Response to Reviewer #3

March 13, 2015

1 Summary

The first order, or Blatter-Pattyn equations describing the diagnostic solution to ice flow are discretized using the finite element method and solved using modular and highly scalable software libraries. The libraries provide built-in facilities for UQ (uncertainty quantification) and optimization via AD (automatic differentiation). Similar models have been constructed in numerous recent publications (see references in this paper), and its completion is not novel enough to merit publication. However, several interesting applications of the model are performed to explore its numerical properties, making this an interesting paper. The applications are: 1) the homotopy, or a systematic method of reducing a viscosity regularization parameter which improves convergence, 2) the manufactured solutions which allow for verification of the conservation of momentum solutions, 3) application of a multi-level algebraic preconditioner, and 4) exploration of the vertical mesh spacing requirements of the model. Each of the 4 applications have been performed in some way by others, but I am not aware of another paper featuring as systematic and rigorous approach to model testing. Because of this, ice-sheet modelers should find the results interesting and worthwhile to compare their own models to.

We thank the reviewer for his/her thoughtful feedback, which has been addressed in the revised manuscript, and has helped us to improve our paper. All changes to the original manuscript are marked in blue in the revision. Responses to your comments (including those above) can be found below, also in blue.

2 Review

The utility of this paper is found in its applications of the new momentum solver. I will treat the novelty and utility of each of the applications, in turn.

Homotopy is cool, but it looks like the stopping value of the regularization parameter is close to the value used by many (10^{-10}), which makes it seem more like an additional burden faced by this model, rather than a feature that improves numerical performance, particularly in the numerous models that do not have (or perhaps fail to report on) difficulties reaching convergence. I might find the treatment more satisfying if the sequence of alphas had been continued to make the regularization exactly zero, assuring the choice of parameter does not influence results. I am not clear on how the sequence of values for alpha are selected, this should be clarified.

The sequence of values of α's were calculated using a natural continuation algorithm implemented in the *LOCA* package of *Trilinos*. Please see our response to your comment #1 in Section 3 below for more details.

AMG is nice, but somewhat lacking in terms of details of how it is implemented. The paper provides a good general overview of the method, as well as insight into the particular challenges posed by ice-sheet modeling; low aspect ratios and their impact semi-coarsening techniques. However, the discussion was not useful to a modeler interested in how AMG might be applied to a particular matrix generated by Stokes equations. This comes back to the editor's note about code release and versioning; for the material on AMG to be useful, it needs to be (minimally) accessible in the form of source code, and preferably accompanied with a better explanation. The authors do state that a second paper dealing with AMG is in preparation, and I think that some details can be delayed, but more implementation specifics should be provided now.

It was brought to our attention by the editor of *Geophysical Model Development*, as well as several reviewers, that all papers submitted to this journal must include a section entitled "Code availability" that contains instructions for obtaining the code described in the paper and licensing information. We have added such a section to the paper. In it are paths to the currently released versions of *Trilinos* and *Albany*. The AMG algorithm is available in the *ML* package of *Trilinos* (version 11.12 or later). An addendum to (Gee, 2014) describes how the multi-grid semi-coarsening algorithm is specified from a user's perspective (Tuminaro, 2014).

MMF Equations 22-23 are the shallow shelf approximation (SSA) equa-

tions, right? Where are the vertical shear terms (du/dz, dv/dz, see eqns 2-8)? Are these zero in the case where the surface slope is zero? That makes sense, but I wonder if the system has been simplified to the point where verifications is done on a specialized case of the FO equations (SSA) and not the true FO system? In this case, you've lost the most interesting part of the FO eqns; the ability to estimate BOTH vertical shear AND membrane stresses. This is probably the biggest issue with the paper as it is now written.

Equations (22)–(23) were obtained from the 3D FO Stokes equations by neglecting the $\partial/\partial z$ terms. This is why the vertical shear terms are missing from the equations. The test cases were not intended to verify the 3D FO Stokes equations: rather, they were intended to be used as part of a multi-stage code verification that includes also verification of the 3D FO Stokes equations using code-to-code comparisons and mesh convergence studies on realistic geometries (Sections 5 and 6 in the paper). We have attempted to make this more clear in the paper, and also have made it clear that our MMS problems are for a 2D version of the FO Stokes equations, not the 3D equations.

