Roads and Power Grids

15-48 EC masters' thesis project

For:	Master Students that know how to program - and are interested. If you've ever done a course in heuristics, it is a pre.
Load:	15 - 48 EC, depending on blocks (see below)
Period (approx.):	From September 2021 onward
Supervisor:	Daan van den Berg
Contact:	daan@yamasan.nl, kawarimasen0010@gmail.com
Introduction video:	https://bit.ly/3DWGFrG

About the project

Imagine a map with some dots which need to be connected. Could be towns to be connected by roads, could be power supplies to be connected by cables (Figure 1). The shortest way to connect these points is by a *minimum spanning tree*. Sometimes, adding extra points can result in a shorter total tree. These extra points are called *Steiner points* and the resulting tree is called a *Steiner tree*.

Now imagine hard-to-access areas. Could be swamps, could be mountains, could be nature reserves that don't allow cables or roads. How does this influence the length of the minimum Steiner tree, and the difficulty of finding it? This is what Manou Rosenberg and her colleagues investigated, and the results were published at GECCO in 2021 [1]. In this project, we'll set out to replicate her research, and expand on this knowledge.

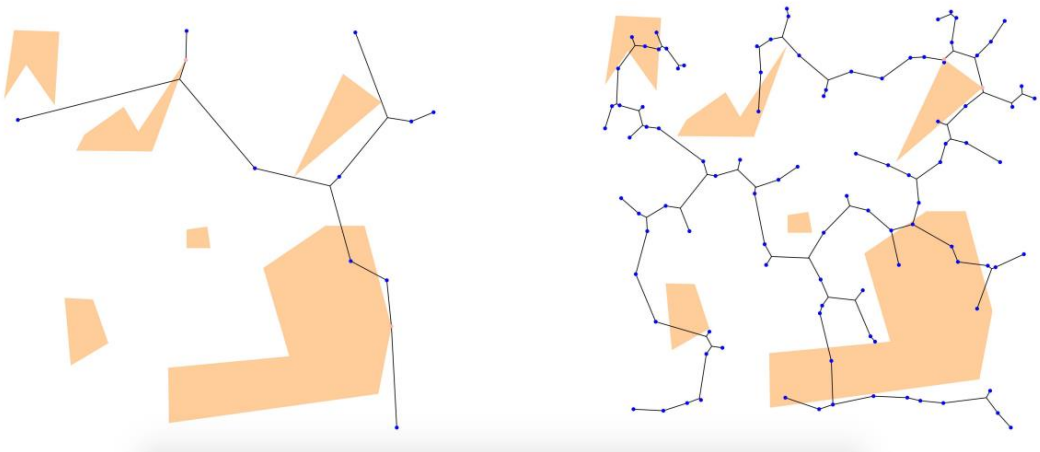

Figure 1 (adapted from Rosenberg et al) Steiner trees are great ways of connecting the dots (by roads or power cables), but how to find the shortest?

Each block is about 15-18 EC's worth. You can choose blocks to your liking, though there is a strong emphasis on doing Block #1 first. Generally, we can adapt blocks as we go along, but once a block is finished, we're not going back, only forward.

Block #1
Start reading Rosenberg's paper, and design a (partial) replication. Depending on many factors (open source code, implementationary hardness etcetera) this block *could* be split up further. Make sure that between approximately 1000 maps, in 10 varieties (perhaps numbers of obstacles, or XY-dimensions) are stored in an open repository.

Block #2
Enrich the experiment by adding more heuristic algorithms, such as PPA and/or simulated annealing. Compare the results.

Block #3
For any of the heuristic algorithms, find good parameterizations. If you choose this block, we'll sit and talk a bit, as we have lots of previous experience and good pointers. It also depends on whether and how you've completed Block #2.

Block #4
Measure the fractality for a number of full scale template-instances for various algorithms. And make comparisons. To what extent this is reasonable needs to be discussed, as it involves generating a lot of data.

Block #5
Check out the spectacular theorem by Frieze. Can we do anything with that? Test it? Compare it?

References

[1] Rosenberg, Manou, et al. "A genetic algorithm approach for the Euclidean Steiner tree problem with soft obstacles." Proceedings of the Genetic and Evolutionary Computation Conference. 2021.