Design and Analysis of Algorithms (CS345A)

Practice-sheet : **Maximum Flow**

1. (**Flow fundamentals**)

   Suppose you are given a directed graph $G = (V, E)$ with a positive integer capacity c_e on each edge $e \in E$, 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 $e \in E$. 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. (**Mobile phone and base stations**)

   Consider a set of mobile computing clients in a certain town who each need to be connected to one of several possible base stations. There are n clients with the position of each client specified by its (x, y) coordinates in the plane. There are also k base stations; the position of each of these is specified by (x, y) coordinates as well.

   For each client, we wish to connect it to exactly one of the base stations. Our choice of connections is constrained in the following ways. There is a range-parameter r - a client can only be connected to a base station that is within distance r. There is also a load parameter L - no more than L clients can be connected to any single base station.

   Your goal is to design a polynomial time algorithm for the following problem. Given the position of a set of clients and a set of base stations, as well as the range and load parameters, decide whether every client can be connected simultaneously to a base station, subject to the range and load conditions in the previous paragraph.

4. (**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.

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. (**Farthest Min-cut**)

   There may be many min-cuts in a flow network from s to t. The min-cut defined by $(A_0, \bar{A}_0)$ with $s \in A_0$ and $t \in \bar{A}_0$ is said to be the farthest min-cut from s to t if for every other min-cut defined by $(A, \bar{A}')$, the subset A' must be contained in A_0. Design an efficient algorithm to compute the farthest min-cut from s to t.

8. New problem
   (**bounding the value of mincut**) There is a flow network where capacity of each edge is 1. Furthermore, the distance from s to t is $n^{2/3}$. Show that the Ford-Fulkerson algorithm will take $O(mn^{2/3})$ to compute the maximum flow from s to t.
   Hint: The title of the problem is the hint.

For FUN only

Exercise: There is a directed flow network $G = (V, E)$ with a designated source s and sink t. Let f be a maximum (s,t)- flow. Design a polynomial time algorithm to determine if f is unique maximum (s,t)-flow.
Hint: Focus on the residual network G_f to establish a necessary and sufficient condition for the network G to have multiple maximum flows from s to t.

Establish the following result:

Theorem 1. *There exists a maximum (s,t)-flow f^* different from f if and only if G_f has a cycle of length 3 or more.*

Proof. **Part 1:** *(easy)*
If there is a cycle of length 3 in G_f, then there exists a maximum (s,t)-flow f^* different from f.

Let C be a cycle in G_f with length at least 3. Let c be the capacity of the least capacity path from s to t in C. Process each edge of this cycle and update the flow just like we did in the case of FF algorithm. Along similar lines as in the analysis, the new flow f^* is a valid flow : conservation and capacity constraints hold. Note that if s (similarly t) appears on C, then it must have a backward and a forward edge incident on it. Therefore, it can be shown that the net flow leaving s (or entering t) remains unchanged. Hence value(f) = value(f^*). Note that the fact that cycle C must be of length 3 is exploited to ensure that the edges of C incident on any vertex of C must have distinct endpoints.

Part 2: *(slightly longer)*
Let f^* be a maximum (s,t)-flow that differs from f. Then there exists a cycle in G_f of length at least 3.
Hint: Let (u,v) be an edge such that $f(u,v) > f^*(u,v)$. Then there is a cycle in G_f passing through edge (v,u). $\square$

sketch of **Part 2:**
For the sake of simplicity, the proof is provided for unit capacity network, that is, each edge in G has unit capacity. However, it can be extended for positive and non-integral capacity.

Since the graph is unit capacity and $f(u,v) > f^*(u,v)$, so $f(u,v) = 1$ and $f^*(u,v) = 0$. Let P be any (s,t)-path in G passing through (u,v) such that $f(e) > 0$ for each edge e on P. Let P^r denote the reverse of P. It can be seen that P^r exists in G_f.

We reduce the flow f on each edge of path P by 1 unit. Let f' be resulting flow in G. Note that $f'(u,v) = 0$. Let $G' = G \backslash \{(u,v)\}$. It can be observed that f' is a valid flow in G'.

Lemma 2. *There is a (s,t)-path in $G'_{f'}$.*

Proof. Recall than f^* is a maximum (s,t) flow in G and $f^*(u,v) = 0$. So f^* is also a maximum flow in G' as well. Since the value of f^* differs from value of f' by 1. So there must be an augmenting path from s to t in $G'_{f'}$. □

Using Lemma 2, let P' be a (s,t)-path in $G'_{f'}$. It can be seen that (u,v) does not belong to P'. We can use P' and P^r to show the existence of a cycle in G_f that passes through (v,u) as follows. Let $P = P_1 :: (u,v) :: P_2$. It can be seen that P_1 and P_2 exist in $G'_{f'}$. P' might intersect P_1 and P multiple times. Let x be the vertex on P_1 such that P' after passing through x never intersects P_1. Let y be the first vertex of P_2 visited by P' after x. Observe that the possibility of $x = u$ and $y = v$ is ruled out since $(u,v)\ \not{i}nP'$. It can be seen that the portion of P' from x to y is present in G_f as well, and let us denote it by $p_{x,y}$. Recall that P^r is present in G_f and it passes through (v,u). The segment of this path from y to x concatenated with $p_{x,y}$ is a cycle in G_f. This cycle has length at least 3 since either $x \neq u$ or $y \neq v$ as stated above.