Review: MPAS-Albany Land Ice (MALI): A variable resolution ice sheet model for Earth system modeling using Voronoi grids, by Hoffman et al.

General Impression

The authors present a new land-ice model, MALI, implemented in the MPAS framework that includes two different solver versions: 1) a directly in the framework implemented lowest order Shallow Ice Solver and 2) a to the framework linked Finite Element (FE) first-order (Blatter-Pattyn) solver based on the code ALBANY. The model is validated against benchmarks, reaching from the EISMINT up to the MISMIP3D tests. Further, the authors present runs of both, the ice sheet as well as the hydrological model on a coarse-mesh Antarctic setup.

The whole project appears to be part of a bigger plan to integrate high-fidelity ice sheet modelling into Earth system models (ESM). This is a positive development and – in particular with the first order approximation – the community will get a tool that should be able to treat marine ice-sheet dynamics to a high accuracy within an ESM framework. Thus, from a general point of view, this publication is a good match to GMD's scope and I in principle support publication.

Major point of critics

The Antarctic test cases, both for ice dynamics (spinup) and the hydrology are run on equal-spaced (or equal sized, if talking of tessellations) cells with **horizontal resolution** of 20 km, which in my view is **insufficient to test the physics implemented**. The underlying hydrology as well as the dynamics of the fast outlet systems in Antarctica demand a way higher resolution (as outlet ice-streams usually are of similar or even smaller width). As I see it, one cannot assume that you capture the dynamics of such a system by placing a single cell to resolve it horizontally. In particular, you claim that the hydrology solver is grid-dependent, arguing that usually topography dominates the routing of water, which one certainly not resolves with 20 km. MPAS' and ALBANY's numerics to me seem to be optimized to be deployed on HPC systems as you are utilizing highly parallel scalable linear solvers provided by Trilinos. If so, I suggest to increase the number of cells to overcome that issue. Or – should you by some reason be confined in problem size – just focus on a single outlet system. In that connection I have a further question: On page two, line 21 you mention that your model has the possibility to utilize varying mesh density – why then using so large equal-sized cells in this test case? Could you resolve the critical areas with a sufficient dense mesh?

I like the elegance how you use the connection via the geometric dual between the triangulation of the FE model and the Voronoi tessellation to solve the **transport equations**. That seems to ease the problem of horizontal velocity interpolation. Nevertheless, for me it raises also a question on how **vertical advection** in those equations where it in my view is needed (heat transfer, passive tracer and age/depth) is introduced. You thoroughly explain how you split the vertical diffusion of the heat transfer equation (HTEQ) from the horizontal advection and ignore horizontal diffusion. But: how do you treat vertical advection? Usually, the in SIA and Blatter-Pattyn missing vertical velocity component is obtained from the three-dimensional field of horizontal velocities by utilizing the incompressibility condition, resolving it with respect to the vertical component and integrating it up to a certain z-coordinate (e.g., Greve and Blatter, 2007). Are you performing the same procedure? Else, I would suggest to give a clear explanation why you neglect vertical advection or – in case this applies – how it somehow implicitly is accounted for, for instance within the derivatives of your layer thickness in equation (20). In the latter case, please mention it in the text and

introduce this formulation into the HTEQ, such that the reader can link (22) and (35) to (20). In that connection, it would be from my point of view generally valuable to indicate which vectors and operators are defined in 2D and which in 3D. For instance, I read your HTEQ (22) to be defined for $i=1,2,3$, hence containing a vertical velocity that remains unaccounted for in the description of the numerical implementation given by (35). In my opinion, also the between equations varying index and vector notations do not really help to sort things for the reader.

Thirdly, I am missing an **analysis on the parallel performance** of your implementation. You are mentioning the recent HPC-buzzword "Exascale", but do not really provide any numbers. What would be interesting (at least to me): Is the ice sheet part negligible or significant in terms of computational resources needed if run inside an ESM? How is the performance of the Blatter-Pattyn model compared to a complete Stokes solution and how much more faster is the shallow ice solution compared to the 1st order solver? Do you have to adapt a general partitioning scheme for all MPAS-cores, or can you choose the number of CPU-cores to be deployed for each sub-model? Do you provide load balancing in case of changing meshes/domains? Those are questions I think would be valuable information to the reader, as you raise the interest by advertising this code to be massive parallel scalable and to operate within an ESM-framework. If the project has not proceeded so far that you would be able to provide these figures at the moment, then please mention it in the text, as it might be valuable information for people that else might think that this is ready for production.

