

Interactive comment on “GHOST: Geoscientific Hollow Sphere Tessellation” by Cedric Thieulot

C. Hüttig (Referee)

christian.huettig@dlr.de

Received and published: 3 May 2018

GENERAL

The paper presents open-source software that generates radially projected spherical shell grids, used typically in geophysical applications. Although no novel scientific method is presented, the code will serve well as an exercise for everyone starting with numerical simulations in this field.

Unfortunately the source code is only in Fortran, many other modern languages would benefit from such a library as well.

SPECIFIC COMMENTS

As with every algorithm, complexity is key. From your timing plot in Fig 4 it looks like linear complexity $O(n)$, which would be ideal. If so please make a statement.

C1

In your introduction you highlight the difficulty to derive the connectivity between cells. I agree and would like to see a chapter on how you tackle this. Another quite similar topic is avoiding duplicate vertices in recursive building algorithms.

For numerical simulations it is often important that cells stay as geometrically constant as possible to avoid introducing errors based on mesh irregularities. As you have figured out, a good measure for this error is the area variance of the cells within a shell. Unfortunately you plotted this error in a way that makes selecting the “best” method concerning this error impossible. Also, why the volumetric relative error is of importance in a projected scenario (Fig.7) puzzles me.

I do not understand what you do in chapter 3. Comparing analytical solutions to numerical results is always a good idea. But how do you solve U ? What method, order, ... The nature of your problem suggests a spectral method as others would struggle with the asymptotic boundary condition, please elaborate. Also, as the absolute error within a single shell (lateral only, for i.e. the middle shell) for each cell is interesting in this scenario as it also reflects the sensitivity of the numerical method to mesh irregularities.

In chapter 4 the application is quite useful, but you seem to describe an interpolation method with your statement “I have written a simple interface ...”. Please describe in detail how you interpolate the data onto the mesh (generic algorithm based on connectivity or specific algorithms for each grid type? order of interpolation?). Do not use the word simple.

All the best,

Christian Hüttig

Interactive comment on Solid Earth Discuss., <https://doi.org/10.5194/se-2018-7>, 2018.

C2