COL758: Advanced Algorithms **Spring 2019**

Lecture 11: February 14

Lecturer: Naveen Garg *Scribe: Ankita Raj*

11.1 Previous lecture

In the last lecture, we studied the multiplicative weights update algorithm for simultaneous minimization. Given some convex functions $h_1(x), h_2(x), \ldots h_m(x)$ over a convex domain Q, we want to find a point in Q where $h_i(x) \leq 1 \; \forall i$. The algorithm is as follows:

Algorithm 1 Multiplicative weights update for simultaneous minimization

1: **init:** $y_i = 1 \; \forall i$
2: **repeat**
3: In round r,
4: Find $x^r \in Q$ such that $\sum_i y_i^r h_i(x^r) \leq \sum_i y_i^r$ (by calling *Oracle*)
5: Set $w^r = \frac{1}{\max_i h_i(x^r)}$
6: Update y_i: $y_i \leftarrow y_i e^{\epsilon w^r h_i(x^r)}$
7: **until** $\sum_r w^r \geq \frac{\ln m}{\epsilon^2}$
8: **return** $\overline{x} = \frac{\sum_r w^r x^r}{\sum_r w^r}$

We saw that:

- At the output point $\overline{x}$, $h_i(\overline{x}) \leq 1 + 2\epsilon \quad \forall i$.
- If $\rho = \max_{i,r} h_i(x^r)$ is the width of the algorithm, the algorithm completes in $N \leq \frac{\rho \ln m}{\epsilon^2}$ rounds.

11.2 Using Electrical Flows to find Max Flow

Given a graph $G = (V, E)$ with n vertices and m edges, we need to find the maximum flow from source vertex s to sink vertex t which satisfies the capacity constraints and conservation constraints. In order to find the max-flow using electrical flows, we make two simplifying assumptions:

1. The amount of flow to be routed through the graph, F, is already given. If F is a feasible flow, our algorithm should return a valid flow vector (flow through each edge on the graph).
2. Each edge has unit capacity, $c_e = 1 \quad \forall e \in E$.

11.2.1 Algorithm

Define Q as the set of all flows of value F from s to t. These flow vectors obey the conservation constraint but may violate capacity constraints. Q is a convex set since the convex combination of any two flow vectors in Q would also lie in Q. For any $x \in Q$, let $f_i(x)$ denote the flow on edge i of the graph. Our objective is to find a flow $x \in Q$ which satisfies the capacity constraints, *i.e.* $f_i(x) \leq 1 \; \forall i \in E$.

We can use the multiplicative weights update algorithm (Algorithm-1) to find such a flow if it exists. We can think of each edge of the graph as a resistor with some initial resistance. In round r, we set the resistance of edge i as:

$$R_i^r = y_i^r + \frac{\epsilon}{m}\sum_i y_i^r \tag{11.1}$$

and use s-t electrical flows as an oracle to find a flow x^r according to these resistances. Note that the resulting electrical flow x^r is ignorant of the capacity constraints on edges, but does obey the conservation constraints (KCL), hence $x^r \in Q$. In the subsequent rounds, resistances are increased in proportion to the amount of current flowing through the edges.

11.2.2 Electrical flow as an Oracle

Given a current source F and edge resistances R_i, the oracle computes a valid electrical flow of value F from s to t. Let x^r be the vector returned by the Oracle in round r, corresponding to which the flow values are $f_i(x^r)$. In order for electrical flows to work as a valid oracle in Algorithm-1, the following must hold:

$$\sum_i y_i^r f_i(x^r) \le \sum_i y_i^r \tag{11.2}$$

We now analyse the electrical flow oracle to see if this holds.

Cauchy-Schwartz inequality states that for any two vectors u and v,

$$\langle u, v\rangle \le \|u\|_2 \|v\|_2 \tag{11.3}$$

where $\langle \cdot, \cdot \rangle$ is the inner product. It can also be written as

$$|\langle u, v\rangle|^2 \le \|u\|_2^2 \|v\|_2^2 \tag{11.4}$$

Setting $u_i = \sqrt{y_i}$ and $v_i = f_i(x)\sqrt{y_i}$, we get from (11.4),

$$\begin{aligned}\Big[\sum_i y_i f_i(x)\Big]^2 &\le \sum_i y_i \times \sum_i y_i f_i^2(x) \\ &\le \sum_i y_i \times \sum_i R_i f_i^2(x) \qquad [\because R_i \ge y_i] \end{aligned} \tag{11.5}$$

We will use the following lemmas:

Lemma 1. *An electrical flow from s to t is the minimum-energy flow among all s-t flows of value F.*

Lemma 2. *Let x be the electrical flow returned by the oracle at any round r for which y_i are the multipliers and R_i are the corresponding edge resistances. Then the following holds:*

$$\sum_i R_i f_i^2(x) \le (1+\epsilon)\sum_i y_i$$

Proof. For a flow vector x, the term $\sum_i R_i f_i^2(x)$ represents the energy of the flow. Let $x^* \in Q$ be a flow vector which respects edge capacities. From Lemma 1, we know that electrical flow is the minimum energy flow. Therefore,

$$\begin{aligned}\sum_i R_i f_i^2(x) &\le \sum_i R_i f_i^2(x^*) \\ &\le \sum_i R_i \qquad [\because f_i(x^*) \le 1] \\ &= (1+\epsilon)\sum_i y_i \qquad \text{(from (11.1))}\end{aligned}$$

Hence proved. ■

Using Lemma 2 and (11.5), we get:

$$\Big[\sum_i y_i f_i(x)\Big]^2 \le \sum_i y_i \times (1+\epsilon) \sum_i y_i$$
$$\sum_i y_i f_i(x) \le (1+\epsilon)^{1/2} \sum_i y_i$$

