Minimum k-Chinese Postman Problem

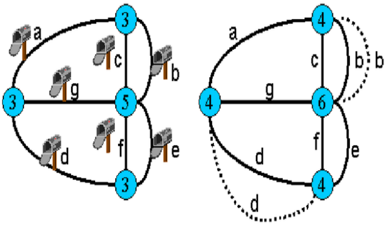
First Tracking

Akif Kartal

Advisor: Doç. Dr. Didem GÖZÜPEK

6 April 2022

Recall Project Definition



- s, initial vertex
- k, given positive number
- *l(e)*, length for each edge
- n, number of vertices(nodes)

Given a multigraph $G = (V, E)$ initial vertex $s \in V$ length $l(e) \in N$ for each $e \in E$ the *minimum k-Chinese postman problem* is to find k tours such that each edge of the graph has been traversed at least once and the most expensive tour is minimized.[1]

Algorithm to be implement

In order to solve this problem, I will implement heuristic augment-merge algorithm.

The idea of the algorithm is roughly as follows. We start with a closed walk C_e for each edge $e = v_i, v_j \in E$, which consists of the edges on the shortest path between the depot node v_1 and v_i, the edge e itself, and the edges on the shortest path between v_j and v_1, i.e. $C_e = (SP(v_1, v_i), e, SP(v_j, v_1))$. Then we successively merge two closed walks trying to keep the tour weights low and balanced until we arrive at k tours. [2]

Algorithm Steps

1. Sort the edges e in decreasing order according to their weight $w(C_e)$.

2. In decreasing order according to $w(C_e)$, for each $e = v_i, v_j \in E$, create the closed walk $C_e = (SP(v_1, v_i), e, SP(v_j, v_1))$, if e is not already covered by an existing tour. Let $C = (C_1, \ldots ,C_m)$ be the resulting set of tours. Note that the tours are sorted according to their length, i.e. $w(C_1) \geq w(C_2) \geq \ldots \geq w(C_m)$. If $m \geq k$ we are done and have computed an optimal k -postman tour. If $m < k$ we add $k - m$ “dummy” tours to C, each consisting of twice the cheapest edge incident to the depot node.

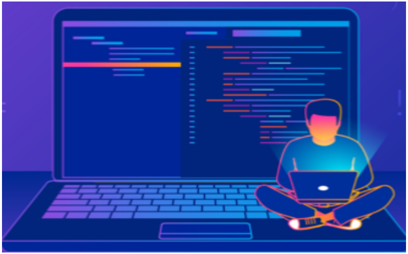
3. While $|C| > k$ we merge tour C_{k+1} with a tour from $C_1, \ldots, C_k$ such that the weight of the merged tour is minimized.

Starting to implement

- I have started to implement project in python. First we need to create an abstract representation of a graph such as edges will be represented as an object like this.

  edge(start node, end node, length)

- After finishing coding this step, I will start to implement my algorithm.



Project Timeline

1^{st} Meeting	Making literature research. Understanding the problem.
2^{nd} Meeting	**Continue literature research.** **Determine the steps of the algorithm.** **Start to implement project**.
3^{rd} Meeting	Finish algorithm implementation. Test with different parameters. Show the results on charts.
4^{th} Meeting	Creating Different Random Graphs. Preparing a Graphical User Interface. Comparing results with the literature results.

Success Criteria

1. Heuristic algorithm complexity will be better than $\mathcal{O}(|E|^4)$
2. Creating at least 30 different graphs.
3. Getting algorithm results in less than 5 seconds with small parameters which means when $n < 25$ and $k < 5$.

References

[1] A. Hölscher, *A cycle-trade heuristic for the weighted k-chinese postman problem*, 2018.

[2] D. Ahr and G. Reinelt, *New heuristics and lower bounds for the min-max k-chinese postman problem*, 2002.

[3] D. Ahr, *A tabu search algorithm for the min–max k-chinese postman problem*, 2005.

[4] S. Liu, *A genetic algorithm for min-max k-chinese postman problem with applications to bridge inspection*, 2019.

Thank You