Combinatorial optimization
Exercise sheet 7

Solutions by: Anjana E Jeevanand and David Čadež

25. November 2023

Exercise 7.1. Hint already gives us the graph, we just have to prove it satisfies the requirements. So let (K_n, c) be a graph with weights $c(\{i,j\}) = \lambda_{i,j}$ (we use $\lambda_{i,j} = \lambda_{j,i}\ \forall i,j \in K_n$ for this to be well-defined) and T be a maximum weight spanning tree in (K_n, c). Lets show local edge-connectivities in T are exactly $\lambda_{i,j}$.

Since T is a tree, local edge-connectivity for any pair of vertices is the minimum of weights of edges on the path between them.

Take $i, j \in T$.

Condition $\lambda_{i,k} \geq \min\{\lambda_{i,j}\lambda_{j,k}\}$ clearly implies $\lambda_{i,k} \geq \min_{e \in P_{i,k}} \lambda_e$, where $P_{i,k}$ is the edge set of the path between i and k. This already proves that local edge-connectivity for a pair i, j is smaller or equal to $\lambda_{i,j}$.

Now suppose the inequality would be strict. Let $\{k,l\} \in P_{i,j}$ be an edge on the path between i and j with $\lambda_{k,l} < \lambda_{i,j}$. Then we could simply replace $\{k,l\} \in T$ with $\{i,j\}$ and obtain a tree with strictly bigger weight, which contradicts our assumption that T is maximum weight.

Therefore local edge-connectivity is exactly $\lambda_{i,j}$ for every $i, j \in T$.