Design and Analysis of Algorithms

Practice-sheet 4: **Maximum Flow**

Date: *4th October, 2015*

1. (**Flow fundamental**) Suppose you are given a directed graph $G = (V, E)$ with a positive integer capacity c_e on each edge, a designated source $s \in V$, and a designated sink $t \in V$. You are also given an integer maximum $s - t$ flow in G, defined by a flow value f_e on each edge e.

   Now suppose we pick a specific edge $e \in E$ and increase its capacity by one unit. Show how to find a maximum flow in the resulting capacitated graph in time $O(m + n)$, where m is the number of edges in G and n is the number of vertices in G.

2. **Blood bank problem**

   We all know the basic rule for blood donation: A patient of blood group A can receive only blood of group A or O. A patient of blood group B can receive only blood of group A or O. A patient of blood group O can receive only blood of group O. A patient of blood group AB can receive blood of any group.

   Let s_O, s_A,s_B, s_{AB} denote the supply in whole units of the different blood types on hand. Assume that the hospital knows the projected demand for each blood type d_O,d_A,d_B, and d_{AB} for the coming week. Give a polynomial time algorithm to evaluate if the blood on hand would suffice for the projected need. You should formulate this problem as a max-flow problem, establish a relation between the two problems by stating a theorem, and then you should prove the theorem.

3. (**Max-damage to network**)

   You are given a flow network with unit capacity edges: It consists of a directed graph $G = (V, E)$, a source $s \in V$, and a sink $t \in V$; and $c_e = 1$ for every $e \in E$. You are also given a parameter k.

   The goal is to delete k edges so as to reduce the maximum-flow on G by as much as possible. In other words, you should find a set of edges $F \subseteq E$ so that $|F| = k$ and the maximum $s - t$ flow in $G' = (V, E - F)$ is as small as possible subject to this. Give a polynomial time algorithm to solve this problem.

4. (**Negative edge capacities**)

   Let $G = (V, E)$ be a directed graph with source $s \in V$, sink $t \in V$ and edge capacities $\{c_e\}$. Suppose that for each edge that has neither s nor t as an endpoint, we have $c_e \geq 0$. Thus c_e can be negative for edges e that have at least one endpoint equal to s or t. Give a polynomial time algorithm to find an $s - t$ cut of minimum value in this graph. (Despite the new nonnegativity requirements, we still define the value of an $s - t$ cut (A, B) to be the sum of capacities of all edges for which the tail of e is in A and the head of e is in B.)

5. (**unique min-cut**)

   Let $G = (V, E)$ be a directed graph with source $s \in V$, sink $t \in V$ and nonnegative edge capacities $\{c_e\}$. Give a polynomial time algorithm to decide whether G has a unique minimum s-t cut (i.e., an s-t cut of capacity strictly less than that of all other s-t cuts.)

6. (**Vertex disjoint paths**)

   There is a directed graph $G = (V, E)$ on n vertices and m edges. There are two vertices $s, t \in V$. Two paths from s and t are said to be vertex disjoint if they do not share any vertex except s and t Design a polynomial time algorithm to compute the maximum number of vertex disjoint paths from s to t.

7. (**Application with lower bound on flow**)

   There is an airline which has to serve n flights per day. Each flight i has four parameters: <origin, destination, departure-time, arrival-time>. One of the biggest factor of running the airline is the number of carriers (airplanes) that it requires to serve all n flights. It can be observed that a single airplane can serve multiple flights. In particular, if there are flights i and j with parameters $< s_i, t_i, d_i, a_i >$ and $< s_j, t_j, d_j, a_j >$ such that $t_i = s_j$ and $a_i < d_j$, then the airplane serving flight i can also serve flight j. In this manner, a single airplane can serve multiple flights. Design a polynomial time algorithm to compute the least number of airplanes needed to serve all n flights.

8. (**Circulation with lower bound on flow**)

   Recall the circulation problem which we solved by reducing to max-flow problem. We shall now extend this problem further.

   There is a flow network $G = (V, E)$ with source, sink $t \in V$ and nonnegative edge capacities $\{c_e\}$. Each vertex v has a demand d_v which is a real number. In addition each edge has a nonnegative number ℓ_e. Design a polynomial time algorithm to determine if there exists a circulation $f : E \to R$ such that

   (a) For each vertex v, $f_{in}(v) - f_{out}(v) = d_v$.

   (b) For each edge e, $\ell_e \le f(e) \le c_e$.

   Hint: Reduce this problem to an instance of circulation problem without any lower bound on edges.

9. **An amazing application of Min-cut**

   Suppose we are given a directed network $G = (V, E)$ with a root node r and a set of *terminals* $T \subseteq V$. We would like to disconnect many terminals from r, while cutting relatively few edges.

   We make this trade-off precise as follows. For a set of edges $F \subseteq E$, let $q(F)$ denote the number of nodes $v \in T$ such that there is no $r - v$ path in the subgraph $(V, E - F)$. Give a polynomial time algorithm to find a set F of edhes that maximizes the quantity $q(F) - |F|$. (Note that setting F equal to the empty set is an option.)