Message to MGGG meant for meeting on 29th July 2020

Overview

Apologies for not being able to join this meeting in person: I am most probably on the Autobahn going back home from a vacation in the Schwartzwald.

I have been thinking about quite a few issues over the past two weeks and would like to solicit everyone's opinion.

Expanding contiguity and cut edge visualisation to more states

I am officially taking a punt on trying to find a "canonical" dual graph for each state. The problem is too difficult. Instead I will handle islands and discontiguities on a case-by-case basis. Are there best practices for this? Otherwise, I might just make something up that seems reasonable for every state.

Migrating the current server to a more robust solution

The current server is a test server running on a free PythonAnywhere account. In order to expand the visualisation to more states, I need more storage space than is provided in the free account to store the dual graphs and shapefiles. This means paying for a server: either PythonAnywhere or some other cloud provider.

Nick has suggested setting up an organisation account to handle the billing. Does anyone know how would I go about doing this?

I have also been thinking about the time it takes to process a request. Does anyone have any statistics about how many people use Districtr per hour/day? This information will help me decide whether or not to optimise the request made in the contig/cut edge query.

Improving how contiguity and cut edges are calculated

Thomas gave a good suggestion to change the way that contiguity is calculated. The problem is that my contiguity check currently calls a GerryChain function, which in turn calls NetworkX: link to docs here. These functions don't play well with incomplete partition assignments, and hence what I've done is to assign all unassigned district to an imaginary district (-1). But this results in some unwanted behaviour which Thomas pointed out.

I was hoping to make some upstream addition to the GerryChain library to handle incomplete partitions. Can I do this? Is this a good idea? Who should I talk to if so?

Improving how cut edges are shown to the user

Gabe gave me several excellent suggestions to improve the UI/UX of the current display.

One decision point I'd like to raise for consideration is to show

1. the current number of cut edges compared to the minimum and maximum cut edges, or
2. the ratio of cut edges/total edges.

Which one do we like better? The maximum k-cut edge is an NP-hard problem, but there are approximations [1], [2] in polynomial time that are reasonably close enough.

[1]: https://www.math.cmu.edu/~af1p/Texfiles/cuts.pdf

[2]: https://drops.dagstuhl.de/opus/volltexte/2018/8309/pdf/OASIcs-SOSA-2018-13.pdf

I'm also open to any suggestions about how to make the contiguity/ cut edge data more useful and pleasing to users.

Conclusion

Please drop me a line if you have any insight to provide about any of these three things!

[1] 1

[2] 2