Exercise Set 9

Exercise 9.1. Let $(U, \mathcal{S}, c)$ be an instance of Minimum Weight Set Cover and $p := \max_{S\in\mathcal{S}} |S|$. Consider the following algorithm:

1. For $u \in U$, define $c_u := \min\{c(S) : u \in S \in \mathcal{S}\}$.
2. Let $\mathcal{R} := \emptyset$ and let $W := U$ be the set of elements uncovered by $\mathcal{R}$.
3. While $\mathcal{R}$ is not a feasible solution, do the following:
   - Choose a set $S \in \mathcal{S}$ that minimizes $\frac{c(S)}{\sum_{u\in(S\cap W)} c_u}$.
   - Add S to $\mathcal{R}$.
   - Replace W by $W \setminus S$.
4. Return $\mathcal{R}$.

(a) Prove that in every iteration of the above algorithm, we have

$$\frac{c(S)}{\sum_{u\in(S\cap W)} c_u} \leq \min\{1, \frac{\mathrm{OPT}}{\sum_{u\in W} c_u}\},$$

where OPT denotes the value of an optimum solution.

(b) Prove that the above algorithm is a $(1 + \ln(p))$-approximation for Minimum Weight Set Cover.

(2+3 points)

Exercise 9.2. Let $(U, \mathcal{S}, c)$ be an instance of Minimum Weight Set Cover and $p := \max_{S\in\mathcal{S}} |S|$. Assume w.l.o.g. that for every set $S \in \mathcal{S}$ also all its subsets $R \subseteq S$ are present in $\mathcal{S}$ and we have $c(R) \leq c(S)$. Consider the following algorithm:

1. Let $\mathcal{R}$ be an arbitrary set cover solution, where w.l.o.g. $\mathcal{R}$ is a partition of U.
2. Do the following as long as it decreases the potential function $\Phi(\mathcal{R}) := \sum_{R\in\mathcal{R}} H_{|R|} \cdot c(R)$ (where for $n \in \mathbb{N}$, H_n denotes the n-th harmonic number):
   - For each set $R \in \mathcal{R}$, define $\bar{c}(u) := \frac{1}{|R|} \cdot c(R)$ for all $u \in R$.
   - Choose $S \in \mathcal{S}$ that minimizes $H_{|S|} \cdot c(S) - \sum_{u\in S} \bar{c}(u)$.
   - Replace every set $R \in \mathcal{R}$ by $R \setminus S$.
   - Add S to $\mathcal{R}$.
3. Return $\mathcal{R}$.

(a) Prove that the above algorithm terminates in finite time.

(b) Prove that the output $\mathcal{R}$ of the above algorithm satisfies $c(\mathcal{R}) \leq H_p \cdot \mathrm{OPT}$, where OPT denotes the value of an optimum solution.

(c) How can the above algorithm be modified to run in polynomial time while loosing only an arbitrarily small error of $\varepsilon > 0$ in the approximation ratio?

(1+4+2 points)

Exercise 9.3. Consider an undirected graph $G = (V, E)$ with non-negative edge weights $c : E \to \mathbb{R}_+$ and terminal set $T \subseteq V$. The Bottleneck Steiner Tree Problem asks for a Steiner tree S for T in G whose bottleneck weight $\max_{e \in E(S)} c(e)$ is minimum. Show that Bottleneck Steiner Tree can be solved in polynomial time.

(3 points)

Exercise 9.4. Let (G, c, T) be an instance of Steiner Tree. Fix $k \in \mathbb{N}$. Let $\mathcal{C}^k$ be the set of pairs (t, R) for all $t \in R \subseteq T$ with $|R| \leq k$ (called *directed components*). For $C = (t, R) \in \mathcal{C}^k$, let $c(C)$ denote the minimum cost of a Steiner tree for R in (G, c). Fix an arbitrary terminal $r \in T$ as root. For $r \in U \subset T$, we denote by $\delta^+_{\mathcal{C}^k}(U)$ the set of all $(t, R) \in \mathcal{C}^k$ with $t \in U$ and $R \setminus U \neq \emptyset$. The so-called *directed component LP* then reads as follows:

$$\begin{aligned} \min \quad & \sum_{C \in \mathcal{C}^k} c(C) x_C \\ & \sum_{C \in \delta^+_{\mathcal{C}^k}(U)} x_C \geq 1 \quad \forall r \in U \subset T \\ & x_C \geq 0 \quad \forall C \in \mathcal{C}^k \ . \end{aligned}$$

Prove that for any fixed $k \in \mathbb{N}$, the directed component LP for the problem of finding a k-restricted Steiner tree can be solved in polynomial time.

(5 points)

Deadline: Tuesday, June 18$^{\text{th}}$, before the lecture. The websites for lecture and exercises can be found at:

`http://www.or.uni-bonn.de/lectures/ss24/appr_ss24_ex.html`

In case of any questions feel free to contact me at blauth@or.uni-bonn.de.