Detailed Points

Points I see to be either corrected or elaborated on, sorted by their occurrence within the text.

page 3, line 23: *Shared memory parallelization through OpenMP is also supported, but the implementation is left up to each core.*

What does that mean? I would consider multi-threading to make sense only with multiple cores (talking of CPU-cores) that share a common memory. If you mean "core" in the sense of MPAS model component, then please use "MPAS-cores", as it else might be misleading.

page 4, line 29: *Planar meshes can easily be made periodic by taking advantage of the unstructured mesh specification.*

A minor issue, but I cannot follow your argumentation here: Why would the unstructured nature of grids ease the introduction of periodic conditions? In contrary, if I have a nicely structured mesh, the mapping of conformal nodes by indexes should be easier.

page 4, line 31: *Each core chooses its own vertical coordinate system.*

Again, I presume this is "core" used in the sense of MPAS model component but by a computational scientist it could be misinterpreted as CPU-core, also in view of a later occurrence (page 6, line 11) where you actually use "core" in the sense of CPU-core: *It is a massively parallel code by design and recently it has been adopting the Kokkos (Edwards et al., 2014a) programming model to provide manycore performance portability (Demeshko et al., 2018) on major HPC platforms*

page 5, line 18: *Additionally, the JIGSAW(GEO) mesh generation tool (Engwirda, 2017a, b) can be used to generate high quality variable resolution meshes with data-based density functions very efficiently.*

Please, explain if and how this is relevant to your application. Do you for instance use some Hessian of velocity field in order to manipulate mesh density?

page 6, line 27. eq. (2): $$\frac{\partial \sigma_{ij}}{\partial x_j} + \rho g = 0, \quad i,j = 1,2,3$$

Gravity should be displayed as a vector in this balance and hence is missing the index *i* .

page 7, line 9: *In Equation 5, A is a temperature dependent rate factor, n is an exponent commonly taken as 3 for polycrystalline glacier ice, and γ is an ice "stiffness" factor (inverse enhancement factor) commonly used to account for other impacts on ice rheology, such as impurities or crystal anisotropy (see also Section 6.1).*

If you are referring to the commonly applied enhancement factor, E, as a pre-factor to the rate factor in Glen's flow law, then not its inverse (as at least I would interpret it), $\gamma = E^{-1}$, but rather $\gamma = E^{-1/n}$ would be correct. This might not be unimportant to spell out, as you use this factor in the inversions and – presumably – plug it back into your forward model.

page 7, line 18: in which A_0 is a constant, (T^*) is the absolute temperature (i.e., corrected for the dependence of melt temperature on ice pressure), Q_a is the activation energy for crystal creep, and R is the gas constant.

I would see T^* (BTW. why do you put it into brackets?) to be (quoting Greve and Blatter,2007) the *temperature relative to the pressure melting point* and not the *absolute temperature*.

page 8, line 2: *Ice sheets typically have a small aspect ratio, small surface and bed slopes, and vertical pressure distributions that are very nearly hydrostatic.*

In the light that you apply Blatter-Pattyn, in my opinion it would be better not to refer to pressure but rather the vertical distribution of the vertical normal Cauchy stress, i.e. the *hydrostatic stress approximation* (Greve and Blatter, 2007).

page 9, line 3, equation (15): $$2\mu\dot{\boldsymbol{\epsilon}}_1 \cdot \mathbf{n} + \beta u^m = 0, \quad 2\mu\dot{\boldsymbol{\epsilon}}_2 \cdot \mathbf{n} + \beta v^m = 0, \quad \text{sliding},$$

