CSE 521: Design and Analysis of Algorithms I **Spring 2016**

Lecture 17: Maximum Flow and Minimum Cut

Lecturer: Shayan Oveis Gharan *May 23rd* *Scribe: Utku Eren*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications.*

Last lecture we studied duality of linear programs (LP), specifically how to construct the dual, the relation between the optimum of an LP and its dual, and some duality applications. In this lecture, we will talk about another application of duality to prove one of the theorems in combinatorics so called *Maximum Flow-Minimum Cut Problem*. The theorem roughly says that in any graph, the value of maximum flow is equal to capacity of minimum cut.

17.1 Maximum Flow

Let G be a directed graph, assume that $c : E \to \mathbb{R}_+$ is the cost per capacity. The maximum flow problem is defined as, given two specific vertices namely source s and sink t, the objective is to find flows for each path that obeys capacities along the path and maximizes the net flow from s to t. To get a better understanding of the problem lets examine the following graph. There are three different paths on the graph in Figure 17.1

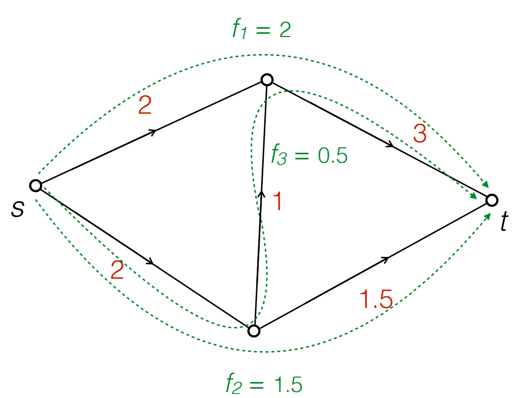

Figure 17.1: Maximum Flow on an example graph

and capacities on each edge are given with values typed in red color. We can send a flow of $f_1 = 2$ from path 1 and send another flow of $f_2 = 1.5$ from path 2. This configuration fully utilizes the capacities of some edges along the paths with net flow of $\sum_i f_i = 3.5$, however it is not the maximum flow. To maximize the flow we can send another flow $f_3 = 0.5$ from path 3 without violating any capacity constraints, which maximizes the net flow from s to t as $\sum_i f_i = 4.0$. To justify this is the maximum flow, we can divide graph into two sets as the source and the rest of the graph. We can consider this division as a cut separating the source from the sink. The summation of capacities of edges crossing this cut from the source side the destination side yields is 4. So, the maximum possible flow from s to t is at most 4. In this lecture we see that the maximum flow from s to t is equal to the smallest cut separating s from t with minimum sum of capacities. This is what Maximum Flow-Minimum Cut Theorem states.

17.1.1 LP Formulations for Maximum Flow

Before delve into the *Maximum Flow-Minimum Cut Theorem*, lets focus on the Maximum Flow problem, specifically, how to find the maximum flow in any graph. For this purpose, we can cast the problem as a linear program (LP). First, we define the notation given in Figure 17.2, where for a node v we label the incoming edges as $d^-(v)$ and the outgoing ones as $d^+(v)$.

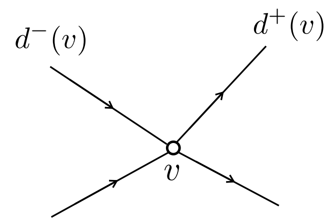

Figure 17.2: Notation on incoming and outgoing edges

First of all, because of the capacity constraints on each edge we need whole flow over an edge to be at most the capacity of the edge. Secondly, to satisfy physics of the flow, we need to have a conservation of flow equation, specifically, over all nodes except s and t, summation of all incoming flows has to be equal to summation of all outgoing flows. Now we can write an LP to solve for the maximum flow as follows:

$$\begin{aligned} \max \quad & \sum_e x_e \\ \text{s.t.} \quad & 0 \le x_e \le c_e & \forall e \\ & \sum_{e \in d^-(v)} x_e = \sum_{e \in d^+(v)} x_e & \forall v \in V \setminus \{s,t\} \end{aligned} \tag{17.1}$$

Remark 17.1. *This is not necessarily the best algorithm to find the maximum flow and there are many other combinatorial algorithms for this task. One significant advantage of LP is to ease of posing the problem and once the problem is formulate by an LP, there are many of-the-shelf LP solvers to get the optimum solution. So the personal effort of this approach is minimum.*

Remark 17.2. *Once the LP is formulated, it is easier to extend the problem to more complicated cases. For instance, one can try to find a maximum flow with the smallest cost over the network.*

