Reply to Review by Shijie Zhong

This paper introduced a finite element modeling framework HyTeG for modeling mantle convection. The framework appears to be very versatile with a lot of capabilities. For example, it is capable of 2-D and 3-D modeling of thermal convection with variable viscosity. It uses triangle/tetrahedral elements with quadratic/linear shape functions for velocity/pressure for numerical stability. It employs a matrix-free solver with multi-grid solver capability that does not require storage of the stiffness matrix, enabling modeling problems with extremely large number of unknowns (10^{11}). It uses an Eulerian-Lagrangian approach (or semi-Lagrangian method?) to solve the energy equation. The paper covers a lot of topics, as this type of papers often do, including governing equations of compressible mantle convection, numerical methods, and some benchmark results. In general, I support technical effort like what this paper presents. I can see that this paper will be eventually published, but there are a few issues I think that the authors should address and improve before it can be published.

First, some main comments:

1. On the benchmark. For the stationary benchmark in section 4.1 (i.e., for the Stokes flow problem), I think that the authors should present the dynamic topography and geoid benchmark results for two important reasons: a) they are geophysically important and relevant, and b) the geoid anomalies are very sensitive to the solution quality for the pressure and velocity. Analytical solutions for the geoid and dynamic topography are widely available. For example, CitcomS benchmarks in Zhong et al., [2008] have a big section on this type of benchmarks (or even in Zhong et al., [2000]). Fig. 1 for the norm-2 for flow velocity and pressure errors is encouraging, but dynamic topography and the geoid benchmarks will be much more relevant, making the code more useful and appealing to potential users.

   Authors' reply: Your suggestion to additionally add dynamic topography and geoid results for the verification of the code is a valid point. In the revised version, we are planning to present the response kernels for dynamic topography at the surface and CMB with respect to density perturbations (Dirac delta form of spherical harmonics) at various depths inside the mantle. As the dynamic topography and geoid are derived from the computed velocity (stress) field, we studied convergence of the radial stress on the boundary. Taking into account other comments, we are planning to add a plot presenting the order of convergence of the error in FE computed radial stress employing the CBF (Consistent Boundary Flux) method as well as an $\mathcal{L}_2$ projection of the gradient.

   On the same topic, in lines 305-306, authors referenced several recent papers (since 2016) on developing semi-analytical solutions for incompressible Stokes flow problem in spherical shell using spectral methods. However, such solutions were developed in geodynamics well before 2016. For example, Tan et al. (2011) showed such analytical solutions for compressible Stokes flow in spherical shell geometry and used them to benchmark compressible version of CitcomS. Zhong et al. (2008, 2000) did the same for incompressible Stokes' flow and used them for benchmarks of incompressible CitcomS. I think that the authors should acknowledge these studies. Actually, the way that the delta function was treated in numerical benchmark calculations presented in 4.1.1 is actually the same as how it was done in CitcomS benchmark in Zhong et al., 2008. Again, some suitable reference is needed for this (see more comments on this type of issue later).

   Authors' reply: As we used approaches that arrive at the semi-analytical solution without the propagator matrix method, these references were unfortunately missed out. It is indeed important to mention it, and we will add these in the paper. Thanks for pointing us towards the compressible benchmark. It would definitely be a nice test for the Stokes solver in the compressible case, which we could perform for our framework, potentially in this or in a future work.

2. On the benchmark result in Figure 2's Nussult numbers vs different resolutions for 2-D compressible mantle convection, I believe that there are some errors in how King et al. (2010)'s results were presented in this Figure. The figure showed the current study's Nu's are slightly less than 7.4 at the highest resolution, but the authors presented a range of possible solutions from King et al. (2010) between 7.3 and 7.7, thus justifying their results. However, the supplementary Table S6 from King et al. (2010) showed that Nu for this case (Di=0.5 and Ra=1e5) ranges from 7.587 to 7.63 for 5 of 6 different codes, including three well-known finite element codes Citcom, Conman and UM that would be very similar to the code in this paper. Only one code in King's benchmark study had Nu at 7.50. In any case, the authors need to clarify the origin of King's benchmark results of Nu from 7.3 to 7.7. From my view, Nu=7.4 for this low Ra number case differs quite significantly from King's benchmark results, suggesting to me a concerning level of numerical error in their solutions.

Authors' reply: The supplementary material from King et al. (2010), unfortunately, is no longer available from the journal's webpages. Hence, we attempted to extract the values from the plot in the paper, which of course is an inexact approach. In the meantime we contacted Scott King directly, and he was so kind as to provide us with the data. We used these to update the paper. As for the argument on the values itself, the treatment of advection is quite different in our code (semi-Lagrangian). This is similar to approach used in the CZ code considered in the King benchmark. CZ was also on the lower side of the reported Nusselt numbers. Re-examining our code we also detected a minor issue in our TALA implementation. You were, thus, correct in your assessment, although even with the issue present we obtained Nusselt numbers close to the range reported. We fixed the problem and also added Nusselt number computation with the CBF method (we will be update the code version on Zenodo). We re-ran the experiment and now obtain Nusselt numbers ($\simeq 7.50 \pm 0.01$). Also velocity RMS values are very close to those for the CZ code from the benchmark.

