Geosci. Model Dev. Discuss.,
doi:10.5194/gmd-2015-267-RC2, 2016

Geoscientific Model Development Discussions

Interactive comment on "YAC 1.2.0: An extendable coupling software for Earth system modelling" *by* M. Hanke et al.

Anonymous Referee #2

Received and published: 19 February 2016

General Overview:

This paper provides an overview of the YAC1.2.0 coupler focusing mainly on the interpolation weights generation. It is reasonably well organized and written and with revisions, will contribute to the literature on ESM coupling.

I am a bit torn about the organization of this paper. The paper focuses on the YAC weights generation, but the title suggests the paper is a description of the YAC coupler. I think this paper should be revised to clarify the focus. I recommend this paper focus on interpolation in YAC, by providing more details about the weight generation algorithms, show more scaling data for other weights generation options, describe the weights generation API in YAC, describe the remapping algorithm in the code and add some scaling data about the remapping cost in YAC. I believe this paper could meet those

goals relatively easily, although some of the current text may need to be removed and refactored. I would also recommend a change in the title in this case.

An alternative would be to turn this into a YAC overview, but in my opinion, that would require significant rework. The paper would need to add more information about hooking up models to YAC including some further overview of the strategy, it would have to discuss how data is passed between models and the strategy employed in communication, lags, coupling frequency management, sequencing, concurrency, and other aspects of the system. It would also need to add further historical description of other ESM couplers in the introduction in order to change the focus.

In many ways, I think both papers above are worth pursuing. This software effort needs to be published. There are several interesting and unique aspects of the weights generation that can stand alone in a paper. A YAC overview paper is also appropriate to describe overall design to the community, especially if YAC is available for use within the community. Unfortunately, I think the current paper is a bit of both and is not properly focused or complete in either case.

Specific Comments:

Overall, I believe there is a shortfall in the number of cited references. If this is a focus on weights generation, there are a number of references that should probably be sited in the introduction and text. If this is a focus on new ESM coupling software, again there needs to be additional references and some additional historical perspective added. Even without changes, references need to be added (for instance, Oasis3 on page 2, line 8 and MCT on page 2, line 12 but there are others). The authors need to make an effort to properly reference work sited in the paper.

Please clarify the following. Are the weights generated on the source or target side, on the union of all tasks or is it up to the user? Is remapping done on the source or target side or as part of communicating data between models? Is the communication of data between models separate from remapping or part of remapping? There are hints of

the answers in the text, but it isn't clear.

The authors are extremely familiar with Oasis4, but it's unlikely the readers will be. It is fine to compare to Oasis4 and point readers to an Oasis4 reference, but it is also important to make sure the description does not assume the readers are inherently familiar with Oasis4 and that it does not require a study of Oasis4 as a prerequisite.

How does YAC ensure that the weights generated on the fly are of high quality? The text notes problems at poles in various algorithms, issues with different types of grid-cells and different edge options, the requirement that one grid have either convex edges or be rectangular. Are those things checked on the fly? Are the weights somehow checked for conservative or gradient properties independently after they are generated? Are the properties at the pole checked? Is this a concern?

Page 1, Line 20: "not critical" is not a proper description. ESMs have been carrying out computations on moderate to high resolution grids for several years and the (pre) computation of interpolation weights in terms of both performance and quality has been a critical issue. This issue has taken up not insignificant resources in several projects including OASIS, ESMF, and SCRIP.

Section 3.2 describes the communication implementation and is far more important than much of the other material before it. I would like the description here to be expanded a bit, especially as related to lines 10-22. You need to describe "the non-blocking buffered send from Oasis4" for those that are not familiar. Please explain how the callback and data pointer work in a bit more detail. This is not entirely clear. Remember that your audience didn't work on either the Oasis4 or YAC implementation. Also, maybe there needs to be some further clarification on what aspect is being described. There is communication associated with weights generation, interpolation, and coupling data. Is this communication approach used for all of these?

