



EGUsphere, referee comment RC1
<https://doi.org/10.5194/egusphere-2022-685-RC1>, 2022
© Author(s) 2022. This work is distributed under
the Creative Commons Attribution 4.0 License.

Comment on egusphere-2022-685

Eric de Kemp (Referee)

Referee comment on "PySubdiv 1.0: open-source geological modeling and reconstruction by non-manifold subdivision surfaces" by Mohammad Moulaeifard et al., EGU sphere, <https://doi.org/10.5194/egusphere-2022-685-RC1>, 2022

Overall this is a reasonable paper that presents an important contribution using a Python implementation of Subdivision Surface codes (PySubDiv), that are used to demonstrate the benefits of this technology extension. Having the code released in an open-source journal such as GMD is useful to the whole geo-modelling community. The application of Subdivision Surfaces for geology is particularly useful with many geological situations where discontinuities such as faults, unconformities and intrusive bodies are considered. However, There are some major issues with the paper in my opinion. The paper needs to become more distinctive from the Mathematical Geosciences paper from the same team with same lead author;

Mohammad Moulaeifard, Florian Wellmann, Simon Bernard, Miguel de la Varga, David Bommers, Subdivide and Conquer: Adapting Non-Manifold, Subdivision Surfaces to Surface-Based Representation and Reconstruction of Complex Geological Structures Math Geosci, 2022, <https://doi.org/10.1007/s11004-022-10017-x>

This will be a challenge because most of what this current submission to GMD offers has already been presented in Math Geosci, but does have the addition of Python demonstration codes to show the utility of the methods. The challenge is to make this paper a) state more clearly and concisely why we need these tools in the community for geological applications b) while making the code more accessible, and with a bit of effort on c) better user instructions. The paper tends to get into a lot of details without saying why Sub-division surfaces are so important and why our current state of the art (Implicit?) tools is not properly handling these situations. Why do we need to preserve sharp edges in some cases and smooth edges in others? Also importantly, when the whole geological modelling community is moving to implicit surfaces (level set derived) are we going to go back to explicit approaches, even if it means we get some accuracy benefits? So just a better context would be good. This is mostly in the Math Geosc paper but could be restated in this submission in a clear and unique manner that motivates the reader to continue.

I have tried to test all the python codes with limited success. The most crucial point was to do interactive editing and updating of the edge values (CSV) for crease sharpness (sharp -1 to smooth-0). I tested with Windows 10, using Pycharm, Anaconda, (Python v3.9 in virtual environment). It is important to provide working code to the perceived user community, which in my opinion would be dominantly windows users. It would also be helpful if a more detailed tutorial could be provided, perhaps with a bit more details such as in the main authors thesis, to help with editing processes for the graphics. These are currently quite cumbersome and cryptic. Please know that I was helped by two of my colleagues, Michael Hillier and Hossein Jodeiri who both have a pretty good understanding of 3D graphics and meshing issues as well as use of Python codes with various ML supported packages. So collectively we had a good go at trying to make things work putting in several days to troubleshoot and tweek without full success.

If the authors choose to update the useability of their codes, it will widen the user community for these tools and provide better discussions for those wishing to implement their proposed solutions.

The main PySubDiv.py interface code was slightly modified to run in a windows environment for attribute editing and mesh manipulation;

```
# import the necessary packages
import sys
import os
from PySubdiv.create_control_cage import control_cage
from PySubdiv.data import files
from qtpy import QtWidgets
os.environ['QT_API'] = 'pyqt5'

# starting the PyQt application
if __name__ == '__main__':
    app = QtWidgets.QApplication(sys.argv)
    # EdK insert from QT forum
    # https://forum.qt.io/topic/132110/pyqt-program-crashes-without-error-when-run-from-the-command-line
    app = QtWidgets.QApplication.instance()
    if app is None:
        app = QtWidgets.QApplication(sys.argv)
    # end of new insert EdK Nov. 15, 2020
    window = control_cage.PySubdivGUI([])
    window.main()
    sys.exit(app.exec_())
```

Note that despite this there was only limited functionality such as IO of meshes, selection of nodes, edges and partial meshes. There was no deletion of triangles or ways to polygonise a partial surface. Importantly there was no ability to update edge crease values (1-0).

All the SubDivision surface codes performed well but only generated smooth refined meshes or coarse grip frames. Not composite straight and smooth non-manifold object types. This is a big stumbling block at this stage since the users have no way to fully evaluate the utility or advantages of the code without a working application.

There are detailed comments in the attached PDF file and highlights with various colours. Red highlights are of most concern.

Detailed suggestions:

Clarify better distinction between NURB and sub-division surfaces. Could add a graphic with the 3D graphics rendering zoology. Showing spectrum of explicit approaches.

- 2L20, Swarm optimization, this is usually to resolve a minimization or loss function. Also for sparse data extension but how is it used here? Maybe a short line about it if it is important.
- 2 L41, Definitely needs a more in-depth introduction. Clearly indicating the nature of the 2 methods. Needs also some discussion as to where DSI (Explicit and Discrete Smooth Interpolation al la GoCad) and implicit modelling, methods fit into this problem domain. Much has been summed up in Wellmann, F. and Caumon, G2018, but the reader should get a quick summary.

Remember DSI was originally a response to the parametric approach (Manufacturing , Car Industry tools from Paul Bézier not fit for purpose of geology modelling). So, Mallet (DSI) using internal constrains with local and global smoothing approach while dual point co-kriging with 3D WEG emerged for geology applications. (BRGM Geomodeler precursor). Fundamentally we should get some background on why we have difficulty in representing geological structures and manipulating them to be more believable with computer approaches. Simple in the head not so simple in computer graphics and physical world for 3D printing.

