Exercises for Approximation Algorithms 1

Inge Li Gørtz

Minimum Maximal Matching

Design a 2-approximation algorithm for finding a minimum cardinality maximal matching in an undirected graph.

Hint Use the fact that any maximal matching is at least half the maximum maximal matching.

Acyclic Graph

Given a directed graph $G = (V, E)$, pick a maximum cardinality set of edges from E such that the resulting graph is acyclic. Give a 1/2-approximation algorithm for this problem.

Hint Arbitrarily number the vertices and pick the bigger of the two sets, the forward going edges and the backward going edges.

The k-supplier problem

The *k-supplier* problem is similar to the k-center problem, but the vertices are partitioned into *suppliers* $F \subseteq V$ and *customers* $C \subseteq V$. The goal is to find k suppliers such that the maximum distance from a customer to a supplier is minimized. Give a 3-approximation algorithm for the k-suppliers problem.

R. Bar-Yehuda

Consider the following 2-approximation algorithm for the (cardinality) vertex cover problem. Find a depth first seach tree in G and output the set S of all nonleaf vertices of this tree. Show that S is a vertex cover for G and $|S| \leq 2 \cdot \mathrm{OPT}$. Give a tight example showing that this bound is tight.

Hint Show that G has a matching of size $|S|/2$.

Priority k-center

Consider the following variant of the k-center problem, where the vertices have priorities: Each vertex have a weight, and we want to find a set of k centers so that the maximum *prioritized* distance of a vertex to its closest center is minimized.

Formally, in the *prioritized k-center problem* we are given a complete graph $G = (V, E)$ with a cost function on the edges $d : E \to Q^+$ satisfying the triangle inequality, a weight function on vertices: $w : V \to \mathbb{R}^+$, and a positive integer k. The problem is to find a set of centers $C \subseteq V$ with $|C| \leq k$ minimizing

$$r(C) = \max_{v \in V} w(v) \cdot d(v, C),$$

where

$$d(v, C) = \min_{u \in C} d(v, u).$$

Algorithm for prioritized k-center The following algorithm for the prioritized k-center problem assumes we know the optimal radius r.

Algorithm 1 kCenter (G, r)

```
Set S = V and C = ∅.
while S ≠ ∅ do
    Select the heaviest vertex v ∈ S (the vertex with highest weight)
    Set C = C ∪ {v}
    Remove all vertices u from S with w(u) · d(u, v) ≤ 2r from v.
end while
Return C
```

Assignment 1 Assume we know the optimum covering radius r. Let C be the set of centers computed by the algorithm $kCenter(G, r)$, let C^* be the set of optimal clusters (each vertex is assigned to its closest center). Consider an iteration of the algorithm and let v be the vertex chosen in this iteration.

(a) Let c^* be the cluster v is contained in in the optimal solution. Let z be a vertex assigned to c^* in the optimal solution and that is in S in the iteration when v is chosen. Show that $w(z) \cdot d(z, v) \leq 2r$.

(b) Show that at most one vertex from each cluster from C^* belongs to C.

Assignment 2 Prove that Algorithm 1 is a 2-approximation algorithm for the prioritized k-center problem (assuming that we know the optimal covering radius r).

Mandatory Exercise

Let $G = (V, E)$ be a complete undirected graph with edge costs satisfying the triangle inequality, and let k be a positive integer. The problem is to partition V into sets $V_1, \ldots, V_k$ so as to minimize the costliest edge between two vertices in the same set, i.e., minimize

$$\max_{1 \leq i \leq k,\ u,v, \in V_i} c(u, v).$$

Give a 2-approximation algorithm for this problem.