Geoscientific Model Development Discussions

Geosci. Model Dev. Discuss., referee comment RC2
https://doi.org/10.5194/gmd-2021-335-RC2, 2021

Comment on gmd-2021-335

Anonymous Referee #2

Referee comment on "Large-eddy simulations with ClimateMachine v0.2.0: a new open-source code for atmospheric simulations on GPUs and CPUs" by Akshay Sridhar et al., Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2021-335-RC2, 2021

The manuscript covers an interim (as suggested by the version number) report on the development of the atmospheric component of the Earth System modelling suite developed by the CliMA team. The paper reads well. If the goal of the manuscript is to guide potential users (and developers) of the system through the LES functionality of ClimateMachine using a set of examples depicting capabilities of the framework, including achievable scaling, this goal is achieved. Moreover, such a goal is certainly in line with the journal scope.

Such a goal is unfortunately not fully reflected in the "content balance" in the paper. The first ten pages of the work read as a general description of a GCM dynamical core design. While it is of course relevant to subsequent material, numerous introduced aspects of the model are not supported with the examples covered in the paper, e.g., thermodynamics of the ice phase, or "physics" (e.g., precipitation) source terms. Presented examples are discussed more briefly than the not-exemplified model formulation aspects leading to a bit puzzling set of material. Reading through the paper, a question arises: will future papers on ClimateMachine repeat the material from the first ten pages or will they refer to "Large-eddy simulations with ..." for a description of the GCM dynamical core, thermodynamic state description, etc - both options seem undesired.

Why not set up a special issue in GMD (or a GCM/ACP/WCD/ESD/... inter-journal SI) devoted to CliMA developments, and extract some of the "commons" from the present work into shorter papers? For instance: (i) the DG numerics with examples supporting their choice; (ii) the thermodynamic variables and examples supporting their choice; (iii) the engineering aspects including the choice of Julia, the parallelisation strategies and benchmark results supporting the choices? Just an idea. Even if the Authors and Editor deem the current content balance OK, perhaps it is worth considering such an option for future publications?

General remarks:

First of all, ClimateMachine is a project which, in an exemplary fashion, consistently and boldly employs the very best practices in software engineering and reproducible research - a still rare example in geosciences. The modularity and documentation availability is simply stunning. It is unfortunate that almost none of these "engineering" aspects have the proper coverage in the manuscript. It is merely mentioned that the Julia language was chosen, that the code is compilable for both CPU and GPU execution, the Data Availability section reveals some of the reproducibility aspects. Unless this is meant to be covered elsewhere (which would be even better, see my comment above on a "special issue" idea), devoting a full section on software architecture, design and technological stack choices will likely be highly appreciated by the readers and encourage them to invest the time to get familiar with the new tool. The technologies used are novel and not widely used in the community. Elaborating on these aspects will also support numerous statements and conclusions of the paper.

The choice of prognostic variables is described as an important aspect and notable feature of the model, yet little references are made to discussions in literature of this choice, for example, Satoh 2003 (doi:10.1175/1520-0493(2003)131<1033:CSFACN>2.0.CO;2), Duarte et al. 2014 (doi:10.1175/MWR-D-13-00368.1), Tomita and Satoh (doi:10.1016/j.Fuiddyn.2004.03.003). Providing an outline of alternative choices, e.g. referring to Table 3 in Ulrich et al. 2017 (doi:10.5194/gmd-10-4477-2017) for GCM, and referring to other LES system descriptions would also help to present the choice in a wider context.

Low communication overhead of the embraced Discontinuous-Galerkin numerics is presented as enabling scaling on manycore processors. Usage of MPI for communication is presented as obvious throughout the paper. Yet, multiple cores of modern CPUs (and multiple threads within a block on GPUs) have shared access to memory, making data transfers not needed and reducing communication to barrier housekeeping within a given thread pool (see e.g., Hoefler et al. 2014, doi:10.1007/s00607-013-0324-2). The statement on low communication overhead and manycore processors is thus at least too general. For instance, as indicated in Appendix C, the employed hardware had 28 cores per node - 28 cores sharing access to the same memory. Overall, it would be of great value to elaborate (or at least acknowledge) on this very aspect of parallelization. At present, it seems to be only commented in the paper with the p27/l543 statement that "Scaling across multiple threads is not assessed in the present work".

The DG numerics are presented as somewhat flawless and trouble-free. Yet, generally, the presented examples do not depict cases of transport of quantities particularly "allergic" to oscillations, smoothing or spurious (negative) values. It would be adequate to extend section 4.3 and at least acknowledge what to expect when using the LES for setups involving chemical and microphysical fields and refer to works discussing it (e.g., Light & Durran 2016, doi:10.1175/MWR-D-16-0220.1).

Specific comments:

- title (and elsewhere): is the project named ClimateMachine or ClimateMachine.jl?
- page 1:

  - "The use of Julia aims to increase accessibility..." - I doubt that employment of a new language with a still minuscule user base (https://insights.stackoverflow.com/survey/2021) helps to increase the accessibility. On the other hand, embracing Julia makes adherence to best practices feasible and manageable. As a result, there is a prospect for nurturing modularity, testability and clarity of the code (and the same for its legacy-free dependencies). There are novel coding, debugging, profiling, testing and documentation-generation tools available; the community is vibrant. All this works for the improvement of the code quality and development agility, which will certainly bring benefits to the developers' team and the software users... suggest devoting a separate paragraph to explain to the "average FORTRAN coder" the rationale and expected benefits from shifting to an entirely new simulation-engineering ecosystem, which is a bold and important step.
- page 3:
  - Lilly (1962) and Smagorinsky (1963)? (chronology of paper dates)
- page 4:
  - symbol conflict between omega in eq. (4) and domain-defining omega (the bold is merely noticeable)
- page 5:
  - Table 1. providing values of constants up to 5 significant digits and providing constants for ice thermodynamics seems unneeded given the scope of the paper. In turn, mentioning the CLIMAParameters.jl package, and the "Overriding defaults" section in its documentation seems more adequate!
- page 10:
  - "elements which share boundaries across MPI ranks": this is the very first mention in the text of MPI, ranks, shared boundaries - please first prepare the reader explaining the rank vs. core/CPU/node settings. Mentioning earlier on that the code uses MPI would also make it read better, even if this is "obvious".
- page 12:
  - unclear if the paragraph starting on line 307 on page 12 applies to section 4.2 only or to 4.1 as well
- page 13:
  - line 326: it seems more adequate to mention that the Siebesma et al. 2003 case is a non-precipitating boundary-layer shallow-cumulus convection case than bringing up the 1970-ties Barbados experiment origins of the initial thermodynamic profiles used.
- page 23:
  - line 505: mention of Coriolis forcing is likely misleading as it is not the general formulation as given in eq. (4), right?
- page 23, 24:
  - Figs 7 and 8 are presented in low quality (look like screenshots)
- page 24:
  - line 518: the different behaviour during the first hour spin up (ClimateMachine vs. PyCLES) calls for elaboration.
- page 30:
  - Code availability: please state the licensing terms of the code
  - line 590: rephrase around "Runtime" (not to confuse with compile-time/run-time)
- page 34:
  - NUMA, VMS, PPM acronyms appear only in the table and its caption (NUMA is not even mentioned in the caption), appendix B contains just a single sentence, suggest making the table and its discussion a proper part of the text (and if not, renaming the table from A1 to B1 is likely needed as it is in appendix B, not A).
- page 35:

    - there are two appendices labelled B
- references:
    - some entries have DOIs given, some not;
    - some journal names as abbreviated, some not;
    - title capitalisation is inconsistent (check proper names: taylor-green vortex).