

Response to review #1

Thank you Doug for this helpful review! Original comments are in blue, our responses in black.

Summary

In ‘icepack: a new glacier flow modeling package in Python, version 1.0’, Shapero and co-authors present a promising new ice sheet modeling framework. The framework contains mechanisms for solving both the prognostic mass balance equations for updating ice sheet geometry, as well as diagnostic solvers for approximations to the non-linear Stokes equations. Throughout both the software and the manuscript describing it, the authors focus on ensuring usability (readability), a trait that is bound to make this software (and paper) frequently used. Despite its accessibility, the capabilities of icepack are already impressive, made all the more so by its explicit design prioritization of easy extensibility. I find the manuscript to be exceptionally well-crafted, and I think that it could be published as is. That said, I offer a few suggestions, comments, and points of clarification below.

Minor points

L15-18 It would be nice to have a cited example or three for each of these suggested use cases. This would help the reader identify the kinds of practical problems where icepack might fill a need.

We’ve added one or two references for each.

L74 A low aspect ratio isn’t really an approximation; it’s an existential fact. The approximation that the first order approximation makes is that vertical resistive stresses (or bridging stresses as they are referred to later in this manuscript) are small, pressure is hydrostatic, and bed slopes are small.

Changed to: “The first-order model is based on an asymptotic expansion of the Stokes equations in the ratio of the ice thickness to a typical horizontal length scale. The aspect ratio of most glacier flows is on the order of $1/20$ or less, although there are some exceptions. For example, the main trunk of Jakobshavn Isbrae in Greenland flows through a very deep and narrow trough with an aspect ratio closer to $1/5$. ”

L92 It would be useful to offer a reference regarding an anisotropic fluidity.

Added a reference to Gillet-Chaulet et al. 2006.

L100 Not clear where the Legendre transform enters: the viscous and frictional dissipation can be read off from Eq. 4.

Other reviewers didn't find this digression useful so we've removed it.

Eqs. 6 and 12 While I understand that it is convenient to manipulate the action to reflect the algebraic manipulations to yield the analytical SIA solution, the break in symmetry between Eq. 6 and Eq. 12 is frustrating, given that they both are name 'gravity', and that they should in some sense be the same regardless of which strain rates are assumed to be zero.

These modules were implemented by two different authors, the SSA / hybrid models by myself (DRS) and the SIA model by Jessica Badgeley (second author) as a project for her PhD thesis and to learn more about finite element analysis. We recognize that there's a discrepancy here and this will be corrected in future releases of the package.

L144 That the terminal potential term, Eq. 13 disappears is not obvious. This should probably be clarified, since many readers will be surprised by this.

Expanded the explanation to: "Additionally, the terminal stress term of the action disappears after applying integration by parts to the gravity term to shift the gradient of the surface elevation over onto the velocity."

L152 Cite the method of manufactured solutions.

Added a citation to Roache 2002, Code verification by the method of manufactured solutions.

L159 The benefit to avoiding complicated 3D meshing should not be under- stated, in addition to the reduction in the cost of computational solution.

3D meshing is something to be avoided, but we make a big deal elsewhere in the paper about using extruded meshes. With extruded meshes you only need to do 2D meshing, for which there are algorithms with much better guarantees, but solving 3D problems like the Stokes equations is still possible. We chose not to add any text here as it would then contradict or obscure some of the points we make later.

L188 This is a bit of a red herring, given that impenetrability is a natural boundary condition on the incompressibility equation. It can go right in the action principle, no extra Lagrange multiplier (besides pressure) needed.

I don't think this is entirely correct. Imposing zero velocity at the boundary is easy, but imposing no normal flow or a fixed normal flow together with friction along the opposite directions is much more challenging. When the boundary is flat, you can set one component of the velocity to 0. But if the finite element basis for the velocity has degrees of freedom that are located at the mesh vertices, you have to confront the fact that the unit outward normal

vector isn't uniquely defined at the mesh vertices. In the next sentence we cited a paper from the Elmer/Ice group where they devised an ad-hoc (but still very effective) scheme for defining the normal vector at mesh vertices and thus imposing this boundary condition. You can also do it with Nitsche's method for linear problems but to my knowledge no one has figured that out for power-law fluids. I think that the natural boundary condition you're thinking of in most weak formulations of the Stokes equations is that the average of the pressure is zero.

