Geosci. Model Dev. Discuss.,
https://doi.org/10.5194/gmd-2018-194-RC2, 2018

Geoscientific Model Development Discussions
Open Access

EGU

Interactive comment on "Implementation and Performance of Adaptive Mesh Refinement in the Ice Sheet System Model (ISSM v4.14)" *by* Thiago Dias dos Santos et al.

D. Martin (Referee)

dfmartin@lbl.gov

Received and published: 30 October 2018

Overview

Adaptive mesh refinement (AMR) is a useful numerical tool in ice sheet modeling due to the wide range of resolution requirements, particularly for accurately modeling marine ice sheet dynamics. A large body of evidence points toward the need for very fine model resolution concentrated near grounding lines (GLs), which can migrate over hundreds of kilometers in long-time simulations. AMR provides the capability to efficiently deploy the required fine resolution adaptively only where needed, resulting in

Printer-friendly version

Discussion paper

potentially large savings in computational resource requirements. This paper describes an approach taken to add adaptive mesh refinement to the finite-element-based ISSM ice sheet model. The authors present a description of the algorithm in use, including examples incorporating two different pre-existing remeshing packages. The paper is fairly well-written and quite readable, although could use some editing to fix some English-grammar issues. It does a fairly good job of representing the details of the approach taken and represents a useful addition to the literature. I support publication after some relatively minor changes.

The movies in the supplement are very short and fast, making it hard to see the details of what is happening. I'd suggest slowing them down somewhat (maybe by a factor of 2 or so?). They do look impressive, however.

Another general comment – all of the results are presented in terms of GL position, which can be a bit tricky to work with. I'd suggest supplementing with an integrated quantity like ice volume, grounded area, or volume above flotation if that presents a clearer picture.

Specific comments

1. p1, line 9: You say here that you use different combinations of the two error estimators, but I didn't see any combinations in the paper – the examples appeared to use *either* distance from the GL or the ZZ error estimator, but never both. In practice, we've found that the combination of the two works best (although as is pointed out elsewhere in the text, we generally use the undivided Laplacian of the velocity field as a proxy for the truncation error).
2. p2, line 10: You might also want to mention the role of ice-shelf buttressing for completeness.

Printer-friendly version

Discussion paper

3. p2, line 16: I'd suggest changing "high grid resolution" to "sufficiently high grid resolution" (or "sufficiently fine") – the whole point here is to (locally) apply sufficient resolution to resolve the dynamics in play. What is "sufficient" depends on where you are.

4. p3, line 5: BISICLES is actually at its heart a 2D model, although "$2\frac{1}{2}D$" might be more descriptive due to the vertical reconstruction entailed by the Schoof-Hindmarsh scheme along with the fully 3D temperature/enthalpy discretization.

5. p4, line 20: It would be helpful if you could add a figure showing examples of how the mesh refinement occurs for the two schemes (start with a coarse base mesh, then show how the mesh is evolved in a picture or two). It's too hard to see that level of detail in your figures and animations. I think I can imagine it, but a picture would be helpful here.

6. p4, line 21: "numerical perturbations" – what you're really talking about is "numerical *errors*" (so it would be good to say that explicitly) introduced due to interpolation. It's important to call this out as a source of error that is introduced by the AMR scheme. The reality is that AMR schemes introduce errors; the goal is to ensure that these errors are outweighed by the improved accuracy and efficiency. The fact that you're able to minimize these by minimizing the need to interpolate from old fine-mesh region to new fine-mesh region as the mesh evolves is an important advantage of your approach.

7. p5, line 21: Are you saying that you are reducing all of the interpolated fields (thickness, bedrock, etc) to processor 0 and then broadcasting them all out? If so, this all-to-one reduction will eventually overwhelm you as you push to larger problems at much higher concurrencies. If that's not what you're doing, please clarify which fields you're talking about here.

8. p7, line 9: I suspect that the ZZ error estimator will be much more useful when ap-

plied to either the velocity field or the stress tensor, since this will indicate where there is instantaneous error in the dynamics (which is what will be improved via AMR). Since ice thickness is essentially a time-integral of the divergence of the velocity field, applying an error estimator to the thickness field is likely to indicate where errors have accumulated over time; adding refinement in those regions is likely too late.

9. p10, line 23: Which is correct? You should use a uniform (very-)fine-mesh solution as a comparison.

10. p11, line 14: Stability isn't the only reason to refine your timestep – assuming your time integration scheme is consistent, you should also see an $O(\Delta t)$ or possibly $O(\Delta t^2)$ component of the error, which can become important if you refine only spatially without a corresponding reduction in the timestep.

11. p11, line 19: If I understand the approach suggested here, I think the proposal is to initialize the model using a uniform coarse-resolution solution, and then turn on AMR and add resolution as you begin to evolve the ice sheet (and justify it by stating that's how one begins a realistic simulation based on observations). This is almost certain to produce numerical artifacts due to the sudden change in the mesh setup. We have found that it's important to initialize the model (including data assimilation, inverse-problem solution, etc) with at least the resolution configuration that the model will start with to produce an initial condition with as few numerical artifacts as possible (and you actually essentially make that point in the next sentence about requiring both the steady-state and perturbation experiments be carried out using AMR). I realize that this particular sentence only applies to the performance experiments, but it represents a bad idea which might be misconstrued as a suggestion. Why doesn't it make sense to simply run each mesh configuration forward from its own self-consistent steady state? That would be more representative of how one would initialize the model in a realistic config-

Printer-friendly version

Discussion paper

uration.

