The manuscript “Increasing numerical stability of mountain valley glacier simulations: implementation and testing of free-surface stabilization in Elmer/Ice” applies a method developed in the context of mantle-convection modeling to stabilize the evolution of the free surface. The method has been already studied in the context of ice sheet modeling but with simplified problems. I think the paper can be useful to the ice sheet community. However, the presentation should be improved and critical details about the discretization of the equations should be added. In general, I wish the authors were more ambitious with this work, trying to understand better the method from a theoretical point of view, considering higher-order time-discretizations and exploring the effect of different spatial discretizations of the free-surface equation with the coupling method proposed. Below are more detailed comments.

`Eq(7):` This is a pet peeve of mine, but I think that calling the friction coefficient β^2 is really bad notation despite being commonly used in ice sheet modeling. It makes you think that its square root is a meaningful physical quantity. Why not just call it β, or C or μ?

`Section 3.1:` This section contrasts an explicit solution of the Stokes free-surface equation with an implicit solution of such equations. However, there are hidden assumptions in both approaches. As for the explicit solution, it seems that the authors have in mind a low-order, explicit time-integration scheme like Forward Euler. However, other high-order scheme could be used. As an example, one could use the Runge-Kutta scheme, where the Stokes equation would need to be solved multiple time per time step (at each stage of the scheme). Regarding the implicit solution, in addition of limiting the analysis to a low-order scheme like Backward Euler, they also assume that the coupled Stokes free-surface problem would be solved by iterating the solution of Stokes and the free-surface problem until convergence. This implies an outer nonlinear iteration loop (at each iteration the Stokes and free-surface equations are solved) and an inner nonlinear iteration loop for solving the Stokes problem. While this might be the simplest method to implement, it is likely not the most efficient. In fact, one could consider the Stokes and free-surface coupled problem in a monolithic fashion and solve it with a single iterative scheme, linearizing at the same time Stokes and the coupled Stokes free-surface problem. I recommend that the authors better explain their choices and possible alternatives.

`Eq (11):` I would write here eq (10) here as well, with u evaluated at times t^k and all the domains evaluated at time t^k except for the forcing term that is evaluated at time t^{k+1}. This is the scheme you are after, that is, account for the evaluation of the forcing term at the time t^{k+1} to anticipate the effect of the domain change at least on the forcing term.

`Figure 2:` Please mention at least in the caption that these are not the only two time-stepping / coupling options.

`Line 131:` Definition of inner product. Inner product has two arguments. What you wrote and what you used in eq. (10) is just an integral over the domain Ω. If you want to call it inner product then use the notation $(\cdot,\cdot)_\Omega$.

`Lines 150-153:` This part is confusing. I think there are a couple of typos. Eq (13) is referenced instead of equation (12), and "first term on the left-had side" should be "first term on the right-hand side". Further, it is not true that in eq. (12) the domain is assumed to move only due to the velocity of the deformation. Eq. (12) is Reynolds theorem and is valid in general. However, u_b is the velocity of the ice boundary, which is not the velocity of the ice at the boundary. At the surface, $u_b = u + a_s\hat{z}$, that is, the velocity of the surface is the sum of the velocity of the ice u and of the accumulation/ablation rate. Please rephrase this paragraph.

`Eq (13):` typo, check the subscripts.

`Section 3:` Please add details about the discretization of the Stokes equations and the free surface equation. In particular, are you using any stabilization for (11), e.g., upwind, flux limiters, SUPG? How the spatial discretization of (11) and (14) affect the effectiveness of the proposed approach?

`Section 4.2.1:` In addition to the results presented, it would be informative to have results where the same (relatively fine) mesh is used for all the simulation (including the reference one). This would help separating the effect of the spatial discretization from the time discretization, which is the main focus of the paper.

`Appendix A:` Can you detail how the "random generated gradients" are generated?