There is another way to write the maximum flow problem using an LP (with exponentially many constraints). We can do this using a nonnegative variable corresponding to each path connecting s to t. Let $\mathcal{P}$ be the set of all such paths. Then,

$$\begin{aligned} \max \quad & \sum_{P \in \mathcal{P}} f_P \\ \text{s.t.} \quad & \sum_{P: e \in P} f_P \le c_e & \forall e \\ & f_P \ge 0 & \forall p \in \mathcal{P} \end{aligned} \tag{17.2}$$

Basically, this formulation assigns a nonnegative amount of flow to each path as f_P, and solves for these values. Capacity constraints are imposed on the utilization of an edge by all paths which includes that edge. Formulations (17.1) and (17.2) are equivalent. This is a nice exercise. But formulation (17.2) can be harder to solve, since number of variables are the all possible paths from s to t, and there may be exponential number of these paths. Although we do not describe here, but in principle one can use the ellipsoid method to find an optimum solution of the second LP.

The reason that we describe (17.1) is that it has a an easier to understand dual. Writing the dual for the formulation (17.2) yields the following:

$$\begin{aligned} \min \quad & \sum_e c_e y_e, \\ \text{s.t.} \quad & \sum_{e \in P} y_e \geq 1 \quad \forall P, \\ & y_e \geq 0 \qquad \forall e. \end{aligned} \tag{17.3}$$

The primal is certainly feasible (we can let $f_P = 0$ for all $P \in \mathcal{P}$) and the dual is also feasible (we can let $y_e = 1$ for all edges), so strong duality hols. Thus objective value of the primal problem is equal to objective of the dual problem.

17.2 Minimum Cut

Before we talk about the minimum cut problem let's start with providing a definition for $s - t$ cut.

Definition 17.3. *For a set $S \subseteq V$, we say $(S, \overline{S})$ is an $s - t$ cut if $s \in S$ and $t \notin S$*

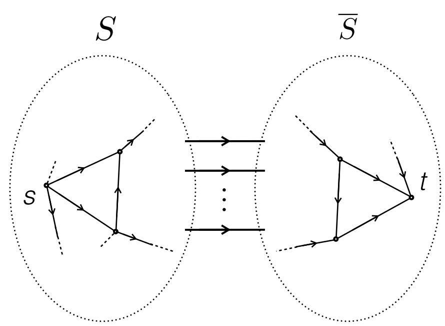

Figure 17.3: $s - t$ cut illustration

The capacity of an $s - t$ cut $(S, \overline{S})$ is defined as follows:

$$c(S, \overline{S}) := \sum_{\substack{(u,v) \\ u \in S, v \notin S}} c_{u,v} \tag{17.4}$$

In the minimum $s - t$ cut problem we want to find the an $s - t$ cut with minimum capacity,

$$\min_{S \text{ is } s-t \text{ cut}} c(S, \overline{S}).$$

The following theorem is the main result that we prove in this lecture.

Theorem 17.4 (Maximum-flow Minimum-cut theorem)**.** *For any graph G, and any two vertices $s, t \in V$, the size of maximum $s - t$ flow is equal to size of the minimum $s - t$ cut.*

Proof. We already know that maximum flow is the solution of LP (17.1) which is equivalent to LP (17.2). Also we know from the strong duality that primal ((17.2)) cost is equal to dual ((17.3)). Therefore all we need to show is that the minimum cut is always equal to this dual. Let y^* be an optimum solution for the dual problem. So, all we need to show is that

$$\sum c_e y_e^* = \min_{s-t \text{ cut}} c(S, \overline{S}). \tag{17.5}$$

We prove this equality in two steps. First, we show that

$$\sum c_e y_e^* \leq \min_{s-t \text{ cut}} c(S, \overline{S}). \tag{17.6}$$

Remember we have y^* is an optimizer of a minimization problem. Therefore, it is enough to show that for any $s-t$ cut $(S, \overline{S})$, there exists a vector y such that y is a feasible LP solution and that

$$\sum c_e y_e \leq c(S, \overline{S}). \tag{17.7}$$

So, fix an $s-t$ cut $(S, \overline{S})$. For any edge e we let,

$$y_e = \begin{cases} 1 & \text{if } \ e \in (S, \overline{S}) \\ 0 & \text{otherwise} \end{cases} \tag{17.8}$$

Observe that the above vector is a feasible solution to LP 17.3; this is because any path between s and t has to cross $(S, \overline{S})$ at least ones. Since y is equal to 1 for any edge in that (directed) cut, the sum of y_e's along any $s-t$ path is at least 1. So, it remains to verify (17.7). Observe that

$$\sum_e c_e y_e = \sum_{u \sim v : u \in S, v \notin S} c_{u,v} y_{u,v} = \sum_{u \sim v : u \in S, v \notin S} c_{u,v} = c(S, \overline{S}),$$

as desired. This proves (17.6).

