GMD Reviews and Authors' Response concerning the paper "A Scalability Study of the Ice-sheet and Sea-level System Model (ISSM, Version 4.18)"

Yannic Fischler[1], Martin Rückamp[2], Christian Bischof[1], Vadym Aizinger[3], Mathieu Morlighem[4,5], and Angelika Humbert[2,6]

[1]Department of Computer Science, Technical University Darmstadt, Darmstadt, Hesse, Germany
[2]Alfred-Wegener-Institut, Helmholtz-Zentrum für Polar- und Meeresforschung, Bremerhaven, Bremen, Germany
[3]Chair of Scientific Computing, University of Bayreuth, Bayreuth, Bavaria, Germany
[4]Department of Earth Sciences, Dartmouth College, Hanover, United States of America
[5]Department of Earth System Science, University of California Irvine, United States of America
[6]Faculty of Geosciences, University of Bremen, Bremen, Germany

Correspondence: Yannic Fischler (yannic.fischler@tu-darmstadt.de)

1 General Comments

We thank Thomas Zwinger for the second review. He highlighted some points, which we have clarified to not state out misleading information. In the following we answer each point of the review individually.

2 Review by Thomas Zwinger

2.1 General Impression

The authors have addressed most of the points I raised during the previous review. In particular, ambiguities concerning the relation of the here presented scalability study to preparations for exascale computing have been corrected. I still have a single remaining comment concerning the lines of arguments from the reply letter and one new paragraph on optimization levels. Else, **I see this article ready for publication**, suggesting this **last remaining point** to be addressed.

2.2 Remaining points of critics

As I wrote before, I think that the main points of criticism have been addressed. It is a valuable contribution to focus on pure MPI performance, but at the same time important to avoid any impression that this suffices to guarantee performance on modern CPU clusters. The only remaining concerning point is, that **for me you seem to motivate a pure MPI scalability study by**

rendering SIMD to be of limited importance. In an added paragraph and your reply letter you mention the insignificant performance gap between binaries with and without AVX512. Whereas I get the impression that you interpret this insignificant performance difference of a few percent as a justification to focus on MPI scalability, only, I cannot avoid the suspicion that this rather points to the situation that the investigated code does not sufficiently utilize the long vector units on modern CPU's, in particular during the matrix assembly. In the end, one would need details on vector-unit memory utilization to draw a clear conclusion. In particular – referring to the justification in your letter of reply - I doubt that a pure MPI code without any SIMD optimization will excel on platforms like the A64FX CPUs in Fugaku (which derives the majority of its computing power from its 512-bit vectors). Most likely you will need the deployment of higher order elements (Isaac, et al., 2015) to effectively utilize those. With respect to modern architectures and even the memory-bounded nature of ISSM, I remain with my viewpoint that SIMD (AVX2/AVX512) performance is a to MPI scalability equally essential component to get a code performing on modern CPU clusters (e.g., Byckling, et al. 2017) and that it is important to not leave the reader with a possible interpretation that because of equal run-times of AVX512 enabled and disabled binaries SIMD would not be needed. Could you perhaps add a sentence on that aspect to clarify?

2.3 Detailed list of remaining issues

The list of issues is in the order of their occurrence in the text. Quote from the manuscript are kept in blue text.

2.3.1 abstract – line 2:

In this paper, we discuss the scaling of the Ice-sheet and Sea-level System Model (ISSM) applied to the Greenland ice sheet with up to G250 resolution.

As the definition of the different abbreviation of runs is happening somewhere in the text (Table 1), in my opinion this should not be used in an abstract – my suggestion to make the abstract more informative on that point would be to simply write out the range of resolution used in G250.

You are right. We have rewritten the sentence to "... with horizontal grid resolutions varying between 10 and 0.25 km".

2.3.2 page 2 – line:

While MPI scalability is not sufficient for reaching exascale performance, it seems to be necessary as corroborated, for example, by the MPI-based programming style of the 24 applications chosen for the exascale computing project (see www.exascaleproject.org)

This is good information to add to the article. Yet, I tried to find the list of those 24 applications behind the link given here. I did not succeed. Could you please either specify the location with a more detailed link or – ideally, as links do not have DOI's – add a citable paper/report?

