Review “A GPU-parallelization of the neXtSIM-DG dynamical core

In general a nice and well written manuscript. It runs through a number of general purpose GPU methods for optimization of the stress part of mEVP and inclusion of this into neXtSIM-DG. The manuscript also look into the how well these frameworks can be transferred to CPU’s which I think is important as well as the performance. The advantage Is that the frameworks are relatively easy to plugin. The disadvantage is that it is a black box to some extend.

The one thing that I miss the most is a real life example and not just the theoretical example. One of the challenges in traditional sea ice models is the inclusion of land and ocean points, where active sea ice points may be a minimal fraction of the total number of points. If I remember correct neXtSIM-DG has a fixed grid which could potentially leave a significant number of points icefree/inactive. I think that this at least deserves some comments and ideally a testcase. I do realize that running a new test case may be a bit out of scope.

The mEVP and the neXtSIM/Elasto-Brittle seems to be mixed. It may be true that the dynamics of neXtSIM are similar in the numerical sense but it does represent two different variation of sea ice dynamics. If neXtSIM has reverted to use mEVP (I don’t think is the case) then the refactorization within this paper is still of interest. I just don’t understand why this is investigated. For instance in table 1 timings of mEVP? It would be good if the authors put in a comment why the mEVP method has been refactored instead of the Elasto Brittle if this is the case. Alternatively please call the rheology something else than mEVP if this is not the rheology that is being refactored and describe the relevant rheology.

There are places like introduction of the mEVP solver and the discetization where the authors introduce it but cut it off very quickly with a reference. A bit more elaboration would help.

In addition the manuscript opens several topics for discussion and further development throughout the paper. This leaves the reader with the thought that optimization with the methods described here requires some soft of re-evaluation every time a new computer is used. Is this correct or not?

Abstract:
I would leave out the comment about energy in line 3. It is definitely of interest and important but it is not really the scope of this manuscript. It would be preferable with a conclusion in the abstract if this is deemed important enough to be mentioned in the abstract.

Introduction

Line 24: I am not sure that the order of the solver is the biggest inaccuracy of the solution in a climate model. I would add “numercial” in front of accuracy.

Line 34: I would not include stand-alone systems in this context as these are mostly used demonstrators and test as it is the case in the manuscript. The model would almost always be run as part of a coupled system, especially for longer simulations.

Line 41 - 45: Energy is mentioned in the abstract. If that is kept it should also be mentioned here.

Line 74 Section Sect. 2: I would remove Sect.

Line 96 and the first line in 2.1 is a repetition. I would remove one of them.

Line 99: Mapping of spherical coordinates (lat,lon?) to computational domain. A bit more description of this would be good. Is it also quadrilateral? Geophysical models refer to “lat/lon” but none of them use it for the computational grid. Which shape (if any specific) is the computational grid on?

Comparison of CPU vs GPU: Are all comparisons based on a 10cpu run?
Is the cpu runs executed on System 1?
One can always compare the compute time but it is difficult to compare an “openMP” code run on CPU directly with a GPU simulation.

Table 2 It would be beneficial to add the standard deviation of these timings in order to show if the difference in runtime is significant compared to the difference of the same run.

Section 3
This section explores a number of different optimization efforts. These are all carried out for 1 setup. The only viable solution is CUDA and KOKKOS. The rest fail due to crashes, limitation when using EIGEN. It is a concern that the use of Eigen code crashes when used within OpenMP/OpenACC. An explanation is found with the SYCL code. This leads to a question whether the choice of EIGEN should be reconsidered?

CUDA+KOKKOS: Has the authors tested if the results are different for different domains and resolutions? The memory available is described to be less than half of what is available. Does this imply that if the resolution were doubled then the memory access would be important? Should a new refactorization be carried out then?
Are the results bit for bit (probably not) when compared to openMP/single processor solutions? Do they change significantly? This has partly been checked later in the study regarding precision.

Section 4
I think that the description of the numerical experiment should be before the description of the different optimization methods as timings of the experiments are used. If I read correct it is the largest experiment that was used for section 3. Is that correct? It could be made more clear.

Figure 2: It would be helpful if the cpu runs had the same symbol or line style (e.g. square symbol or dashed line) and the GPU’s was marked in a different way.

Line 329: Ice models are often a part of an Earth System model

Line 357: The fact that this is an argument make a direct comparison impossible.

Figure 5. I am not entirely sure what the Noise amplitude is on the x axis. Please clarify.

Line 389: It is true that if I could get observation of concentration and thickness within 2% then I would be very happy. This uncertainty varies a lot depending on location and time of year. The example of a one day run satisfy one end of the time scales, however it does not say if this is good enough for e.g. longer climate series, where years are simulated. Conclusions on this should be formulated weak.

Line 397. What is the expected benchmark? This paragraph needs a few references or descriptions of why the speedup is anticipated to increase differently.

Line 425: If such a slowdown is imposed by higher order methods. Are they then worth the effort?

Line 491: This should also be considered on CPU’s

Line 506. These runs are already at ~5km and lower. It would be nice if the authors elaborated a bit on the different test domains, how the compare to reality and the influence on the results.

Table 4: Please specify ns, ng and na. Can this be moved closer to line ~440 where it is used.

Figure 8+ implementation within : Why is Cuda numbers presented if the implementation in section 5 is Kokkos