Geoscientific

Model Development

Discussions

GMDD

Interactive

comment

Interactive comment on "HOMMEXX 1.0: A Performance Portable Atmospheric Dynamical Core for the Energy Exascale Earth System Model" *by* Luca Bertagna et al.

Luca Bertagna et al.

lbertag@sandia.gov

Received and published: 1 February 2019

We thank the reviewer for their comments and remarks. Here, we summarize how we addressed their comments.

1) RC: The authors do say that there was substantial work involved in this refactoring, but I am interested in them quantifying it in some way (if possible) - e.g., people hours, or lines of code touched, or percentage of HOMME code that is unchanged in HOMMEXX ?

AC: We estimated the effort in terms of lines of code. In the introduction, we added

this text: "This effort produced 13,000 new lines of code in C++ and 2,000 new lines of code in Fortran. Most of these lines replace code that solves the dynamical and tracer equations in HOMME. HOMMEXX uses HOMME's Fortran initialization routines."

2) RC: On the same line of thinking, can the authors speculate how this effort would compare to the effort that would be required to use OpenACC to port HOMME to GPUs. (Section 5 indicates that the authors are considering this undertaking as well.)

AC: We extended section 3.1 to give a better idea of where Kokkos stands in the broader context of performance portability. We listed other possible approaches, highlighting features of the different approaches. We believe this should help the reader to better understand how the current approach compares with others.

3) RC: I would have liked to see some performance metrics in the paper that are in more common use by the climate modeling community, e.g. simulated years per compute day or CPU-core hours per simulation year.

AC: We now include SYPD figures for the case of ne=120. For the other figures, we like to use a metric that is invariant to the number of elements in the mesh.

4) RC: Because the stated motivation of this work is running at exascale, it seems that the Dennis et al 2017 paper on this subject ("Preparing the Community Earth System Model for Exascale Computing") should probably be cited and mentioned in Section 1.

AC: Thank you for pointing out this very relevant reference; we added it to the introduction.

5) RC: Page 4: text refers to Figure 4 before Figure 3 (swap order of these figures?)

AC: Thank you, a few figures were re-ordered. We are planning to adjust the position of the figures once the manuscript is ready for its final typesetting.

6) RC: When referring to a resolution for the first time in the text via n_e (e.g., n_e = 240), please also mention the corresponding grid degree (as done for n_e = 30 on

page 5 and in Fig. 4 caption).

AC: Throughout the text, we added resolutions in degrees next to each n_e parameter.

7) RC: Section 2: Why only use the outer threading for HOMME? Please explain/justify.

AC: We edited the text as follows: "On conventional CPUs, HOMME supports outer OpenMP threading over elements and inner OpenMP threading over vertical levels and tracers. The outer threads are dispatched at the driver's top-level loop in one large parallel region. Each type of threading can be turned on or off at configure time, and, if both are on, they lead to nested OpenMP regions. If the number of elements in a rank is at least as large as the number of hardware threads associated with the rank, then it is best to enable only the outer threads. For this reason, in our performance comparisons we configure HOMME with outer threads only, unless stated otherwise."

8) RC: In general, I'd like to see more detail given in Section 3, given that this is a journal paper (as opposed to length-constrained C.S. conference submission) that should be of interest to climate modelers. It may be necessary to break up section 3 if significant more detail is added.

AC: In section 3.2 we highlighted in bold where a new design choice starts.

a) Section 3.1: I really would like more justification for the choice of using Kokkos. And I'd like to understand more of what was involved to use it - how about a simple example?

AC: In section 3.4.2 we added a code snippet showing how a common kernel (the divergence operator) is implemented in HOMMEXX using Kokkos. This snippet also highlights the three design choices mentioned in section 3.3.

b) Section 3.2: What is the correctness-testing build? (p.8. line 15). Is this what lines 16-17 are describing? I assume that this is not what you use for performance testing? Please clarify and consider expanding the correctness discussion.

AC: We expanded the paragraph a little bit, to help clarify what we meant with the concept of correctness-testing build.

c) Section 3.3: this section could benefit from some code snipets/pseudocode to clarify

AC: See answer to point a).

9) RC: page 11, line 9: What is a subview?