L367 I'm surprised by this. For challenging geometries, I've always needed to stabilize even when using implicit Euler. Are you sure that implicit Euler is unconditionally stable even for non-linear advection like this?

See Donea and Huerta, *Finite element methods for flow problems*, chapter 3, sections 4-6. This section was also written very early on but became out of date by the time we submitted the paper. Since then, we added an implicit version of the Lax-Wendroff scheme, which has better stability properties and higher order accuracy. The text has been amended to reflect this change.

We did have similar experiences to what you describe when preparing the MISMP+ test case for this paper. The melt phase produces very high ablation rates concentrated in a small region right near the grounded line. In our initial setup, the mesh we used didn't adequately resolve this feature. The finite element interpolation errors can then have an oscillatory pattern that, while not directly amplified by the implicit Euler timestepping scheme as such, still persists and gives a nonsense solution. This behavior manifests even at timesteps substantially below the CFL timestep. High-amplitude oscillatory garbage in the thickness field can then result in unrealistically large driving stresses and crash the nonlinear solver for the velocity. The extra terms from the Lax-Wendroff method help to diffuse out these oscillatory features. Both schemes give good results when the mesh is sufficiently refined; the Lax-Wendroff scheme just requires less refinement. So this could be more an issue of spatial resolution than stability of the timestepping scheme. You're right that saving users from having to think about the CFL condition doesn't completely alleviate all the difficulties and we state as much in the next sentence.

L453 Many advances have occurred in the last 5 years regarding gradient descent due to its necessity for optimizing neural networks. These may yet be useful in this context if you have to deal with a large scale optimization problem where forming the Hessian becomes prohibitive.

One hard criterion we have is that the method needs to naturally map from the dual of the parameter space (where the gradient lives) back to the parameter space itself. Neglecting to do so often results in unspeakable horrors like the vertex degrees of freedom converging at a different rate than the edge degrees of freedom under mesh refinement. See Schwedes et al. 2017, *Mesh Dependence in PDE-Constrained Optimisation*. Using the Hessian or an approximation does that. The acceleration tricks for first-order optimization

methods are really amazing but I've yet to find a nice way to adapt them to problems posed over Sobolev spaces. By contrast, I know that I can scale second-order methods to larger problems than we're solving now by using matrix-free application of the Hessian and coming up with better preconditioners.

L459 Gauss-Newton needs a reference if you're not going to describe it here.

We added a reference to Pratt et al. 1998 which, although focused on applications in seismology, I think does a better job describing it than any other reference.

Eq. 34 It's worth noting that the Schoof law is phenomenological, and was selected because it has the right shape and obeys Iken's bound. As such, if your sliding law obeys Iken's bound and looks right, then it's not any less valid.

This was stated around line 586 but it makes more sense to say that earlier in the text. We've moved the statement accordingly.

L580 I think that having to specify a variational principle for the sliding law is useful because it guarantees a law that is positive (semi-)definite, as it must be to be physical.

I'm not completely sure what you mean. If the basal shear stress has the wrong sign this is arguably just as easy to check by looking at the action functional as it is a nonlinear system of equations. If you mean that the action has to be convex then this isn't true, the sliding law could be rate-weakening, but then there might be multiple steady states.

L701-713 I enjoyed reading this paragraph, but I wonder if stabilizing the Stokes equations is the best illustration of the point, given that icepack does not in fact have Stokes equations implemented (although I imagine it could be done in short order). Maybe stabilizing a transport equation would be a more appropriate case?

This would indeed be a stronger point if we actually had a Stokes solver (it's in the works now) but the example was just to be illustrative. No change to the text.