We feel despite being simplified, our MMS test cases are nonetheless useful. The task of deriving source terms for an MMS study for the 3D FO Stokes equations is cumbersome, if not intractable. In contrast, our MMS problems are simple enough to be implemented by anyone simply by referring to the expressions in our paper.

We do agree with the reviewer that a test case based on equations that include vertical shear would be worthwhile. To address this point, we have derived an MMS test case for the FO Stokes equations in the x–z plane (obtained by neglecting the v and $\partial/\partial y$ terms in the equations), and showed some mesh convergence results for our code on this test case. Please see Section 4.2 of the revised manuscript.

The exploration of the convergence for different mesh resolutions is sensible and provides clear results. It's a little difficult to relate reported errors to something practical, like; how many layers are needed to eliminate errors in prognostic runs of several hundreds of years. That is probably not an easy question to answer, but if the paper is going to have significant impact, making a clear statement about how important the errors are would be helpful.

3 Summary Statement

This is a long and important paper; providing a new means of solving a complex set of equations, and giving significantly expanded means of verifying the results are correct. There are some matters that have to be addressed before it is publishable. They are:

1. Minor changes in the section dealing with homotopy to explain the sequence of values that are used for alpha. It would be really nice to know if homotopy is needed in time dependent problems, but it seems beyond the scope of the problem. Could homotopy be used when moving from coarser to finer grid resolutions to see if it reduces Newton iterations?

   For time-dependent problems, we would recommend using the homotopy only in the first time step (diagnostic solve), since the converged solution at time t_n is likely to be a reasonable initial guess for Newton's method in time t_{n+1}, for $n \in \mathbb{N}$. We have noted this in Section 3.1.1, but do not provide numerical results, as time-dependent simulations are beyond the scope of this paper.

   We had stated in the paper that the sequence of α's are calculated using a step size control algorithm. We have added a sentence to add more information on this algorithm, and added a reference to the LOCA manual (Salinger et al., 2002) where the detailed algorithm is published. The Allgower reference is appropriate as well.

   Moving from coarser to finer meshes as a method to converge the nonlinear system on the finest mesh more rapidly (often called mesh sequencing) would be a valid approach, and could likely reduce computation time for convergence on the finest mesh. Homotopy would not be appropriate to step through mesh resolution because mesh size is not a continuous parameter. For this same reason, the mesh sequencing approach might be fragile in that there is no easy path to recovery from a failed step. Furthermore, this would need to be implemented with scripts that create and interpolate between meshes, a process that does not match HPC platform usage (batch queuing systems) very well. We have not implemented a mesh sequencing approach.

2. Major changes should be made to the section on AMG, providing better insight into the implementation details. This might be most easily done

by, Release source code, identify where it can be accessed, and provide versioning, as per the requirements of GMD.

We have added the "Code Availability" section describing how the codes can be obtained, in particular *Trilinos*, which contains the package where the AMG preconditioner is implemented, called *ML*. We have also added a reference to a document (Tuminaro, 2014) that describes how the multi-grid semi-coarsening algorithm is specified from a user's perspective (Section 3.1.2).

3. Reconsider the approach to manufactured solutions, which in the present form do not appear to be verifying the solution of the FO equations.

   We did reconsider the approach, as explained above. We have attempted to make it more clear that our MMS problems are for a 2D version of the FO Stokes equations, not the 3D equations. We have also extended our MMS test cases to include one for the x–z FO Stokes equations (obtained by neglecting the v and $\partial/\partial y$ terms in the equations). These provide a test in which the equations have vertical shear terms, and in which the solution is more physical (the sum of SIA and SSA solutions). The new MMS test case and some convergence results for this problem can be found in Section 4.2.

4. Provide a roadmap detailing how this advancement in a single component of an ice-sheet model will integrate with other components to be more comprehensive and useful. How will the energy balance be computed? What about ice transport? Integration with climate models? I understand that this effort is coming from the national laboratories, where there is significant activity on all of these components, but the authors need to demonstrate that this is destined to become more than 'just' another momentum solver with no capacity for prognostic ice sheet modeling.

   Please see our response to Comment 2 below.

All of these can be addressed in the context of minor revisions. Unless I misunderstand, the MMF will require serious reconsideration, but there is enough in this paper that without the MMF the paper is still interesting.

Specific, but less significant issues in the text.

Abstract:

- Why is "Template-Based Generic Programming" capitalized?

  We agree that it need not be capitalized, and have changed it in the paper.