If the symbol μ stands for the effective viscosity (that is what I presume), you are deviating from the symbol η_f you used in equation (9) – which by subscript (e -> f) again deviates from its initial definition in equation (5) and also from the definition in the table of symbols - and you should correct it. If it defines something else, then you should declare it. Same accounts for equation (16). For someone who wants to really understand what you are doing in your code, consistency in notation is very helpful.

page 9, line 5, equation (16): $$2\mu\dot{\boldsymbol{\epsilon}}_i \cdot \mathbf{n} - \rho g(s-z)\mathbf{n} = \rho_o g \max(z,0)\mathbf{n},$$

The first term in equation (16) to me reads a scalar-product between two vectors, which leads to a scalar, but the other terms to me read as a with the surface normal aligned vector. To me this seems to be inconsistent.

page 12, line 26: … *k is assumed constant and uniform*

The temperature dependent heat conductivity of ice varies about 34% between atmospheric pressure melting point and -50 C (Greve and Blatter, 2007). As this is a temperature span that easily can occur in a single column of the Antarctic ice sheet, could you please elaborate how big of an error do you introduce by using a constant value, and which value (corresponding to which temperature) you are choosing?

page 19, line 17, equation (52): $$\boldsymbol{\tau_b} = C_0 N \boldsymbol{u_b}^m$$

A minor issue, but this type of notation leaves room for (mis)interpretation of a vector subject to an exponent. Looking at equation (15), I conclude you mean that the exponent applies component-wise. My suggestion: Either mention that in the text or perhaps write (52) out on a component level – just like (15).

page 21, line 1*: … , suggesting a basal friction law based on the subglacial hydrologic state could be configured to yield realistic ice velocity.*

This links to one of the major points of criticism: I do not think you can draw that conclusion from a run done on 20 km grid spacing, when the hydrology in the major outlet systems takes place on subgrid scale.

page 21, line 10*: The calving front is maintained at its initial location by adding or removing ice after thickness evolution is complete. This option does not conserve mass or energy but provides a simple way to maintain a realistic ice shelf extent (e.g., for model spinup).*

I don't get the point of this statement: Doesn't this apply to any sort of calving process? In my opinion introduction of discontinuous calving processes in a continuum model, simply by the fact that it instantaneously removes parts of the continuum, by definition drains mass and energy. I guess you referring to the artificially added mass – then it would be interesting to know how you implemented this. Do you add a layer of minimum ice-depth of 1m as on land?

page 31, line 10*: The Antarctica model configuration we demonstrate here uses a 20 km uniform resolution mesh, …*

As I explained before, I see 20 km resolution as a problem, in particular in connection with the marine ice sheet dynamics. In particular, as you claim yourself on page 29: *Thus for marine ice sheets with similar configuration to the MISMIP3d test, we recommend using MALI with the grounding line parameterization and a resolution of 1 km or less.*

Typos and type-setting

page 7, line 23: *Ice sheet models solve Eq. 2-8 with …*

Please, check how GMD wants to have references to equations. I see inconsistencies throughout the text, with either the word Equation/s spelled out or abbreviated (like here). Also check, whether you should put the equation numbers into brackets or not.

page 17, line 7: … *space can be represented by the effective water depth in the macorporous sheet, W:*

macorporous -> macroporous

page 35, line 17: *An implicit subglacial hydrology model based on existing ~~such~~ models (Werder et al., 2013; Hoffman and Price, 2014) is under development using the Albany framework.*

To me this sentence reads strange, except if the word “such” is removed

page 42, line 7: Bueler, E., Lingle, C. S., Kallen-Brown, J. a., Covey, D. N., and Bowman, L. N.: Exact solutions and verification of numerical models for isothermal ice sheets, Journal of Glaciology, 51, 291–306, doi:10.3189/172756505781829449, http://openurl.ingenta.com/content/xref?genre=article{&}issn=0022-1430{&}volume=51{&}issue=173{&}spage=291, 2005.

Those longish (and in this case for my browser not existing) links could be dropped - these occur several times throughout the list of references. In general, I think the DOI link is sufficient and no other link is needed. Perhaps GMD has a policy for reference-styles.

References

Greve R. and H. Batter (2007): *Dynamics of Ice Sheets and Glaciers*, 1st ed. , Springer, Berlin