

Response to Reviewer's Comments

Authors appreciate careful reading of the manuscript by the reviewers, and are thankful for helpful suggestions for its improvement. Authors have modified the manuscript substantially in light of the reviewer's comments. All the modifications and changes made are highlighted in the Marked Revised Manuscript. Issues and concerns raised by the reviewers are discussed as follows:

Reviewer #1

I find the reviewer responses to be quite "dismissive" to some of my comments and the comments of the other reviewer. Adding a small paragraph here and there will not cut it. Also, putting the table where computational costs grow tenfold and condition numbers increase by orders of magnitude and NOT including it, and the corresponding discussion, in the body of the paper is not acceptable. This was my exact issue: Is the method justified (beyond being a computational math curiosity)? Your new table suggests that you might as well forego FEM and just do Meshfree.

I do appreciate the effort on additional computations. However, while illuminating, these are raising more questions than providing answers. For the cantilever beam, Cook's membrane, and compression block problems the following needs to be addressed:

Comment 1. *The authors are right. Since gradients of RK functions are not integrated, this makes the case simpler. However, I'm not convinced "lower-order Gauss quadrature works". Please specify precisely the Gauss quadrature used for each case. Make sure results are independent of Gauss quadrature. (You are still integrating a complicated function with a support that does not conform to the mesh...) Keep increasing the number of Gauss points per element to show that the results are not changing.*

Response:

Comment 2. *Why is the cantilever beam problem solved with Quads and*

plate with a hole with Tris only? Please show both element types on both problems. The Quads and Tris seem to behave very differently.

Response:

Comment 3. *Figure 12: Quadratic Quads error is completely insensitive to the number of constraints used. This makes no sense. In addition, as the number of constraints goes to zero, the errors in the strain and pressure do not increase. Why is that? Aren't you losing incompressibility completely in this limit? There must be some "pathology" with this problem, and this may not be a good example.*

Response:

Comment 4. *Figure 15: The Tri3 results suggest that the estimate for n_s is not very good. There is no sharp drop in the strain error around that value at all. It looks like significantly increasing the number of incompressibility constraints does not result in higher errors. Why is that? Shouldn't you be locking at some point? Tri6, however, seems to behave as designed. Very confusing and not substantiated by the analysis or other intuition/experience.*

Response:

The compression block problem really comes from large-deformation formulations. For a linear problem it does not make sense to do four levels of loading because the displacement results will just linearly scale with the applied load. So, just one load level is enough, unless the authors are willing to develop a large-deformation formulation... From the results it's not clear if the formulation is converging because no comparison is made to anything. Perhaps a computation using B-Bar or Taylor-Hood can be shown for comparison?

Response:

Reviewer #2

Thank you for your detailed responses.

While your revisions address all the raised points, several key clarifications remain only in your response letter and are missing from the manuscript itself.

Please add the following directly into the text:

Comment 1. *Table 1: Condition number and efficiency comparison for Cook's membrane problem*

Response:

Comment 2. *In the abstract, it should be mentioned that you test only structured meshes*

Response:

Comment 3. *In the conclusions, it should be mentioned that this method has worse conditioning than mesh-based methods (as many mesh-free methods do)*

Response:

I look forward to your updated manuscript.