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)

Copyright statement. ©2021 all rights reserved

1 General Comments

We thank the anonymous referee for the detailed and insightful comments. They helped us to sharpen some important issues and avoid potential misunderstandings.

In the following two sections we address each referee's comments in detail. The text in black is a verbatim rendition of the referee's comments. Our responses are typeset in the color of this paragraph underneath the referee's comments that they pertain to. In summary, the following points have been added/changed in the revised manuscript.

- The reviewers mentioned the somewhat vague discussion in the original manuscript regarding the future computing architectures and exascale computing. We clarified the pertaining points and clearly stated the goals and the scope of our study: an in-depth investigation of MPI-parallel scaling performance of ISSM and its main compute kernels.
- We added specific model throughput goals for use of ISSM within coupled ESM-runs based on the published ocean simulation throughput data for FESOM2.
- We included several relevant references as suggested by reviewers
- In the revised version, we (will) clearly state from the abstract to conclusion, that we only assessed the higher-order Blatter-Pattyn approximation part of the code.

– Last but not least – we corrected throughout the manuscript a number of small mistakes, sharpened vague formulations, clarified our claims – thanks to the careful reading and constructive criticism by the reviewers!

2 Review 2 by an anonymous referee

In the manuscript in question, the authors perform a detailed study of the overall scaling of the Ice-sheet and Sea-level System Model (ISSM) in the context of a Greenland ice sheet simulation and the higher- order (HO) Blatter-Pattyn model for the ice sheet velocities/momentum balance. The authors describe a low-overhead performance instrumentation using Score-P developed within this code base to enable continuous performance monitoring. The scalability study reveals that the matrix assembly part of the computation is the main bottleneck when it comes to scalability/performance, and should be examined further.

The manuscript in question is well-written, interesting and a good fit for GMD. My recommendation is publication following a minor revision. I ask that the authors please address the following questions/comments in their revision.

– The authors mention exascale readiness in the introduction, but there is no discussion in the paper of whether ISSM is portable to up-and-comping heterogenous architectures (GPUs). Is it? Can the present study be repeated on a set of GPUs? Some discussion of this is warranted.

  We have toned down the exascale issue. The purpose of this study was to investigate MPI-Scaling of ISSM. It is our conviction that MPI will be the backbone of any exascale architecture, and if the code already exhibits scaling issues at the number of cores that we can bring to bear in our configuration, it will be a show-stopper on the road to exascale. How exascale-systems will look like at the node level is hard to predict indeed. GPU-accelerated systems are a possibility, but the FUGAKU system at Riken, which employs Fujitsu's ARM-based CPU with vector extensions, claims (single-precision) exascale capabilities without GPUs (see https://www.r-ccs.riken.jp/en/fugaku/about/). The importance of MPI scaling is exemplified, for example, by the 24 applications chosen for the exascale computing project (see https://www.exascaleproject.org/, which employ MPI plus other techniques for exploiting GPUs and/or thread-level parallelism. Node-level issues, such as GPU accelerators or vector units, are an important possible issue in improving node performance, but orthogonal to this issue. For example, Brædstrup et al. (2014) presents a GPU implementation of an ice-sheet model. But they does not run all modules ISSM runs. We noted that ISSM is massively memory bound which suggests that significant algorithmic work needs to be invested before ISSM can profitably use GPUs or vector acceleration.

– On line 77 of the introduction, the authors mention that they are using a GMRES linear solver preconditioned with a simple block Jacobi preconditioner. The HO Stokes equations are symmetric. Have the authors tried using Conjugate Gradient? I additionally worry that the Jacobi preconditioner is inadequate for problems with floating ice, e.g., Antarctica, as shown in the references by Tezaur et al. and additionally: (1) T. Isaac, G. Stadler, and O. Ghattas, 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 (2015), pp. B804–B833, doi:10.1137/140974407 and (2) R. Tuminaro, M. Perego, I. Tezaur, A. Salinger, and S. Price. A Matrix Dependent/Algebraic Multigrid Approach for Extruded Meshes with Applications to Ice Sheet Modeling. SIAM Journal on Scientific Computing 2016 38:5, C504- C532. Are more sophisticated preconditioners required for Antarctica? Is scaling of those comparable to what you show for Greenland?

Within our setup we have almost no floating ice (the SICOPOLIS simulation used as initial condition prevents the occurrence of floating ice; floating ice in our setup might occur due to the interpolation of SICOPOLIS data to the ISSM grid), so we are not suffering under these conditions. However, we have not observed a poor performance with GMRES+bjacobi once floating ice is included. We tested various solver+preconditioner combinations (BICGstab, ILU, ASM, etc), but GMRES+bjacobi shows equal results to other combinations.

- Line 131-132: the linear solver convergence tolerances stated here seem loose to me. Have the authors verified that the solutions they have obtained at all their mesh resolutions are sufficiently converged/accurate? Accuracy/verification is an important thing to establish prior to studying scalability/performance.

