



Algorithm Theory Exercise Sheet 13

Due: Friday, 6st of February, 2026, 10:00 am

Exercise 1: The Densest Subgraph (10 Points)

Let $G = (V, E)$ be a graph, $S \subseteq V$, and $E(S) := \{\{u, v\} \in E \mid u, v \in S\}$. We define the density of $S \neq \emptyset$ to be $den(S) := \frac{|E(S)|}{|S|}$. In the *densest subgraph problem*, the goal is to find a subset $S^* \subseteq V$ that maximizes the $den(S)$, i.e., $den(S^*) := \max_{\emptyset \neq S \subseteq V} den(S)$. In this exercise, we will study a greedy algorithm that gives a $\frac{1}{2}$ -approximation to the problem.

Algorithm 1 Greedy Densest Subgraph \triangleright input graph $G = (V, E)$

- 1: Let $S := V$
 - 2: **for** $i = n, \dots, 1$ **do**
 - 3: $S_i := S$
 - 4: Find $v \in S$ with minimum degree in $G[S]$.
 - 5: Delete v from S . \triangleright i.e., $S := S \setminus \{v\}$
 - 6: Compute index $a = \arg \max_{1 \leq i \leq n} \{den(S_i)\}$ \triangleright find best set of S_1, \dots, S_n
 - 7: **return** S_a
-

Note that in every iteration exactly one node get deleted from S . We will therefore define $\overline{deg}(v)$ as the degree of v when it gets deleted from S , that is, when it has minimum degree in the remaining graph $G[S]$.

- (a) Show that for a node $v \in V$ that get deleted in iteration i , it holds that (4 Points)

$$\overline{deg}(v) \leq 2 \cdot den(S_i)$$

- (b) Let S^* be an optimal solution to the problem. Show the following two inequalities:

$$\sum_{u \in S^*} \frac{deg_{G[S^*]}(u)}{2} \leq \sum_{u \in S^*} \overline{deg}(u) \leq 2|S^*| \cdot den(S_a)$$

Here $deg_{G[S^*]}$ denotes the degree of node v in the graph induced by S^* . S_a is the set returned by the algorithm (i.e., the set of maximal density among S_1, S_2, \dots, S_n). (5 Points)

- (c) Conclude that the algorithm computes a $\frac{1}{2}$ -approximation to the problem. (1 Point)

Exercise 2: Partitioning into 3 large cuts (10 Points)

You are given the following optimization problem:

- **Input:** A graph $G = (V, E)$,
- **Solution:** A partition of the set of the nodes into 3 sets, i.e., $V = V_1 \dot{\cup} V_2 \dot{\cup} V_3$,
- **Goal:** Maximize $|\{\{v_i, v_j\} \in E \mid v_i \in V_s, v_j \in V_t, s \neq t\}|$.

Design a $3/2$ -approximation algorithm. Prove that your algorithm terminates, is correct and gives the $3/2$ approximation factor. Your algorithms runtime should be polynomial in the number of nodes (no proof needed if obvious).