

Problem Set #6

Prof. Michael A. Forbes
 Prof. Chandra Chekuri

Due: *Wed., Oct. 23, 2019 (10:00am)*

All (non-optional) problems are of equal value.

1. Suppose f is an (s, t) -flow in a network $G = (V, E)$ and let f' be another (s, t) -flow. Prove that there is an (s, t) -flow of value $|f'| - |f|$ in the residual network G_f , where $|f|, |f'|$ are the values of the flows f, f' respectively.

Note that $|f'| - |f|$ may be *negative*.

2. Suppose you have already computed a maximum flow f^* in a flow network G with *integer* edge capacities.
 - (a) Describe and analyze an efficient algorithm to update the maximum flow after the capacity of a single edge is *increased* by 1.
 - (b) Describe and analyze an efficient algorithm to update the maximum flow after the capacity of a single edge is *decreased* by 1.

For both parts there exist algorithms that run in $O(m + n)$ time where m is the number of edges and n is the number of nodes in the flow network.

3. Let $G = (V, E)$ be a flow network with integer edge capacities. We have seen algorithms that compute *some* minimum (s, t) -cut. For both problems below assume that you only have black-box access to an algorithm that given G and nodes s, t outputs a minimum (s, t) -cut.
 - (a) Given G and s, t and an integer k describe an efficient algorithm that checks whether G has a minimum cut with at most k edges.
 - (b) (**optional, not for submission**) Given G and s, t describe an efficient algorithm that decides whether G has at least two distinct minimum (s, t) -cuts. Alternatively, does G have a *unique* minimum (s, t) -cut?

No proof of correctness necessary but we recommend a brief justification. And make sure you have a clear and understandable algorithm.

4. (**optional, not for submission**) Problem 1 from <https://courses.engr.illinois.edu/cs473/fa2017/hw7.pdf>.
5. (**optional, not for submission**) Given a flow network $G_f = (V, E)$ with edge capacities $c : E \rightarrow \mathbb{Z}_+$ and an assignment of non-negative integers to the edges $f : E \rightarrow \mathbb{Z}$, describe a linear time algorithm to check if f is a maximum flow and to output a minimum (s, t) -cut if it is a maximum flow.