12. section 4.2: There is one more figure which would be very useful here to supplement Figure 3 – a plot which shows both element counts and solution time for each case, each normalized against the values for the equivalent uniform fine-mesh solutions, on the same graph (along the lines of Figure 21 in Martin and Colella, JCP, 2000). This has the advantage of showing both the relative savings due to AMR while also illustrating the overhead due to AMR – as represented by the gap between the cell counts and execution times. (If fully computing the uniform fine-mesh solution is too expensive, you can likely compute a few timesteps and extrapolate).

13. p12, line 6: Using estimates of the error is also commonly used in AMR models in the community. For example (as you mention elsewhere), BISICLES often uses the Laplacian of the velocity field as an estimate of the truncation error (in a second-order discretization). Others do as well, but I don't have them at hand. (You seem to imply that using an error-estimator is novel)

14. p13, line 3: I think you can make a stronger statement than "an error estimator may be more appropriate" here – I think you could say "is more appropriate", or, if you'd prefer "is likely more appropriate". This is a nice example of why it is important to understand the error structure of your problem when constructing refinement criteria.

15. p13, line 25: Is there a reason you don't demonstrate an example which uses the combination of refinement criteria that you recommend here?

16. p13, line 35: In general, actual savings due to AMR are fairly problem-dependent.

17. p.14, line 12: "mainly in setups where bedrock induces strong buttressing" – I'm not sure that's really all that relevant here. The additional error (and hence

the additional refinement) occurred due to topographical features which induced complicated stress distributions unrelated to the buttressing.

18. Algorithm 1: It appears that in your algorithm, the initial timestep is taken on the initial coarse mesh before any refinement is implemented. Is that the case? If so, then you have the problem that the initial coarse-resolution timestep can contaminate the solution with initial-time errors. In general, as I mentioned earlier, you're a lot better off if you initialize the problem with the AMR mesh you're going to be running with.

19. Figure 5: This figure suffers from the choice of colormaps. White lines are hard to see, particularly in the more highly-refined regions, and the choice of blue dots to represent the AMR GL position makes that hard to discern from the blue velocity colormap. The small size of the plots only adds to the difficulty. What about potentially showing the mesh lines colored by the velocity field colormap, and then using entirely different colors for the grounding lines?

20. Figure 6. This figure is central to the entire effort, and raises a few questions/issues. I find quite a bit puzzling here:

    (a) Why is the ZZ error estimator only being used for the NeoPZ case? Is it only available for the NeoPZ runs? (I couldn't find any statement to that effect in the paper, although it's possible I overlooked it).

    (b) Would it be possible to use different line types (or possibly different colors) to distinguish the different lines? Seeing how the different cases follow the uniform-mesh case is part of the goal (as distinct from looking at individual data points), and that's hard to separate out when all of the lines are identical, particularly for the finer-resolution parts of the NeoPZ plot.

    (c) Do you have any idea why R30 improves continuously woth increasing resolution for the NeoPZ runs, but stagnates for the Bamg runs? Can you apply the error estimator to see what's going on there?

(d) The drop-off of the AMR-ZZ case at the finest resolution for the NeoPZ case is problematic, because it *could* represent a saturation of the ability for AMR to improve the accuracy of your solutions and provide fine-grid accuracy. There are a number of reasons why that might be the case, some of which would indicate potentially serious limitations on your scheme. One possibility, as mentioned earlier, is that temporal errors are beginning to dominate, due to the constant timestep across all resolutions. I suspect, however, that what's happening here is the saturation of your error indicator. The error threshold for refinement should scale appropriately (proportional to dx^2, perhaps?) to match the target resolution. If you're using the same numerical value in your error-tagging criterion for all runs (you don't actually mention in the paper how you're choosing that parameter – you should say that), it will act as a switch and turn off once the solution is accurate enough to match the criterion. Trying to ask for more refinement after that won't actually add much, and so the solution improvement will stall. If that's the case, then there are two potentially useful outcomes:

   i. Tightening the refinement criterion for the 0.25km case will improve the accuracy of that result and reduce or eliminate the stalling apparent in figure 6, and

   ii. The corresponding *loosening* of the refinement criterion for coarser cases will potentially reduce the number of refined cells without a corresponding effect on the solution accuracy (improving the computational efficiency of the AMR scheme)

21. Figure 7: What does the error look like for uniform fine-mesh solutions? Also, the white mesh lines are even harder to see here than they were in Figure 5.

Technical corrections

1. p1, line 4: “of grounding line” → “of a grounding line”
2. p1, line 6: “adaptive mesh refinement approach, AMR” → “adaptive mesh refinement (AMR) approach”
3. p1, line 9, elsewhere: “MISMIP3d setup” → “the MISMIP3d setup”
4. p2, line 4: “the collapse of WAIS is based on” suggest replacing with “projections of the collapse of WAIS are based on...”
5. p2, line 20: “flux condition at GL...” → “a flux condition at the GL..”
6. p2, line 24: “allows to apply resources...” → “allows resources to be applied...”
7. p3, line 11: “ice flow Elmer/Ice” → “ice flow model Elmer/Ice”
8. p4, line 25: “high adaptive” → “highly adaptive”?
9. p5, line 4: “transient simulation” → “transient simulations”?
10. p5. line 23: “being ρ_i the ice density” → something like “with ρ_i the ice density...”
11. p5, line 24: “vertical plane view” – I’d suggest “vertical cross-section” (also in the caption for fig. 1)
12. p6, line 1: “of GL” → “of the GL”
13. p12, line 10: “what is useful...” – do you mean “which is useful...” (if not, then I’m not sure what this means)
14. p12, line 15: “scheme” → “schemes”

15. p.14, line 1: "even with a hundred..." → "even with hundreds..."
16. p.14, line 7: "implemented here dynamic..." → "implemented dynamic..."
17. p14, line 23: "compenting" → "competing"

Interactive comment on Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2018-194, 2018.