Spring 2018, CS 583: Approximation Algorithms

Homework 4

Due: 03/30/2018

Instructions and Policy: Each student should write up their own solutions independently. You need to indicate the names of the people you discussed a problem with; ideally you should discuss with no more than two other people. You may be able to find solutions to the problems in various papers and books but it would defeat the purpose of learning if copy them. You should cite all sources that you use.

Read through all the problems and think about them and how they relate to what we covered in the lectures. Solve as many problems as you can. Please submit your solutions to at least 4 problems.

Please write clearly and concisely - clarity and brevity will be rewarded. Refer to known facts as necessary. Your job is to convince me that you know the solution, as quickly as possible.

Problem 1 Metric-TSP-Path problem is the following. Given a metric (V, d) and two nodes s, t the goal is to find a minimum cost s-t spanning path (that is, a path that contains all the nodes). Adapt the Christofides heuristic to obtain a 5/3-approximation.

Problem 2 Let $G = (V, A)$ be a directed graph with arc weights $c : A \to \mathcal{R}^+$. Define the density of a directed cycle C as $\sum_{a \in C} c(a)/|V(C)|$ where $V(C)$ is set of vertices in C.

1. A cycle with the minimum density is called a minimum mean cycle and such a cycle can be computed in polynomial time. How?

   Hint: Given density λ, give a polynomial-time algorithm to test if G contains a cycle of density $< \lambda$. Now use binary search.

2. Consider the following algorithm for ATSP. Given G (with c satisfying asymmetric triangle inequality), compute a minimum mean cycle C. Pick an arbitrary vertex v from C and recurse on the graph $G' = G[V - C \cup \{v\}]$. A solution to the problem on G can be computed by patching C with a tour in the graph G'. Prove that the approximation ratio for this heuristic is at most $2H_n$ where $H_n = 1 + 1/2 + \ldots + 1/n$ is the nth harmonic number.

Problem 3 For Metric-TSP consider the nearest neighbour heuristic discussed in class. Prove that the heuristic yields an $O(\log n)$ approximation. (Hint: use the basic idea in the online greedy algorithm for the Steiner tree problem from Lecture 1 (Spring 2011)). **Extra Credit:** Give an example to show that there is no constant c such that the heuristic is a c-approximation algorithm.

Problem 4 Problem 7.2 from Shmoys-Williamson book.

Problem 5 Let $G = (V, E)$ be a *directed graph* with non-negative edge costs $c : E \to \mathbb{R}_+$. Consider the problem of finding the min-cost strongly connected sub-graph problem. That is, we want to find $E' \subseteq E$ of smallest cost such that $G(V, E')$ is strongly connected.

- The following problem can be solved in polynomial time. Given an edge-weighted directed graph $G = (V, E)$ find the min-cost *arborescence* rooted at a given node $r \in V$. Using this describe a 2-approximation for the min-cost strongly connected subgraph problem by computing an in-arborescence and an out-arborescence.

- Now we will consider the unweighted case of the problem, that is, each edge $e \in E$ has weight 1. Suppose that the longest simple cycle in G has at most k edges. Show that the optimum must contain at least $\frac{k}{k-1}(n-1)$ edges. Now consider the following greedy algorithm. Find a simple cycle C of length at least 3 if it exists; otherwise C is any cycle of length 2. Contract the vertices of the cycle C into a vertex and recurse on the remaining graph. Formalize this algorithm and show that this algorithm gives a 1.75 approximation.

Problem 6 Let $G = (V, E)$ be an undirected graph with non-negative edge-weights. We will be interested in finding the min-cost k-edge-connected subgraph problem — the goal is to find a min-cost set $E' \subseteq E$ such that the graph (V, E') is k-edge-connected. Now consider the rooted counterpart where we are given a specified root node $r \in V$ and the goal is to find a min-cost $E' \subseteq E$ such that for each $v \in V$ the edge-connectivity from r to v is at least k in (V, E').

- Prove that the k-edge-connected subgraph problem is the same as its rooted counterpart in undirected graphs. (The problems are NP-Hard for $k \geq 2$).

- In directed graphs the rooted version is solvable in polynomial time — that is the min-cost set $A' \subseteq A$ (here $H = (V, A)$ is a directed graph) such that in (V, A') there are k edge-disjoint paths from r to v for each $v \in V$. We will use this directed graph rooted result to obtain a 2-approximation for the unweighted k-connected-subgraph problem. Given undirected graph $G = (V, E)$, obtain a directed graph $H = (V, A)$ by replacing each undirected edge $uv \in E$ by directed edges (u, v) and (v, u) with the same cost as that of uv. Pick an arbitrary root r and solve the rooted k-connectivity version of the problem in H. Let $A' \subseteq A$ be the directed edges chosen by the algorithm. Obtain $E' \subseteq E$ by choosing uv to be included in E' if (u, v) *or* (v, u) is in A'. Argue why E' is feasible. Argue that there is an optimum solution $A^* \subseteq A$ of cost at most twice the cost of the optimum solution for the original problem in the undirected graph G. Put things together to prove that the algorithms gives a 2-approximation.

Problem 7 Problem 7.4 from Williamson-Shmoys book.