Universität Bremen **FB3: Mathematik/Informatik**

Dr. Martin Böhm
Dr. Ruben Hoeksma
Prof. Dr. Nicole Megow

Summer 2019

Approximation Algorithms

Homework Sheet 1 (Deadline 23. 04. 2019 12:00 – before the lecture starts.)

All exercises must be done individually. Feel free (and encouraged) to discuss among each otherut each solution must be written independently. Two or more submissions found with exactly the same solution on any of the exercises will be awarded no points for the entire exercise sheet. The same holds in case of directly copying from any other source.

Exercise 1 (*10 points*). Given an undirected graph $G = (V, E)$, a *cut* in G is a subset $S \subset V$. For $S, S' \subset V$ with $S \cap S' = \emptyset$ we denote by $\delta(S, S') \subset E$ the set of edges with one endpoint in S and one endpoint in S' and we call $|\delta(S, V \setminus S)|$ the *size* of the cut S. Max Cut is the following problem:

Input: an undirected graph $G = (V, E)$
Task: Find a cut S in G of maximum size $|\delta(S, V \setminus S)|$.

Consider the following greedy algorithm that partitions the set of vertices into two disjoint sets S and S'.

Greedy Algorithm for Max Cut

---

(1) Fix an ordering on the vertices $v_1, v_2, \dots v_n$.

(2) Let $S = \{v_1\}$ and $S' = \emptyset$.

(3) For $i = 2, \dots, n$, add v_i to one of the sets S or S' such that $|\delta(S, S')|$ is maximized.

(4) Return S.

---

a) For an iteration i of the algorithm, let r_i denote the number of edges that are incident to v_i and some previously added $v_{i'}$ with $i' < i$. Show that

$$\sum_{i \in [n]} r_i = |E|.$$

b) Use (a) to prove that the greedy algorithm is a 1/2-approximation algorithm for Max Cut.

Exercise 2 (*10 points*) Consider the knapsack problem which is defined as follows.

Input: n items, each having associated a value $v_i \geq 0$, a weight $w_i \geq 0$ and a knapsack capacity $b \geq 0$, where $w_i \leq b$ for all $i \in [n]$.
Task: Find a subset of items $S \subseteq [n]$ with $\sum_{i \in S} w_i \leq b$ of maximum total value, i.e., that maximizes $\sum_{i \in S} v_i$.

Consider an instance of the knapsack problem. Let z^* denote the value of an optimal solution and z_{greedy} the value of a knapsack obtained through the following algorithm:

Algorithm for the knapsack problem
(1) Sort and reindex the items such that $\frac{v_1}{w_1} \geq \cdots \geq \frac{v_n}{w_n}$.
(2) Determine $k' := \max\{k \in [n] \, : \, \sum_{i=1}^{k} w_i \leq b\}$.
(3) If $\sum_{i=1}^{k'} v_i > v_{k'+1}$, then return items $S = \{1, \ldots, k'\}$ else return the item $k' + 1$.

a) Show that the algorithm is a 1/2-approximation for the knapsack problem.

b) Prove that the constant 1/2 is best possible for this algorithm, i.e., there is no constant $\alpha > 1/2$ such that $z_{\text{greedy}} \geq \alpha \cdot z^*$. for all possible instances of the knapsack problem.

c) Show that the last step of the algorithm is essential, i.e., prove that by always returning the knapsack solution $S := \{1, \ldots, k'\}$ the ratio $\frac{z_{\text{greedy}}}{z^*}$ can become arbitrarily small.