3. For the four spherical shell convection benchmark cases listed in Table 1 that Euen et al., 2023 used for ASPECT (originally from Zhong et al., 2008), the authors should presented the steady state results of Nu, Vrms, and other quantities, as Zhong et al., (2008) and Euen et al. (2023) did. Currently, the authors only compared the temperature profiles with Euen et al. (2023) in figures. This can be improved by presenting numerical values.

Authors' reply: Thanks for the suggestion. We will follow it and add these values to the updated version.

4. Section 2.3.1 and 2.3.2 discussed treatments of divergence of rho*u and u for compressible convection with no reference. However, the same treatments in finite element codes for compressible mantle convection can be found in Tan et al. 2011 for spherical shell code CitcomS or Leng and Zhong for 2-D Cartesian code. The authors may want to give some references in presenting their methods.

Authors' reply: Thanks for pointing this out. We will add corresponding references for the frozen velocity approach. With respect to the idea of treating the full stress tensor in the momentum equations in a similar way (frozen divergence), we are, to the best of our knowledge, not aware of this having been reported on in the literature so far.

5. The authors highlighted their method as a matrix free method which they stated was the key for solving for a problem with exceedingly large numbers of unknowns. The way I see from reading this paper is that in this method, the elemental stiffness matrix Ke is generated and used to compute Ke*u without the need to assemble elemental Ke to a global stiffness matrix K and without the need to store K of course. If so, I suppose that this will significantly add to the cost of the calculation of Ku, especially if Ke needs to be formed every iteration within a given time step, if you do not store any K or Ke. I can see for constant viscosity and identical element, Ke is probably the same for all the element and only needs to be computed once. However, for variable viscosity, each element may have different Ke which would need to be computed. Therefore, in this matrix-free method, there seems to be a trade-off between computer memory saving and calculation speed. The authors may want to discuss this issue and give an order of magnitude estimate of CPU time for calculations with 10^{11} unknowns.

Authors' reply: It is well-known that compared to standard sparse matrix implementations matrix-free methods are typically more efficient in terms of memory and computational time, especially for medium- to high-order discretizations.

The memory savings are obvious, and crucial for large scale simulations. For the Stokes systems considered here, the number of non-zeros per row is in the hundreds. Adding the indexing data for sparse formats like CSR additionally doubles the memory requirements and would add a memory overhead equivalent to storing that number of additional vectors.

In terms of computational time, the matrix-free operator application is typically faster than a sparse matrix-vector product, because the latter is usually heavily memory bandwidth bound, while the matrix-free operator application tends to be compute bound for medium- to high-order discretizations and complicated bilinear forms (like the viscous terms on curved domains). This fits the performance characteristics of modern hardware much better. Studies of that are ubiquitous in the literature, see, e.g., the series of papers by Kronbichler et al. and also our study for HyTeG in Böhm et al. (2024) in SIAM J. Sci. Comp.

We will extend the discussion of the topic in the introduction accordingly.

Please also note that our problem domain is curved. This means that, even for the constant viscosity case, the local element matrices will not be identical for all elements.

The wall clock time for the 10^{11} DoF case is shown in Figure 9 (right).

Some minor comments:

1. The authors mentioned in a few places the need for mantle convection to achieve 1 km resolution. I was curious how small the time increment would have to be for such a small element, based on the criterion for picking up the time increment. Is it really feasible to use such a small time increment in global models with such a high resolution? Can a different model be formulated if 1 km is indeed needed.

Authors' reply: This is a good point. We will add a comment in the paper on this. Since we use a semi-Lagrangian scheme with an implicit time discretization for the Eulerian part, we end up with a stable scheme even for relatively large timesteps. However, it is correct that we need to complement a high resolution in space with a suitable resolution in time to ensure that the time discretization error does not dominate.

2. The authors may want to double-check the writing. There are quite some grammar issues.

Authors' reply: Thanks for pointing this out. We will check the paper again to improve on spelling and grammar.

3. Line 31, I am not sure if Baumgardner (1985) was for a mantle convection code – its mathematical formulation was very different from any of the mantle convection formulations we know, especially how the continuity equation was treated, as I recalled.

Authors' reply: Baumgardner (1985) is, besides his PhD thesis, the original first reference for the mantle convection code TERRA. Admittedly there have been various changes to the algorithms since that time, but we think it is still a valid citation for that code. Actually, you cited the same reference, as an example for FE methods being 'widely used in the studies of mantle dynamics' in your 2007 chapter in Treatise in Geophysics.

4. For Fig. 1 and 2, can the authors explain what the resolution is for each triangulation level?

Authors' reply: Your comment is correct. The refinement level $\mathcal{T}_k$ alone does not allow to derive the resolution. We will update the paper with the concrete values.