AC: We expanded the sentence to better explain the concept.

10) RC: I believe that Section 4 (results) could be improved quite a bit. It was a bit frustrating at times:

a) section 4.1: I'd like it to be easier to get the info on the various test machines and figure out which was which in the figures - so maybe put them in a bulleted list or large table. I'd suggest that the name of the platform - as referred to in the figures should be in bold and appear first.

AC: We added Table 1, Hardware Summary, to describe each machine/architecture.

b) For those readers not familiar with these DOE machines, it was frustrating that Fig. 5 listed "Edison", which was alternatively referred to by its processor (IB) in Figure 6. I kept having to search through information in paragraphs on pages 12 and 13 while looking at the plots. Similarly in section 4.3 (page 16, line 17), when a reference was made to the Xeon machines as a group, I had to search through these paragraphs. I think that not all readers are familiar with these machines, so please make it more accessible.

AC: We changed the name to "Edison-IB" in the figure to connect the platform used in the large-scale runs with the architecture also studied in the single-node runs.

c) page 14: this discussion of power consumption feels stuck in here. The rough guide-line numbers on power use (lines 10-14) should probably be included in the bulleted list or table of machines. Or maybe this power discussion should be in a new section

4.5 that happens after the reader has seen the other results - making it easier to follow. As it is, figures 5-8 are referred to in line 2 before they have really been presented in the text (which happens in later subsections).

AC: We moved the discussion on power consumption to a separate subsection, We also included the power consumption of each machine in Table 1. Power consumption is a complicated issue; in the absence of systems-level instrumentation, to which we lack access, we must make careful inferences. We are sure to be clear about this uncertainty in the text.

d) page 15, lines 10-11: this info feels like it should be in the intro of section 4 (page 12) - not stuck at the end of 4.1

AC: As suggested, we moved it at the beginning of section 4.

e) section 4.1: I also find it unhelpful to refer to the figures before "presenting" them. For example, page 15, line 1 and page 15, line 3: Theses comments about the plot attributes should go in the sections where the plots are described (4.2 and 4.3)

AC: Thanks. We are planning to adjust the position of the figures once the manuscript is ready for its final typesetting.

f) Consider combining 4.2 and 4.3 into a single "strong scaling" section. Also make the machine labels more consistent between plots 5 and 6, for example. (Even though 5 has fewer platforms.)

AC: We fused section 4.2 and 4.3 into a single strong scaling section.

g) Section 4.4: Overall, interpreting the results could be easier (more readable) by referring in the text to specific examples in the figures. For example, in page 16 line 29, say which platform in Figure 8 is the one that "does not always perform as well" with a large number of threads per rank.

AC: Thanks. We highlighted the architecture name in a few places where the generic

Intel Xeon name was used, or where no particular architecture was named (as in the example mentioned by the reviewer).

11) RC: Table 1: consider naming the kernels for those familiar with HOMME (rather than kernel 1, kernel 2, ...)

AC: Thanks; done.

12) RC: section 4.5, lines 14-15: It seems that Figure 6 does not indicate that V100 is strictly faster than HSW, though this text suggests that (the 1.2x to 3.8x)

AC: Thanks. We revised this sentence to refer to Figure 7 only. Figure 6 includes inter-node communication on a testbed; the intent of this sentence is to summarize on-node/device performance, in this case comparing 1 V100 device with 1 HSW node.

13) RC: page 13, line 8: I don't know whether to be concerned about what else may be hardware-specific in this single source code. What percentage of code is different? What are the types of code constructs that are problematic for Kokkos? What are the broader implications for other codes?

AC: We added the text: "We also use architecture-dependent code for three vertical integrals in the dynamics and the the tracer transport limiter. The total number of architecture-specific lines of code is approximately 800, out of approximately 13,000 C++ lines of code."

14) RC: (1) page 7, line 6: "around" => "on" (2) page 9, line16: "the the" => "the" (3) page 9, line 19: "nested loops," => "nested loops" (4) page 17, line 12: I'd assume that the reader is not necessarily aware of what nvprof is...should at least cite this.

AC: Thank you. Typos were fixed, and a reference to nvprof was added.

———————————————