Reply to Comments by Shangxin Liu

I glimpsed this interesting geodynamic technical paper from the GMD article alert and further read it in detail. This study presents a new finite element modeling framework called HYTEG, which is based on matrix-free geometric multigrid preconditioner (like the one used by the current version of ASPECT) to overcome the need of the large memory for the storage of the stiffness matrix in the classic geometric multigrid preconditioner. The authors show the benchmark results against analytic or previous numerical codes in both 2D and 3D geometries through instantaneous and time-dependent calculations. This new framework provides a useful finite element tool for the community. While the work presented here shows the various capabilities of this new software, I have some comments and suggestions for the authors to consider to further improve the robustness of this study and the HYTEG framework.

1. While the mesh refinement is introduced in the main text, I still found that the number of mesh of each refinement level is not very clear. What are the numbers of the radial elements in each triangulation (refinement) level in each convergence plot for the errors, such as Figs. 1, 2, and 3? For example, in ASPECT, global refinement n means that there is $2^{(n+1)}$ number of radial elements in the 3D spherical shell in the default setting. It's worthwhile to describe the mesh refinement in a clearer way for the convenience of readers of this paper and potential users of the HYTEG framework.

   Authors' reply: Thanks for pointing out this shortcoming. We will make sure to improve the paper in this respect.

2. The mantle response (delta) function benchmark in section 4.1.

   First, the authors should make it clear that how the velocity and pressure errors presented (Fig. 1) are calculated. Are they the errors of the averaged velocity and (dynamic) pressure errors of the whole domain? Which wavelengths (spherical harmonic degree and orders) are presented? These are not quite clear when I read through this part. Although the details may be presented in the earlier studies the authors refer to, I think it's worthwhile to clarify these details in the main text of the paper as well.

   Authors' reply: The error presented is the $\mathcal{L}_2$ norm of the error of the corresponding field over the complete domain. In the delta function cases the surface is considered separately for this. We will clarify this and the spherical harmonic degree and orders in the text.

   Second, only the velocity and pressure solutions are shown. I strongly suggest the authors also calculate the responses of the surface and CMB dynamic topography and geoid. The calculation of the dynamic topography includes the radial derivatives of the velocity, which requires second-order accuracy of the velocity solutions. The geoid solutions are even more sensitive to the computational accuracy because of the counterbalance effect between the buoyancy from density integral and the dynamic topography. If the response functions of the surface and CMB dynamic topography and geoid are also shown to be accurate, the robustness of the Stokes solver for this code can be verified completely.

   Authors' reply: You raise a valid point. Dynamic topography and the geoid are indeed important targets for geophysical inference. In the revised version of the paper, we are planning to show the response functions of surface and CMB topography in the presence of a density anomaly in the dirac delta form of a single spherical harmonic at various depths in the mantle. Also as both of these quantities are computed from the radial stress we are also planning to show the error convergence of the FE computed radial stress to the analytical solution.

   Third, at lines 305-306, several previous efforts in the formulation of the response function benchmark in 3D spherical shell geometry are referenced. However, other previous relevant studies on the same topic by the peers should also be acknowledged as well in this paper. For example, Liu and King, 2019 systematically benchmarked the Stokes solver for the open-source community code ASPECT in 3D spherical shell domain using the similar approach, following the formulation of the earlier work of Zhong et al., 2008 for another popular community code CitcomS. It's noteworthy that Zhong et al., 2008 and Liu and King, 2019 both calculated the response functions not only in isoviscous Stokes system, but also in two-layer viscosity profile with a stiff top lid. I suggest that the authors also calculate the response functions in a two-layer viscosity profile to show that the Stokes solvers of the HYTEG is able to properly handle the radial viscosity jump.