The linear iteration is stopped if two convergence criteria are satisfied, which we think is rather strict. We verified that this choice of tolerance values is sufficient using a number of standard benchmark problems and selected real-life setups.

- It is really great that you have set up a workflow and are doing performance monitoring! I agree that without this, it is inevitable that performance will be compromised in a big code with a lot of moving parts. It isn't entirely clear to me when the performance monitoring tests are run. Does it happen every time there is a PR merged into ISSM? Does it happen automatically or it must be run manually? Some further discussion of this is warranted. All I found was the following phrase: "it is quite feasible to periodically run an instrumented version of the code as part of the regular work of domain scientists", which suggests the performance testing is not run regularly or automatically, but perhaps I am misunderstanding.

Instrumentation-based profiling using Score-P can be easily applied by domain scientists and performance engineers. We have not implemented an automatic workflow which might run on every merged PR. Instead of this we provide a white list filter file for Score-P which can be used when performance monitoring is desired.

- I did not really find discussion of how the load-balancing is done of the mesh on which the simulation proceeds, and how the mesh gets updated in a time-dependent simulation modeling ice sheet evolution. How do you partition the mesh? Do you repartition every time the geometry (active mesh) changes? Or you partition a mesh including active and inactive cells once in the beginning? The latter approach has the potential of giving a lot of procs with no elements or poorly load-balanced meshes. Some discussion of this is warranted. I think the load imbalances you talk about in the paper have a different cause, unless I am misunderstanding.

ISSM distributes all elements evenly using ParMETIS on the MPI processes during the initialization. This does not take into account whether an element is currently active or inactive. Therefore, there is no load balancing or repartitioning. We observed some processes which idle within the stress balance module, but other modules like the moving front module are load-balanced. First attempts to optimize the load balancing in the stress balance module lead to worse load balancing in other modules. Since the current load balancing does not have a strong impact on the scaling, we decided to not pursue this issue further.

- Is there any hope to improve the scalability of the matrix assembly?

  There is some hope, but no easy fix:

  - Currently, MatAssemblyBegin and MatAssemblyEnd are called right behind each other. However, it might be possible to perform some communication between these calls to exploit the asynchronous nature of these assemblies.
  - We have seen that most of the time is spent in creating the initial matrix structure. Therefore, code performance would benefit if more matrices would be reused in a fashion similar to our modification of the stress balance vertical module.
  - In the current implementation, the extrapolation in the moving front module is executed independently for each parameter. This step might run faster if all executions are combined in one large matrix or the matrix which is used for the first extrapolation can be reused for all following extrapolations.
  - Finally, a hybrid (MPI + OpenMP) parallelization would reduce the number of MPI processes which are involved in the assembly, while still employing potentially a large number of threads.

- It would be interesting to compare ISSM performance to that of other open-source ice-sheet models based on the HO Stokes equations. I am not suggesting to do this in the paper, just commenting.

  Yes, we fully agree and it would also be a good means to learn from each other and improve codes. Your comment is very welcome!

Minor comments:

- Change “to solve” in the first line of the abstract to “solving”.

  done

- Line 12: remove commas around “thus”.

  done

- Line 25: I don’t really understand the phrase, “standalone ice sheet projections are suffering from a large spread in climate forcing fields”. I don’t think “suffering” is the right word here. Please rephrase.

  We will rephrase this to ’challenged by’ in the revised version.

- After a colon, one does not use a capital letter. In many such instances, the colon should be replaced with a period.

  done

- Lines 131-132: I suggest stating what is \epsilon_i here so the reader does not have to refer to the appendix.

  Good idea! The revised version mentions the name of the covergence criterion in this sentence, too.

- Line 140: there is a space missing before “For profiling and tracing”.

  done

- Line 147: replace colon with period.

  done

- Line 171: “data are stored” instead of “data is stored”. Change “scalars, both” to “scalars. Both”.

  done

- Line 182: replace colon with period.

  done

- Line 190: change “The instrumented” to “the instrumented”.

  done

- Line 193: change “Sampling” to “sampling”.

  done

- All strong scaling figures: please move the linear scaling line to be either below or above all the other curves. It’s very hard to see it with all the lines on top of it (e.g., in Fig. 3).

  This is a good point - in the revised version we will move the line so that it is clearly visible!

- Line 288: change “It” to “it”.

  done

- Line 293: change “12.000” to “12 000”.

  done

- Some of the figure captions say “(draft)”. Was that intentional? I suspect it was not.

  done

- Line 420: change “to very a modest” to “to a very modest”.

  done

References

Brædstrup, C. F., Damsgaard, A., and Egholm, D. L.: Ice-sheet modelling accelerated by graphics cards, Computers and Geosciences, 72, 210–220, https://doi.org/10.1016/j.cageo.2014.07.019, 2014.