Introduction:

1. "A primary development focus has been on improving the representation of the momentum balance equations over the 'shallow ice' (SIA; Hutter, 1983) and 'shallow-shelf' (SSA; Morland, 1987) approximations through the inclusion of membrane stresses over the entire model domain."

   Careful, membrane stresses are supported over the entire domain when using the SSA. Wouldn't it be more appropriate to say through inclusion of BOTH vertical shear AND membrane stresses over the entire model domain (CAPS just for emphasis). allowing for a quantifiably optimal match between modeled and observed velocities There is no guarantee that this match is globally optimal.

   We have made some changes to this portion of the text along the lines suggested by the reviewer in the first part of this comment. We do not follow entirely the second part of the comment about optimal matching between observed and modeled velocities. We do not make a claim that the match between these two quantities is globally optimal.

2. p 8083, line 1-5: There is mention of integration into ESMs. It is possibly true that the model can be easily integrated into ESMs, but if how that is done is not discussed in the paper, then it really shouldn't be mentioned in the introduction because the issues is complex enough that in a journal like GMD, such claims can not be made without some supporting documentation.

   How we plan to couple to ESMs and do prognostic model runs is a good question. The reviewer is correct that *Albany/FELIX* can only be used for the diagnostic stress-velocity solve, and does not discretize the ice temperature and thickness evolution equations. To do prognostic runs and couple our code to ESMs, we have coupled *Albany/FELIX* to two

other land ice models: the *CISM* (Community Ice Sheet Model) and the *MPAS-LI* (Model for Prediction Across Scales - Land Ice) codes. In the resulting dynamical cores (dycores), termed *CISM-Albany* and *MPAS-Albany* respectively, the steady state stress velocity solve occurs in *Albany/FELIX* and the ice sheet thermal and geometric evolution is calculated in *CISM* or *MPAS-LI*. The details of these ice sheet modeling frameworks will be presented in a subsequent paper, so we had intentionally omitted the discussion of these codes here. We agree with the reviewer that some mention of these dycores is worthwhile so the reader can see how we will do prognostic solves and couple to ESMs. It has been added to the "Conclusions" section of the paper. If the reviewer would like to learn more about *CISM-Albany* and *MPAS-Albany* and see some results from prognostic runs produced by these codes, there are some presentations on the subject available at: `http://www.scidac.gov/PISCEES/presentations.html`.

3. p 8087, line 5 not an energy balance but a conservation of energy model.

   We agree that "conservation of energy" is more appropriate of a term than "energy balance", and have changed the phrase.

4. p 8090, line 11 "horizonal" $\rightarrow$ "horizontal" x2

   This was a typographical error, which we have corrected.

5. p 8090, line 26 How is the domain decomposition performed?

   We partition in a uniform contiguous domain fashion a 2D cross-section of the full 3D mesh, so that in the 3D mesh decomposition, all elements having the same x and y coordinates are on the same processor (see Section 6.1). The partitioning was done using the decomposition utility (called `decomp`) available as a part of *Sandia Engineering Analysis Code Access System (SEACAS)* database of *Trilinos* (see the "Code Availability" section). This utility was used to create a linear decomposition of the 2D mesh. These details have been added to the text.

6. p 8091, line 5. Is the viscosity differentiated too, or just the strain rates? That is to say, its the so called "incomplete adjoint" (Goldberg and Sergienko 2010), or complete adjoint?

   In our implementation, the viscosity is differentiated, not just the strain rates. Since the problem is self-adjoint, the Jacobian is equivalent to

the adjoint matrix. We would like to note that this is only relevant for optimization problems, which go beyond the scope of this paper.

7. p 8093 line 21-23. OK, I don't really know much here, but I thought the basic idea of multigrid techniques was to use multiple resolutions to speed the rate of information transfer across the domain. It's more difficult for me to understand error capture from a high level.

   There are different views of how multigrid works. The basic principle is to utilize multiple resolution versions of the original problem to accelerate the iterative solution procedure. The key idea is that smooth error components (in the current solution approximation) can be efficiently damped by applying a simple iterative process to a coarse resolution version of the problem. This coarse version essentially facilitates the propagation of long range information across the domain. This discussion has been added to the paper for clarity (beginning of Section 3.1.2).

8. Eqn 26 contains the strain rates dot_epsilon_1 and dot_epsilon_2. These should be updated to clarify their 2D counterparts.

   We agree, and have added the definitions of the 2D versions of these quantities.