INTRODUCTION TO APX - HW1

TSP and friends

Every task is worth two points. Deadline: **10. 11. 2015 17:19.** Solutions can be sent via email or handed to me in person.

Exercise one

1. Find a class of graphs showing that the algorithm for metric TSP that uses the minimum spanning tree tour is no better than a 2-approximation.
2. Find a class of graphs showing that Christofides' algorithm for metric TSP is no better than a 3/2-approximation.

In both cases we look for an infinite class of graphs which has a strictly increasing number of vertices, i.e. we want $\{G_i | i \in \mathbb{N}\}$ so that $\forall i \in \mathbb{N}\colon |V(G_{i+1})| > |V(G_i)|$. That is a reasonable request; after all, if the tight bound would hold only for graphs with 20 vertices or less, and the algorithm would be 1.25-approximation for larger graphs, we would say that the algorithm is *asymptotically* a 1.25-approximation.

In this example we do not require tight bounds for small graphs, which means you can for instance prove that Christofides' algorithm on a graph G_i is no better than a $(3/2 - x_i)$-approximation, where $x_i \to 0$. In other words, if your example will „clearly show" that in the limit the bound is 3/2, you are done.

Exercise two

Consider the following algorithm for asymmetric TSP on a graph $\vec{G}$ with a given distance function $d\colon \vec{E} \to \mathbb{R}^+$:

1. We find a directed circuit $\vec{C}$ in $\vec{G}$ which minimizes $\frac{\sum_{\vec{e} \in \vec{C}} d(\vec{e})}{|\vec{C}|}$.
2. We add all the edges $\vec{E}(\vec{C})$ to the solution.
3. We remove all vertices of $\vec{C}$ except one. We continue recursively until $\vec{G}$ is only a single vertex.

Your task is:

- Explain how we can achieve point 1 in polynomial time.
- Prove that the previous algorithm is an $\mathcal{O}(\log n)$-approximation for asymmetric TSP.

Exercise three

In the *Steiner tree problem* we get on input a connected undirected graph $G = (V, E)$, an edge cost function $c : E \to \mathbb{R}^+$, and finally a list of *terminals* $S \subseteq V$. A feasible solution to our problem is any subset of edges $E' \subseteq E$ so that the graph $G' = (V, E')$ has all the terminals in one connected component. We aim to minimize the cost, i.e. $\sum_{e \in E'} c(e)$. Your task is to design a 2-approximation algorithm.

Hint: The graph does not need to satisfy the triangle inequality. First, think about the case when it does (it should be easy then). To solve the general case, try to use some of the techniques from the TSP approximation.

Exercise four

Consider a cubic 2-edge-connected graph G. The word *cubic* means that every degree of the graph is equal to 3. The word *2-edge-connected* means that the graph does not contain a bridge, which is an edge whose removal disconnects the graph. The graph is not weighted, so all the edges have distance one.

1. Show that any such graph has a TSP tour of length at most $4|E|/3$.
2. Prove that the point $(1/3, 1/3, 1/3, \ldots, 1/3)$ lies always in the perfect matching polytope of G.
3. Prove that for G there exists a set of perfect matchings $M_1, \ldots M_k$ of G and a corresponding set of constants $\lambda_1 \geq 0, \lambda_2 \geq 0, \lambda_3 \geq 0, \ldots, \lambda_k \geq 0, \sum_{i=1}^{k} \lambda_i = 1$ having the following property: if we take any perfect matching M_i randomly with probability λ_i, then for *every edge e in the*

whole $E(G)$ it holds that $P[e \in M_i] = 1/3$.

The perfect matching polytope is this one:

$$\begin{aligned}
\forall v \in V\colon && \sum_{e=vx} x_e &= 1\\
\forall S \subsetneq V, S \neq \emptyset, |S| \text{ odd}\colon && \sum_{e \in E(S, V\setminus S)} x_e &\geq 1\\
\forall e \in E\colon && x_e &\geq 0
\end{aligned}$$