Lecture 9

1 The Path Selection Problem

We consider the problem of selecting paths for routing packets in a general graph. We are given a directed graph $G = (V, E)$ and k pair of its vertices $(s_1, t_1), \ldots, (s_k, t_k)$ as input. The problem is to find k paths, one connecting s_i to t_i for each i, such that the maximum number of paths that use any edge (the congestion) is minimized.

A direct translation of the problem as an LP is

$$\begin{array}{ll} \min C & \\ \sum_{p: s_i \rightsquigarrow t_i} f(p) = 1 & \text{for } 1 \le i \le k \\ \sum_{e \in p} f(p) \le C & \text{for edges e} \\ f(p) \ge 0 & \text{for paths p} \end{array} \tag{1}$$

We cannot solve the above LP directly, since there may be an exponential number of paths. Furthermore, the LP does not capture the problem thoroughly, since $f(p)$ should be integer.

2 LP Randomized Rounding

We can rewrite LP 1 by assigning flows to edges:

$$\begin{array}{l} \min C \\ \sum_{e=(u,v)} f^i(e) - \sum_{e=(v,u)} f^i(e) = \left\{ \begin{array}{ll} 1 & \text{for } u = s_i \\ -1 & \text{for } u = t_i \\ 0 & \text{otherwise} \end{array} \right. \\ \sum_i f^i(e) \le C \\ f^i(e) \ge 0 \end{array} \tag{2}$$

The size of the new LP is small enough that we can find the optimal solution of the LP in polynomial time.

In the last lecture, we saw that given edge flows for LP 2 with minimum congestion C, one can compute a solution to LP 1 with congestion C in which at most $k|E|$ paths have nonzero flow. We like to choose k paths from these paths such that the congestion is not much higher than C.

Let $1 \le i \le k$. Among all paths from s_i to t_i, we select a path p randomly with probability $f(p)$ as the route from s_i to t_i. (Note that the sum of $f(p)$ over all paths p from s_i to t_i is 1.)

Let $X(e)$ denote the congestion of edge e. The expected value of $X(e)$ is $\sum_{e \in p} f(p) \le C$. Let $q = \Pr[\text{congestion on } e \ge (1 + \delta)C]$.

Let $X_i = 1$ if the path from s_i to t_i uses edge e, and $X_i = 0$ otherwise. We have $X(e) = \sum_{i=1}^{k} X_i$. Since random variables X_i are independent, we can upper-bound q using the Chernoff bound.

When $C \ll \log |E|$, we have $q \le 2^{-\mu\delta} \le |E|^{-2}$ for $\delta = \Theta(\log |E|/C)$. When $C = \Omega(\log |E|)$, we have $q \le \exp(-\delta^2\mu/4) \le |E|^{-2}$ for $\delta = \Theta(\sqrt{\log |E|/C})$. Therefore with probability $\ge 1 - 1/|E|$, we have

$$\text{congestion } \le C_{LP} + O(\sqrt{C_{LP} \log |E|} + \log |E|).$$

3 The Primal-Dual Algorithm

Now let's look at the dual of LP 1:

$$\begin{array}{ll} \max \sum_i d(i) & \\ d(i) - \sum_{e \in p} d(e) \le 0 & \text{for paths } p : s_i \rightsquigarrow t_i \\ \sum_e d(e) = 1 & \\ d(e) \ge 0 & \text{for edges } e \end{array} \tag{3}$$

This LP can be interpreted as "distributing one unit of weight among edges such that the sum of shortest paths from s_i to t_i for $1 \le i \le k$ is maximized."

Let $d(e)$ be the weight of edge e in an arbitrary nonnegative weighting of the edges, and $d(s_i, t_i)$ the distance between s_i and t_i in the weighted graph. By weak LP duality, we have

$$C_{opt} \ge \frac{\sum_i d(s_i, t_i)}{\sum_e d(e)}. \tag{4}$$

Consider the following algorithm for path selection that has two parameters $\alpha, \beta > 1$:

- Establish one path p_i for each (s_i, t_i).
- Weight edges by the rule $d(e) = \alpha^{f(e)}$ where $f(e)$ is the congestion of edge e.
- While $\sum_i d(p_i) \ge \alpha\beta \sum_i d(s_i, t_i)$, find i such that $d(p_i) - \alpha d(s_i, t_i)$ is maximum. Then replace p_i be the shortest path between s_i and t_i, and update $d(e)$.

Assume that the algorithm terminates, and let C_{max} be the maximum congestion of the selected paths. We will use Inequality 4 to bound C_{max} in terms of C_{opt}. Let n_l denote the number of edges with congestion l. For $l' = C_{max} - 2\log_\alpha |E|$, we have

$$\sum_e d(e) = \sum_{0 \le l \le C_{max}} n_l \alpha^l \le (1 + 1/|E|) \sum_{l' \le l \le C_{max}} n_l \alpha^l,$$

and

$$\sum_i d(s_i, t_i) \ge \frac{1}{\alpha\beta} \sum_i d(p_i) = \frac{1}{\alpha\beta} \sum_{0 \le l \le C_{max}} n_l l \alpha^l.$$

Inequality 4 implies

$$C_{opt} \ge \frac{l'}{(1 + 1/|E|)\alpha\beta}.$$

Hence, we get

$$C_{max} \leq \alpha\beta(1 + 1/|E|)C_{opt} + 2\log_\alpha |E|.$$

Now we prove that the algorithm terminates by showing that $\sum_e d(e)$ is decreasing during the run of the algorithm.

Assume that the algorithm chooses i to replace the path p_i by the shortest path between s_i and t_i. By the choice of i, we have

$$d(p_i) - \alpha d(s_i, t_i) \geq \frac{1}{k}\sum_i (d(p_i) - \alpha d(s_i, t_i)) \geq \frac{1}{k}(1 - \frac{1}{\beta})\sum_i d(p_i).$$

By replacing p_i by the shortest path, $\sum_e d(e)$ is decreased by at least

$$(1 - \frac{1}{\alpha})d(p_i) - (\alpha - 1)d(s_i, t_i) \geq \frac{1}{k}(1 - \frac{1}{\alpha})(1 - \frac{1}{\beta})d(p_i).$$

Initially, we have $\sum_e d(e) \leq |E|\alpha^k$. Therefore the algorithm terminates after at most $O((\alpha - 1)(\beta - 1)k(\log |E| + k \log \alpha))$ steps.