- 3 L65 Reword and make clearer.

P.4 L95 - Reword. Bad grammar....Not clear.

P.5 L110 - Just by displaying what could be a relevant figure and some descriptive text

does not guarantee clarity. I have no idea what is going on with this Loop algorithm as articulated.

P.6 L115 - A lot of more recent work has been done on this see :

P.7 L140 - ... between one and zero... indicating level of resistance to the subdivision algorithm with one having no effect and zero having no resistance thus accepting the full recursion and smoothing of the mesh?

This is worded awkwardly as I could interpret your meaning as the opposite parameterization, meaning 0 intuitively means no re-sampling ie. a sharp edge is respected and 1 would be full re-sampling. I think you want to say that 0 is full smoothing and 1 is sharp? Just make it clear so there is no doubt what the end members mean and note intermediate float values can be assigned as well. I think that is the case?

P.8 L150 Fig. 6 For (c) they all look smooth to me? The black lines are not very distinct from the dark blue ones. Try making them red and blue. The internal surface in(c) is not sharp at the CSV = 0.

- 8 L150 - What is a sub-division surface? That is what most geotypes like me will be asking. Most do not know. 1st make it more clear then you can highlight the benefits. Then you can prove your case. Then make sure the code works for your perceived user base. That means windows users as well.

P.8 L160 - But the explanation has to be sufficient to justify using all these terms so that people will not just get confused. Like me.

- 9 L165 - Maybe a simple flow diagram would help because first you need some idea of the geological feature shape, its topology etc. then you generate the control net or grip frame. Then you invert/compute/remesh/estimate for the more accurate geobody.
- 9 L165 - You don't want to use perfume only on special occasions.

P.9 L170 - For geologists it is a bit disconcerting to say we need to 'estimate the control points' for geobody. We usually think of using existing control points as data to estimate the feature. This is in my opinion an expertise domain semantics problem. You will have to find a good solution so both communities a) graphics and b) geomodellers are not getting confused.

P.9 L170 - Not sure these are the appropriate references here since these authors don't go into this kind of detail on the problem I think you are presenting. Maybe add a more detailed ref?

P.9 L190 - 'b' is what? an arbitrary shape parameter, is it calculated or specified by the user?

P.10 L205 - define somewhere what are Min p & z in eq (7,9).

P.10 L215 - Where does Blender come into this? Do you call PySubdiv from inside Blender or just import the resultant surfaces?

- 11 L240 - a sentence more about what swarm optimization is doing beyond increasing 'efficiency' would be helpful.
- 12 L255 Fig. 7 - This is good but a visual of a simple surface from start to finish would also help. Maybe coming in the case study later in the paper?

P.12 L260 - This we call control points or 'stratigraphic and structural constraint points' i.e. picks.

P.13 L275 - Need to explain more details of what this angle is? IS it only the adjacent face normal angle? Maybe in the discussion highlight some issues with this aspect. Fault and horizon intersections for example, even in water tight models, may have erroneous small triangles that can be in similar orientation to an adjacent feature.

P16 L300 Fig. 10b - Try using a gray scale or white to red colour ramp. transparency when the distance rms is very low. normalize the distances between max and min values for better graphic results. Good to have a true distance scale. Shows deeper structures are more uncertain. Very difficult to see what is going on with dark blue regions.

P.18 L355 - Now a bigger issue! When we throw out the grid how do we do property modelling, flow simulation etc. Briefly describe the issues here and why subdivision surface provide a benefit in themselves for geological applications?

...but don't forget that subdivision surfaces are also based on B-Spline and NURB technology....piecewise polynomial basis pairs. See Loop 1987 M.Sc. thesis and Farin G for history.

P.18 L360 - Not all original observation points are respected, that is preserved. Only the mesh is the input.

Older references you may want to consider inserting:

- Habib and J. Warren: Edge and vertex insertion for a class of C1 subdivision surfaces, *Computer Aided Geometric Design* 16(4) (May 1999) p.223-247.
- Karciauskas and J. Peters: Point-augmented biquadratic C1 subdivision surfaces, *Graphical Models*, 77, p.18-26 [1]
- Peters and U. Reif: *Subdivision Surfaces*, Springer series Geometry and Computing monograph 3, 2008.
- Peters and U. Reif: Analysis of generalized B-spline subdivision algorithms, *SIAM J of Numer. Anal.* 32 (2) 1998, p.728-748
- Peters and U. Reif: The simplest subdivision scheme for smoothing polyhedra, *ACM Transactions on Graphics* 16(4) (October 1997) p.420-431,
- Ulrich Reif. 1995. A unified approach to subdivision algorithms near extraordinary vertices. *Computer Aided Geometric Design.* 12(2)153-174
- Jos Stam, "Exact Evaluation of Catmull-Clark Subdivision Surfaces at Arbitrary Parameter Values", *Proceedings of SIGGRAPH'98*. In *Computer Graphics Proceedings*, ACM SIGGRAPH, 1998, 395-404
- Zorin, Denis; Schröder, Peter; Sweldens, Wim (1996). "Interpolating Subdivision for Meshes with Arbitrary Topology". Department of Computer Science, California Institute of Technology, Pasadena, CA 91125.

Please also note the supplement to this comment:

<https://egosphere.copernicus.org/preprints/2022/egosphere-2022-685/egosphere-2022-685-RC1-supplement.pdf>