   The calculation of the dynamic topography may also involve the use of consistent-boundary-flux (CBF) method to help improve the accuracy of the stress compared with the straightforward pressure smoothing method (Zhong et al., 1993). Including the effects of self-gravity will also significantly change the long-wavelength dynamic topography and geoid. The incorporation of CBF method and self-gravity into a finite element code will require considerable extra work. If not yet, I don't intend to push the authors to add them into HYTEG for this paper, but it would be necessary to make it clear in the main text that whether CBF method and self-gravity has been included in the calculation of the dynamic topography and geoid solutions. Zhong et al., 2008 and Liu and King, 2019 use CBF method to calculate dynamic topography. The two studies present the response function benchmark for Citcoms and ASPECT for both cases with and without self-gravity.

Liu, S., & King, S. D. (2019). A benchmark study of incompressible Stokes flow in a 3-D spherical shell using ASPECT. Geophysical Journal International, 217(1), 650-667.

Authors' reply: Thanks for pointing us to the CBF method for computing gradients on the boundaries and the consideration. As the HYTEG framework is capable of performing surface integrals in a matrix-free fashion, this is actually straightforward to implement in our code. Hence we have used the same for the computation of the radial stress on the surface and CMB for computing the tpography response functions. In addition, we have made an error convergence comparison between computing the gradients with the CBF method and a simple $\mathcal{L}_2$ projection. We will also add additional and relevant references.

3. Time-dependent thermal convection benchmark in 3D spherical shell.

   It's nice to see a match of the average temperature profiles between HYTEG and our recent study presented in Euen et al., 2023. However, to form a complete evaluation of the performance in 3D spherical shell thermal convection, it's necessary to also show the comparison of the RMS velocity profiles, Nusselt numbers at the top, and Nusselt numbers at the bottom between HYTEG and Euen et al., 2023. The Nusselt numbers require the calculation of the heat flux, i.e., temperature gradient. This diagnostic is a better criterion to evaluate the second-order accuracy of the temperature field. Again, CBF method can improve the accuracy of the heat flux calculation as well (Gresho et al., 1987). Whether CBF method is used in the calculation of heat flux needs to be clarified. The heat flux calculated in the Nusselt numbers of Euen et al., 2023 use the CBF algorithm incorporated into the early version of ASPECT.

   Authors' reply: In the revised version, we will present a comparison of Nusselt numbers and the velocity RMS profiles with the data from Euen et al. (2023). For the heat flux calculation, the CBF method is even easier to implement than for computing pointwise variables, for which one needs to solve a system with the mass matrix. We will clarify this in text also.

4. Lines 270-278. The authors talked about the use of the element-wise viscosity averaging for the matrix-free geometric multigrid method. It would be better to further strengthen the purpose of using the element-wide viscosity averaging for this method. It will especially reduce the memory needed for largely variable viscosity cases compared with the same cases without viscosity averaging. This is similar to the handling of this problem in the current version of ASPECT code (Clevenger and Heister, 2021). In addition, the reason that why harmonic averaging works better than arithmetic averaging in nonlinear rheology needs to be specified.

   Authors' reply: You are correct in pointing out the memory savings when using a P_0 type averaging for the viscosity. Although in practice we see that the major memory overhead is dominated by the velocity/pressure functions (and temporaries) one requires for the Stokes solver. In our work and as you mentioned, has been pointed out in Clevenger and Heister (2021), that the viscosity averaging mainly helps to improve the multigrid solver convergence. Our assumption for why we observed a better performance of harmonic averaging for this nonlinear rheology case could be that the coarse grid approximation is more accurate for our discretization. We will make this clear in text.

Minor comments

1. Line 272, computing the an average -> computing an average

   Authors' reply: We will update this.

2. Euen 2023 -> Euen et al., 2023. This issue appears in some places, such as Figs. 4 and 5.

   Authors' reply: We will update the legends of the figures accordingly.

3. From the equations (5) and (6), it appears that HYTEG solves the Stokes system for dynamic pressure instead of total pressure. I suggest making this point clear in the main text. For example, are the pressure terms in the later response function benchmark in section 4.1 the dynamic pressure or the total pressure?

   Authors' reply: You noted this correctly. We can use different approaches here. We will better explain which one was used for which individual benchmark.