Lecture Outline:

- Approximation Schemes for Multicommodity Flow Problems

In this lecture, we will present a general *primal-dual* approach for solving multicommodity flow problems. This technique could also be viewed as a *Lagrangian relaxation* approach, which is widely used in optimization, and can also been seen to be similar to the *exponential weight method* used in online learning and other algorithms. The approach we present extends to more general packing/covering problems. We note that all of these problems, in fact, can be solved optimally by solving associated linear programs. The algorithms we discuss in this lecture will not solve the linear program explicitly and hence are likely to be much faster than LP-based algorithms (but they will only return an approximate solution).

1 Multicommodity Flow Problem

In this section, we will consider the sum-flow version of multicommodity flow. The approach does apply to the demands version as well. The sum-flow version of the problem is defined as:

Problem 1. Given a graph $G(V, E)$, a capacity function $C : E \to Z^+$, a distance function $D : E \to Z^+$ and k commodity (s_i, t_i) pairs, the goal is the maximize the total flow that can be sent from s_i to t_i subject to capacity and flow constraints.

LP for our path-flow based approach:

P_i = set of simple paths from s_i to t_i

$p_i^j = j^{th}$ path in P_i

f_i^j = flow on p_i^j for commodity pair (s_i, t_i)

$$\begin{aligned} \max \sum_{i,j} f_i^j \\ \text{such that} \\ \sum_{e \in p_i^j} f_i^j &\leq c_e \ \forall e \\ f_i^j &\geq 0 \ \forall p_i^j \end{aligned}$$

The dual of the LP above can be written as:

$$\min \sum_e c_e d_e$$
$$\text{such that}$$
$$\begin{aligned} \sum_{e \in p_i^j} d_e &\geq 1 \ \forall p_i^j \\ d_e &\geq 0 \forall e \end{aligned}$$

1.1 Single Source - Single Sink Commodity Pair Version

The algorithm to be proposed is a $(1+\omega)$-approximation algorithm to approximate our dual. In this case, we assume that we have a single source s and a sink t. The algorithm is:

Algorithm 1: Single Commodity Case

1. $d_e \leftarrow \delta$, $\forall e$
2. **repeat**
   - 2.1. Find shortest path P from s to t (send unit flow on the path)
   - 2.2. $d_e \leftarrow d_e(1+\varepsilon)$, $\forall e \in P$

   until *shortest path* $s-t$ *has length* ≥ 1
3. Scale down flow to satisfy capacity constraint

As a warm up, let us consider the single commodity version. Let's assume that it will take t iterations for algorithm to finish. Define:

- f_i = the flow at the end of iteration i
- $D(i)$ = value of the dual at the end of iteration i
- β = optimal dual value = optimal primal value
- $\alpha(i)$ = shortest path distance at iteration i

Lemma 1. *There exists a feasible flow of value* $\frac{t}{\log \frac{1+\varepsilon}{\delta}}$

Proof. Flow along edge $e \leq \log_{1+\varepsilon} \frac{1+\varepsilon}{\delta}$. We obtain the flow f_t sending t units of flow, but $f_t = t$ might not be feasible. Therefore, we scale the flow down by $\log \frac{2}{\delta}$ and obtain a feasible flow of total value $\frac{t}{\log \frac{2}{\delta}}$. □

Now we have to show that $\frac{t}{\log \frac{2}{\delta}}$ is indeed very close to β.

Goal: To show that $\frac{t}{\log \frac{2}{\delta}} \sim \beta$

We know that $D(t) \geq \beta$

$$D(i) = D(i-1) + 1 \times \alpha(i-1)$$

$$\alpha(i-1) \leq \frac{D(i-1)}{\beta} + D(i-1) = D(i-1)(1+\frac{1}{\beta})$$
$$\beta \leq D(t) \leq D(0)(1+\frac{1}{\beta})^t$$
$$\beta \leq \delta m(1+\frac{1}{\beta})^t$$
$$\frac{\beta}{\delta m} \leq e^{\frac{t}{\beta}}$$
$$t \geq \beta \ln(\frac{\beta}{\delta m})$$

$\therefore$ Total flow $\geq \frac{t}{1-\lg\delta} \approx \frac{t}{-\lg\delta}$

$\geq \frac{\beta[\ln(\beta)]-\ln(\delta)-\ln(m)}{-\lg(\delta)} \sim \frac{\beta(-\ln(\delta))}{-\lg(\delta)}$

$\approx \frac{\beta}{\lg(e)} \approx \beta$

Essentially, the algorithm keeps sending unit flow from the source to the sink along the shortest path, updating the "distance" of each edge to be exponential in the flow along the edge. Consequently, over time, the flows are being balanced across the multiple source-sink flow paths. This approach is similar to the ***exponential weight method*** used in online learning and other algorithms.

