University of Freiburg
Dept. of Computer Science
Prof. Dr. F. Kuhn
P. Bamberger, P. Schneider

Algorithms Theory
Exercise Sheet 12

Due: Tuesday, 2nd of February 2021, 4 pm

Exercise 1: Load Balancing *(10 Points)*

Consider the load balancing problem as introduced in the lecture with m machines $M_1, \dots, M_m$ and n jobs with processing times $t_1, \dots, t_n > 0$. Let $L := \sum_{i=1}^{n} t_i$ be the total processing time.

(a) Let $m = 2$. Show how to compute a scheduling whose makespan is a $(1+\varepsilon)$-approximation of that of an optimal scheduling in time $O(n^3/\varepsilon)$. *(3 Points)*

Hint: Try utilize the PTAS for the Knapsack problem.

(b) Let m be arbitrary. Assume for now that all processing times t_i are integer. Show that the load balancing problem can be solved *exactly* in time $O(nmL^m)$. *(4 Points)*

Hint: You can use Dynamic Programming.

(c) Use the algorithm from part b) to develop a PTAS for the load balancing problem in case the number of machines is a constant (i.e., $m = O(1)$). More specifically, show that one can compute a $(1+\varepsilon)$-approximation in time $O\left(\left(\frac{n}{\varepsilon}\right)^c\right)$ for some constant c (c can depend on m). *(3 Points)*

Hint: You can use the same scaling and rounding technique from the PTAS for Knapsack.

Exercise 2: Minimum Set Cover with Bounded Frequency *(10 Points)*

Consider a set cover instance with a set of elements X and a family of subsets $\mathcal{S} \subseteq 2^X$ of X. A set cover instance $(x, \mathcal{S})$ is said to have *frequency* f iff every element $x \in X$ is contained in at most f of the sets $S \in \mathcal{S}$.

(a) First consider the unweighted version of the set cover problem. Give a polynomial-time algorithm with approximation ratio at most f for the minimum set cover problem with frequency f. *(3 Points)*

Hint: The special case where $f = 2$ is equivalent to the minimum vertex cover problem.

(b) Now we consider the weighted set cover problem, where every set $S \in \mathcal{S}$ has a weight $w_S > 0$. We first consider an assignments of prices $p_x \geq 0$ to all elements $x \in X$. We call a price assignment *valid* if for every set $S \in \mathcal{S}$: $\sum_{x \in S} p_x \leq w_S$. Show that for any valid price assignment $p_x \geq 0$ for all $x \in X$, it holds that

$$\sum_{x \in X} p_x \leq \sum_{S \in \mathcal{C}} w_S \quad \text{for every set cover } \mathcal{C}.$$ *(2 points)*

(c) We say that a valid price assignment $p_x \geq 0$ for $x \in X$ is *maximal* if it is not possible to increase some price p_x without violating the validity condition of the price assignment. A maximal price assignment can be computed by a simple greedy algorithm. Show that a maximal price assignment can be used to derive a set cover algorithm with approximation ratio at most f. *(5 Points)*

Hint: Come up with a family of sets $\mathcal{C} \subseteq \mathcal{S}$ based on a maximal price assignment and show that it covers X. Then use the property from part b) to show the approximation ratio.