current coupler implementation. At larger resolutions, each task gets a smaller piece of the mesh along with a smaller footprint of the data as well since MOAB is a fully parallel mesh.

4. Reviewer: How the new capabilities are implemented in E3SM is somewhat unclear. In Figure 1, it looks like there is no longer a coupler. Where are the non-coupling non-mapping coupler operations (merging, atm/ocn flux, diagnos-

tics, etc) being computed? In text, it sounds like the coupler component still exists but that the underlying MCT datatypes were swapped for MOAB datatypes, an additional set of calls were added in the component coupling layer to more fully describe the meshes, the online weights generation was added, and the online sparse matrix multiply was converted from MCT calls to MOAB calls. But then at page 17, line 12-15, it sounds like the coupling is between pairs of components excluding a coupler. It would be good if this were clarified.

Author: The hub coupler still exists. We are currently duplicating the MCT calls alongside the MOAB based coupler in order to fully verify and validate both the accuracy and performance at runtime. After full validation, the MCT coupler will be completely removed from E3SM. The MOAB coupler allows the possibility for ATM to directly compute the remapping weights to project field data to OCN since the intersection will then be carried out through migration of OCN mesh to ATM pes. Hence this pair-wise coupling leads to a more distributed coupling strategy in the future. However, we do envision that there will still be a thin layer of a global coupler, even in the distributed case, to drive the subcycling, to compute merging with weighted combinations of fluxes, for validation and other diagnostics data outputs. We understand that Fig. (1) is somewhat misleading in this context and intend to make modifications to make it clearer.

We welcome any additional comments on this topic although we realize that the discussions are closed at this point.

Sincerely,

Vijay Mahadevan, Robert Jacob, Iulian Grindeanu, Jason Sarich

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2018-280, 2018.