1.2 Multicommodity Version

Here we present the main algorithm for the multicommodity version of the problem:

Algorithm 2: Multicommodity Max-Flow

1. $d_e \leftarrow \delta$, $\forall e$
2. **repeat**
   - 2.1. Find shortest path p from s_i to t_i among all commodities, where the shortest is with respect to the distances given by d_e
   - 2.2. Send max-flow possible along path p subject to capacity constraint ($\min_{e \in p} c_e = c$)
   - 2.3. $d_e \leftarrow d_e(1+\varepsilon\frac{c}{c_e})$

   until *all shortest paths* $s_i - t_i$ *has length* ≥ 1 *for all* i
2. Return union of all flows scaled by $(\log_{1+\varepsilon}(\frac{1+\varepsilon}{\delta}))$

Lemma 2. *There exists a feasible flow of value* $\frac{f_t}{\log_{1+\varepsilon}\frac{1+\varepsilon}{\delta}}$.

Proof. Let f_e be the flow on the edge e after the algorithm finishes, and let this flow be the result of sending ℓ path-flows of values c_1, c_2, ..., c_ℓ. Then, we have

$$d_e = d_e(0)(1+\varepsilon\frac{c_1}{c_e})(1+\varepsilon\frac{c_2}{c_e})\cdots(1+\varepsilon\frac{c_\ell}{c_e})$$

Since $(1+\varepsilon x) \geq (1+\varepsilon)^x$ for $x < 1$, we obtain

$$1+\varepsilon \geq d_e \geq \delta(1+\varepsilon)^{\frac{c_1}{c_e}}(1+\varepsilon)^{\frac{c_2}{c_e}}\cdots(1+\varepsilon)^{\frac{c_\ell}{c_e}}$$

$$1 + \varepsilon \geq \delta(1+\varepsilon)^{\frac{f_e}{c_e}}$$
$$\therefore \frac{f_e}{c_e} \leq \log_{1+\varepsilon}(\frac{1+\varepsilon}{\delta})$$

Thus, scaling the flow down by a factor of $\log_{1+\varepsilon}\frac{1+\varepsilon}{\delta}$ would yield a feasible flow.

□

Lemma 3. *(Running Time) The running time of the algorithm is polynomial in $1/\varepsilon$ and the size of the graph.*

Proof. For every iteration, some edge is the bottleneck edge.

Number of times an edge is bottleneck

$$\leq \log_{1+\varepsilon}(\frac{1+\varepsilon}{\delta})$$

Thus, the total number of iterations is at most

$$\begin{aligned} & m\log_{1+\varepsilon}(\frac{1+\varepsilon}{\delta}) \\ = \; & m(1+\log_{1+\varepsilon}\delta) \\ = \; & m(1-\frac{\log_{1+\varepsilon}\delta}{\log_{1+\varepsilon}}(1+\varepsilon)) \approx O(\frac{-\log_{1+\varepsilon}\delta}{\varepsilon}) \\ = \; & poly(m, \frac{1}{\varepsilon}). \end{aligned}$$

Let k be the number of commodities, and T_{sp} be the time required to find a shortest path between a source and a sink. The total time taken is thus

$$kT_{sp}O(\frac{m(-\log_{1+\varepsilon}\delta)}{\varepsilon})$$

□

We finally need to prove that the value of the flow is close to the optimal.

Lemma 4. $\frac{f_t}{\log_{1+\varepsilon}\frac{1+\varepsilon}{\delta}} \geq \beta/(1+\omega)$*, where β is the optimal dual value and ω can be made arbitrarily close to zero by setting ε arbitrarily close to zero.*

Proof. We first prove that f_t is at least $\beta\ln(1/(\delta n))/\varepsilon$. Let $D(i)$ denote the dual at the end of iteration i and let $\alpha(i)$ denote the shortest source-sink path at the end of iteration i (according to the d_e labels). We have already seen that

$$D(j) = D(j-1) + \varepsilon(f_j - f_{j-1})\alpha(j-1).$$

If we add the above equation for $j = 1$ to i, we obtain

$$D(i) = D(0) + \varepsilon\sum_{j=1}^{i}(f_j - f_{j-1})\alpha(j-1).$$

It can be seen that $(D(i) - D(0))/(\alpha(i) - \alpha(0))$ is at least β since the setting provides a feasible solution to the dual. Substituting this bound on $D(i) - D(0)$ in the above equation, we obtain

$$\alpha(i) \leq \alpha(0) + \frac{\varepsilon}{\beta} \sum_{j=1}^{i} (f_j - f_{j-1}) \alpha(j-1).$$

By induction, we can then prove

$$\begin{aligned} \alpha(i) &\leq \alpha(0)(1 + \prod_{j}^{i} \varepsilon(f_j - f_{j-1})/\beta) \\ &\leq \alpha(0) e^{\varepsilon(f_i - f_{i-1})/\beta} \end{aligned}$$

Since $\alpha(0) \leq \delta m$, we have

$$f_t \geq \frac{\beta}{\varepsilon} \ln(1/(\delta m)).$$

In order to establish the approximation ratio, we bound the ratio of the flow given in Lemma 2 and β. This is at most

$$\frac{\varepsilon \log_{1+\varepsilon}((1+\varepsilon)/\delta)}{\ln(1/(\delta m))} = \frac{\varepsilon}{\ln(1+\varepsilon)} \frac{\ln(1+\varepsilon) - \ln \delta}{-\ln \delta - \ln m}$$

We can set δ much smaller than $\ln m$ and ε, and ε sufficiently small to make the above ratio equal to $1 + \omega$ for ω arbitrarily close to zero. □