Section 3.3. The requirement on the user to decompose the grid in a way that also includes a halo region and the rank owner of each halo gridpoint seems to be rather

inflexible. That datatype and information will almost certainly have to be computed specifically to support the YAC weights generation interface and is unlikely to be a natural part of any model decomposition. The halos are probably only needed in a subset of interpolation methods (like bilinear) and maybe YAC should be computing that connectivity, not the model. In addition, in a decomposition like round robin where each process has a random(ish) set of points, the halo description is going to require that "n" halo points be specified for each gridcell, increasing the memory and complexity. YAC would be much more usable if the connectivity were computed within the coupling layer when needed.

Section 3.4. The interpolation stack feature is well thought out and something that other weights generation methods are not able to do easily yet but is needed. Well done.

Section 3.5. Do you have a bilinear interpolation option? This option is heavily used in ESMs.

Section 3.7. Does the current calculation of intersections guarantee conservation. The overlapping areas have to be computed in a way that the partial areas add up to the areas of each grid cell in total. I think page 14, line 3 suggests the areas are handled properly, but maybe a sentence stating this would clarify.

Section 4. I find the description of the user interface a bit out of place. The article is really focused on the interpolation weights generation. Nothing has been presented about how interpolation is carried out nor how models are coupled. This section, describing how to setup YAC via XML does not seem to fit into the paper.

Section 5. The total time and scaling for weights generation is very good. I think it's also important to show similar timing information for patch (1st, 2nd, and 3rd) and nearest neighbor calculations as well as bilinear if it's available. These are very different algorithms and I think providing scaling information for all three is important for this paper. In our experience, the 1st order conservative weights generation is sometimes

the easiest and fastest to carry out. (This is also noted in Section 6, so data to back it up would be very useful)

Figure 4 title seems incorrect. According to the text, it is the time to generate 1st order conservative mapping weights, not to carry out remapping.

Section 6. Line 25. The only thing that has been shown in the text is the cost of the weights generation for 1st order conservative. It is a stretch to say "YAC scales reasaonbly well". Much of the YAC performance is not shown including remapping performance and cost to communicate data back and forth between models, the ability to support concurrency and other issues. Section 6. Line 30. It would be nice to document the actual time needed to write and read mapping files as this is already available in YAC to have a quantitative comparison against the cost of generating the mapping weights. It would not surprise me if the read operation was more expensive than online weights generation for the test case used, and the actual numbers would add to the paper.

Section 8. The web info is pretty useful. Just FYI, I think it is missing a "getting started" type of documentation for new users to know what calls are needed and how to organize them in their models.

Technical Corrections:

The grammar is pretty good in this paper, but could be improved. I would recommend a review by a native English speaker if possible. A few problems are noted below, but the list below is not comprehensive.

Please make sure to have references for both L'Huilier's Theorem and Girard's Theorem in the paper.

Please add a reference to the YAC Doxygen web site the first time it is mentioned in the paper. I know it's list in section 8, but a reference would also be good.

Page 1, Line 20-22, clean up sentence

Page 2, Line 3: ESMF is not NCAR. It's now NOAA. Get rid of the phrase starting with "mainly written..."

Page 2, Line 13: fix "still requires to perform", not correct grammar

Page 3, line 30: change to "with a few parts written in C"

Page 4, line 7: change "growing" to "grow"

Page 5, line 15: remove "it" from "than it is the case with Fortran"

Page 5, line 20-23: sentence is difficult to read

Page 6, line 8: remove "a" from "Even though a proper documentation"

Page 6, line 13: rebuild -> rebuilt

Page 6, line 14: need a comma after "software"

Page 6, line 18: anymore -> thereafter

Page 6, line 28-29: "We feel very comfortable with" should start the sentence, although that sentence is a bit awkward in either case.

Page 7, line 10: remove "it"

Page 11, line 10: need some commas.

Page 11, Section 3.5, paragraph 1, please rewrite to make a little easier to understand.

Page 11, Section 3.5, paragraph 2. I think the main point is to determine which process will carry out the interpolation and how to collect information for polygons that overlap multiple processes on the other grid. Please rewrite to make it a little easier to understand. Also, could you just add 1 sentence about what FITT does?

Page 12, line 6: one -> on

Page 15, line 9: what is "one couple"?

Page 18, line 1: I would remove this sentence. It's dangerous to extrapolate in these cases and the data speaks for itself.

Page 19, line 19: change to "...,we designed and developed a coupler..."

Interactive comment on Geosci. Model Dev. Discuss., doi:10.5194/gmd-2015-267, 2016.