11.2.3 Width of the algorithm

The width for the s-t flow feasibility algorithm is defined as:

$$\rho = \max_{i,r} f_i(x^r)$$

for flow vector x^r returned by the electrical-flow oracle in round r. We can bound the energy of flow on each edge at any round by:

$$\begin{aligned}
R_i f_i^2(x) &\ge \Big(\frac{\epsilon}{m}\sum_i y_i\Big) f_i^2(x) && \text{(from (11.1))}\\
\therefore \Big(\frac{\epsilon}{m}\sum_i y_i\Big) f_i^2(x) &\le R_i f_i^2(x) \\
&\le \sum_i R_i f_i^2(x) \\
&\le (1+\epsilon)\sum_i y_i && \text{(From Lemma 2)}\\
\therefore \frac{\epsilon}{m} f_i^2(x) &\le 1+\epsilon \\
f_i(x) &\le \sqrt{\frac{m(1+\epsilon)}{\epsilon}} \simeq \sqrt{\frac{m}{\epsilon}}
\end{aligned}$$

Since the above holds for every round, therefore, width of the algorithm is $\rho \le \sqrt{\frac{m}{\epsilon}}$.

11.2.4 Running time

We know from Sec 11.1 that for width ρ, the number of rounds in which the algorithm returns a solution $\overline{x}$ for which $f_i(\overline{x}) \le 1 + 2\epsilon$, is at most:

$$N \le \rho \frac{\ln m}{\epsilon^2}$$

Substituting the value of ρ:

$$N \le \frac{\sqrt{m}\ln m}{\epsilon^{2.5}}$$

Each round invokes a call to the oracle, which computes electrical flows. Since computation of electrical flows in a graph with m edges can be computed in $O(m)$ time using Laplacian, therefore

$$\text{Running time} \le \frac{m\sqrt{m}\ln m}{\epsilon^{2.5}}$$

11.3 Getting rid of the assumptions

Now we remove the two assumptions that we had made in the beginning, and modify the algorithm accordingly.

11.3.1 Assumption 1

Our first assumption was that instead of the max-flow problem, we were solving a feasibility problem where the task was to determine if there is a feasible s-t flow of value F. The max-flow problem can be solved by doing a binary search over the possible flow values. Since flows are integral and max-flow is equal to min-cut, for unit edge capacities, we can run the feasibility check oracle $O(\log m)$ times to find the max-flow.

11.3.2 Assumption 2

The second assumption was that all edges have unit capacities. We now show that by making the following modifications to the previous algorithm, we can can obtain the same approximation guarantees and same width for non-unit edge capacities:

1. Solve for congestion instead of flow on each edge. Define,

$$h_i(x) = \frac{f_i(x)}{c_i}$$

   as the congestion of edge i, where f_i and c_i are the flow and capacity of edge i respectively. Our modified objective is then to find a valid flow vector x for which $h_i(x) \leq 1$, *i.e.* $f_i(x) \leq c_i$.

2. For computation of electrical flows in each round, define edge resistance as:

$$R_i^r = \left(y_i^r + \frac{\epsilon}{m}\sum_i y_i^r\right)\frac{1}{c_i^2} \tag{11.6}$$

11.3.2.1 Analysis

A valid oracle should return a flow vector x for which:

$$\sum_i y_i h_i(x) \leq \sum_i y_i$$

Cauchy-Schwartz inequality gives us:

$$\begin{aligned}
\big[\sum_i y_i h_i(x)\big]^2 &\leq \sum_i y_i \times \sum_i y_i h_i^2(x) \\
&= \sum_i y_i \times \sum_i y_i \frac{f_i^2(x)}{c_i^2} \\
&\leq \sum_i y_i \times \sum_i R_i f_i^2(x) \qquad [\because R_i \geq \frac{y_i}{c_i^2}] \qquad (11.7)
\end{aligned}$$

Lemma 2 still holds, since:

$$\begin{aligned}
\sum_i R_i f_i^2(x) &\leq \sum_i R_i f_i^2(x^*) && \text{(from Lemma 1)} \\
&\leq \sum_i R_i c_i^2 && [\because f_i(x^*) \leq c_i] \\
&= (1+\epsilon)\sum_i y_i && \text{(from (11.6))}
\end{aligned}$$

Using Lemma 2 and (11.7) gives us:

$$\big[\sum_i y_i h_i(x)\big]^2 \leq \sum_i y_i \times (1+\epsilon)\sum_i y_i$$

$$\sum_i y_i h_i(x) \leq (1+\epsilon)^{1/2}\sum_i y_i$$

11.3.2.2 Width

We can bound the energy of flow on each edge at any round by:

$$R_i f_i^2(x) \geq \Big(\frac{\epsilon}{m}\sum_i y_i\Big)\frac{f_i^2(x)}{c_i^2} \qquad \text{(from (11.1))}$$

$$\therefore \Big(\frac{\epsilon}{m}\sum_i y_i\Big)\frac{f_i^2(x)}{c_i^2} \leq R_i f_i^2(x)$$

$$\leq \sum_i R_i f_i^2(x)$$

$$\leq (1+\epsilon)\sum_i y_i \qquad \text{(From Lemma 2)}$$

$$\therefore \frac{\epsilon}{m} \times \frac{f_i^2(x)}{c_i^2} \leq 1+\epsilon$$

$$h_i(x) \leq \sqrt{\frac{m(1+\epsilon)}{\epsilon}} \simeq \sqrt{\frac{m}{\epsilon}}$$

Thus, even for non-unit edge capacities, the algorithm has width $\rho \leq \sqrt{\frac{m}{\epsilon}}$.