As we are not aware of any citable paper/report we specified the link: https://www.exascaleproject.org/about/

2.3.3 page 6 – line 151:

The fact that we only use half of the available hardware cores is due to the fact that at G250 resolution, each MPI process requires 6.2 GB of memory.

I still think this would need the additional information that the excess amount of RAM is only needed in the lowest partition-count (=MPI task count) investigated – at least I presume this is the case. May I add that I still find it strange that you did not opt for a larger minimum partition amount or skipped the lowest partition number in order to completely fill all cores, which would resemble a more realistic production setup (and that is what you claim to investigate), as it is not economic to idle half of the resources one allocated for.

We have added additional information about the memory consumption.

Moreover, since ISSM is memory bound, the compute units of the CPUs are used more efficient while less threads/processes are running. Therefore the generalized statement "as it is not economic to idle half of the resources one allocated for" is not tenable.

Since scaling differs in different scenarios (pure node scaling most likely shows different behavior than scaling on multiple nodes), we see single node execution with 48 processes as the base of our scaling analysis and did not want to exclude it. If we aimed to use all 96 processes per node, we would have to assume a minimum of about 480 processes.

2.3.4 page 6 – line 157:

We compiled both ISSM and PETSc with "-O2" as well as with "-O3 - march=cascadelake -mtune=cascadelake". On a 480-core configuration, the entire calculation (without loading the model), took 1955 seconds compiled with -O2, and 1930 seconds when compiled with higher-level optimization. With 1536 cores, we observed execution times of 763 and 748 seconds, respectively. So the compiler optimization level has an impact of less than 2% in both cases. As the impact of the more aggressive compiler optimizations is low, we stick to - O2 as it avoids some potential numerical issues that can arise with more aggressive compiler options.

See my remaining point of critics above. I would rather interpret this as a sign of underutilized vector units than to use it as an argument that it is of no impact.

we agree that the impression should not be given that MPI scaling makes SIMD superfluous. Use of SIMD instructions is highly important for nodelevel performance which should be investigated separately. We added information to overcome the missleading impression.

2.3.5 page 26 – line 577:

For floating ice $h_b = -H\rho_i\rho_{sw}$ and $h_s = H(1 - \rho_i\rho_{sw}$ with ρ_{sw} being the sea water density.

Something went wrong with the typesetting of these two expressions.

Thanks. Typesetting of the equations is corrected.

2.4 List of typos and to content less important issues

2.4.1 page 4 – line 118/119:

The balance equations for enthalpy and momentum are solved consecutively, no iteration between the two modules within a is done within a time step.

This sentence reads strange to me.

We dropped the first occurrence of 'within a'.

2.4.2 page 18 – line 364:

There is a factor of about 34 between the minimum and the maximum DOF betweeen (-> between) G250 and G4000.

I also would swap the positions of "minimum" and "maximum" in the sentence, as maximum refers to G250 and minimum to G4000.

done

2.4.3 page 5 – table 1:

In the text (same page, line 140) you mention a minimum resolution of 150 m for G250, in the table you claim it to be from 0.25-10 km. A small inconsistency you might want to correct.

We have rewritten the sentence to: The highest resolution is ca. 250 m, which is close to the resolution of the bed topography dataset used in our study.

2.4.4 page 22 – line 117:

we (-> We) also want to emphasize here, that we investigated the performance of an HO application only and that other issues may arise for other momentum balance choices.

done

2.5 References

Byckling, M., J. Kataja, M. Klemm, and T. Zwinger (2017): OpenMP SIMD Vectorization and Threading of the Elmer Finite Element Software, In: de Supinski B., Olivier S., Terboven C., Chapman B., Müller M. (eds) Scaling OpenMP for Exascale Performance and Portability. IWOMP 2017. Lecture Notes in Computer Science, vol 10468. Springer, doi: 10.1007/978-3-319-65578-9_9

Isaac, T., G. Stadler and O. Ghattas (2015): Solution of Nonlinear Stokes Equations Discretized By High-Order Finite Elements on Nonconforming and Anisotropic Meshes, with Application to Ice Sheet Dynamics, SIAM J. Sci. Comput., 37(6), B804–B833, doi: 10.1137/140974407