Summary and general comment, Reviewer 2

This is a very interesting and well-written manuscript that improves my earlier free surface stabilisation (FSSA) algorithm to allow for significantly larger timesteps in glacier and geodynamics codes that have a free surface.
The authors demonstrate that a purely implicit timestepping algorithm doesn't give much advantage compared to an explicit scheme with FSSA if a first-order timestepping scheme is employed. This surprised me somewhat (I had thought otherwise), but is rigorously shown here. In addition, they show that using a second-order timestepping scheme can be made second-order by using a modified FSSA algorithm.
This is initially described for a simple linear viscous setup, which is expanded to a more realistic ice flow simulation in section 7.
Overall, this is a very nice contribution that will be helpful for both the glaciology and geodynamics communities. I think it can be accepted with minor revisions, and I just have a few minor remarks.

Minor remarks:

1. You seem to use Picard fixed-point iterations. If you have a good initial guess, a faster way to converge nonlinear iterations is Newton iterations, which should work particularly well for moderately nonlinear problems such as powerlaw viscous materials with an exponent of n=3. This would potentially also help the convergence if the implicit timesteps, even though it requires deriving the appropriate Jacobian. Is this not commonly done in the glaciology community, or are there other reasons you stick with Picard?

   Reply: For handling the non-Newtonian rheology, Newton solvers are used in ice sheet modelling, typically with some initial Picard iterations. However, the convergence is often poor, and the reason why is to our knowledge not fully understood. In our case we did not use a Newton solver since it is not sufficiently robust in the new vectorised Stokes solver in Elmer/Ice IncompressibleNSVec.f90. For the implicit/coupled iterations we believe Picard iterations are reasonable since only two iterations are needed to reach second order convergence, however it is a very interesting idea to switch to Newton in that context, that we did not think of!

2. eq. 29: Can you give a function or algorithm on how "$octave_?$" is computed? The way this is written is a bit puzzling to me, but this might be because I am not from glaciology...

   Reply: The octaves are cubical splines where the gradients at nodes are randomly generated angles. The approach follows from Perlin 1985 `https://dl.acm.org/doi/pdf/10.1145/325165.325247` *and is described more in detail in Appendix*

A of Löfgren et al 2024 ***https://tc.copernicus.org/articles/18/3453/2024/#bib1.bibx43****. We have clarified the notation in the manuscript and referred to Appendix A in Löfgren et al 2023.*

Minor typos:

1. l. 299: kg `m^{-3}`

   Reply: Corrected.

2. Fig. 8 caption: "stabilisation"

   Reply: Corrected (with American spelling).