Approximation Algorithms

Exercise Sheet 10

Algorithm 1: Steiner Tree Approximation

Input: $G = (V, E)$, $R \subseteq V$, $c: E \to \mathbb{R}_{>0}$
Output: a Steiner tree $T$

$I := \emptyset$

repeat

| foreach $v \in V \setminus (R \cup I)$ do |
| $I' := I \cup \{v\}$ |
| remove vertices from $I'$ having degree 1 or 2 in $\text{MST}(R \cup I')$ |
| if $\text{mst}(R \cup I') < \text{mst}(R \cup I)$ then |
| $I := I'$ |

until no improvement during last iteration

return a Steiner tree corresponding to the MST computed in the last iteration

Exercise 10.1:
Consider the Steiner Tree Problem restricted to instances for which $V \setminus R$ forms a stable set. Give a tight analysis of the approximation algorithm for this problem described by Algorithm 1 i.e., determine the number $\alpha \geq 1$ such that

(i) Algorithm 1 is an $\alpha$-approximation for this restriction of the Steiner Tree Problem and

(ii) there is a class of instances on which the algorithm does not necessarily perform better than an $\alpha$-approximation.

Here, we denote by $\text{MST}(R)$ a minimum spanning tree in the terminal distance graph for the vertex set $R$ and by $\text{mst}(R)$ its length.

(4 points)

Exercise 10.2:
Consider the relative greedy algorithm.

(i) For every $k \in \mathbb{N}$ describe an instance for which the relative greedy algorithm does not necessarily find an optimum solution.

(ii) Can you improve the approximation guarantee of the algorithm for $k = 5$?

(2 + 1 points)
Exercise 10.3:
Consider the restriction of the Traveling Salesman Problem to complete graphs in which all edge lengths are either 1 or 2. Give a $\frac{4}{3}$-approximation algorithm for this problem.

*Hint:* Compute a minimum-weight perfect 2-matching. (3 points)

Exercise 10.4:
Consider the following algorithm for the Symmetric Traveling Salesman Problem with triangle inequality:

Consider an instance $(K_n, c)$. Start with an arbitrary node $u \in V(K_n)$. Find a shortest edge $e = \{u, v\} \in E(K_n)$ connecting $u$ to another node $v$. This yields a subtour $T = (u, v, u)$. Let $U := V \setminus \{u, v\}$. Repeat the following steps until $U = \emptyset$:

(i) Find $w \in U$ with shortest distance to one of the nodes in $T$.

(ii) Add $w$ to $T$ between two neighboring nodes $i, j \in T$ (by deleting edge $\{i, j\}$ and connecting $i$ and $j$ with $w$) such that the cost of the new tour is minimized, i.e., find neighbouring $i, j \in T$ such that that $d(i, w) + d(w, j) - d(i, j)$ is minimum. Remove $w$ from $U$ afterwards.

Show that this is a 2-approximation algorithm. (3 points)

Exercise 10.5:
Show that the following problem is NP-complete: Given a graph $G$ and a Hamiltonian circuit $C \subseteq E(G)$, is there another Hamiltonian circuit $C' \subseteq E(G)$?

*Hint:* Use the gadget below. (3 points)

Please turn in your solutions on Tuesday, June 30th, before the lecture.