So, it remains to prove

$$\sum c_e y_e^* \geq \min_{s-t \text{ cut}} c(S, \overline{S}) \tag{17.9}$$

Since the RHS is a minimization problem it is enough to prove the following: Given a feasible vector y, there exists an $s-t$ cut $(S, \overline{S})$ such that

$$\sum_e c_e y_e \geq \min_{s-t \text{ cut}} c(S, \overline{S}). \tag{17.10}$$

Fix a feasible solution y of the dual. The goal is to round y into an $s-t$ cut $(S, \overline{S})$.. For any edge e let y_e be the length of e. For any vertex $v \in V$, let y_v be the shortest path distance from s to v with respect to the aforementioned lengths. Observe that, since y is a feasible LP solution, we have $y_s = 0$ and $y_t = 1$.

Having $\{y_v\}$'s we just need to run a clustering algorithm. We choose a threshold r and we split the vertices into a set S of those at distance at most r from s and the set $\overline{S}$ of vertices at distance more than r from S. This idea is reminiscent of the spectral clustering algorithm. Except in this case we are running the clustering algorithm with respect to the LP solution as opposed to the second eigenvector of the graph. As a sanity check, observe that, if y is an integral vector, then for a set S of vertices we have $y_v = 0$ and for all vertex $v \notin S$ we have $y_v = 1$. So, the above algorithm correctly recovers an $s-t$ cut of capacity $\sum_e c_e y_e$.

For a threshold r sampled from a uniform distribution $\mathcal{U}(0,1)$, let $S_r = \{v : \ y_v \leq r\}$, We claim that

$$\mathbb{E}_r \left[c(S_r, \overline{S}_r) \right] \leq \sum_e c_e y_e. \tag{17.11}$$

Note that a proof of the above equation directly implies (17.10) which completes the proof of Theorem 17.4.

We say an edge (u, v) is cut by $(S_r, \overline{S_r})$ if $u \in S_r$ and $v \notin S_r$. Observe that whenever (u, v) is cut we have to pay c_e in the LHS. So, by linearity of expectation, to prove the above inequality it is enough to show that any edge (u, v) is cut with probability at most $y_{u,v}$.

So, let us study the probability that an edge (u, v) is cut. Figure 17.4 gives the y_v values of all vertices of G sorted in an increasing order. Consider an edge (u, v) where $y_u = 0.35$ and $y_v = 0.45$. This edge is cut when the threshold r lies in the interval $(0.35, 0.45)$ which happens with probability 0.1. In general edge u, v is cut with probability $\max\{y_v - y_u, 0\}$. But the latter is at most $y_{u,v}$. This is because one option to go from s to v is to first go to u and then go from to v using edge (u, v). Therefore,

$$y_v \leq y_u + y_{u,v}.$$

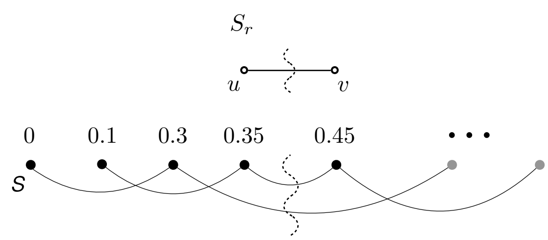

Figure 17.4: Illustration of the probability of that (u, v) is cut.

Consequently, by linearity of expectation we have

$$\begin{aligned} \mathbb{E}_r\left[c(S_r, \overline{S_r})\right] &= \sum_{(u,v)} c_{u,v}\ \mathbb{P}\Big[(u, v) \text{ is cut}\Big] \\ &= \sum_{(u,v)} c_{u,v}\ \mathbb{P}\Big[y_u \leq r \leq y_v\Big] \\ &= \sum_{(u,v)} c_{u,v}\ \Big(y_v - y_u\Big) \\ &\leq \sum_{(u,v)} c_{u,v}\ y_{u,v} \end{aligned} \tag{17.12}$$

This proves (17.11), which implies (17.10). Putting (17.10) and (17.7) together implies (17.5). This completes the proof of Theorem 17.4. □

Remark 17.5. *An interesting thing to note here is that, given y, we can give a distribution over $s - t$ cuts such that the expected capacity of these cuts is less than or equal to $\sum c_e y_e$. Now, suppose we construct the set of $s - t$ cuts with the optimal value i.e. y_e^*, we already know that there are no cuts with a value less than $\sum c_e y_e^*$, so in fact, all of the cuts in this distribution must be minimum $s - t$ cuts.*