
Lecture 23

1 Fast Max-Flow

Given undirected graph $G(V, E)$ where each edge has capacity 1, the objective is to find the maximum flow from s to t , such that the flow on an edge does not exceed its capacity. The running time for the best known max flow algorithm until recently was $O(m^{3/2})$. We discuss the algorithm from [1] for finding ϵ -approximate max-flows in time $\tilde{O}(m^{4/3}/\epsilon)$.

1.1 Electrical Flows

Finding electrical flows: Let L be the Laplacian for graph $G(V, E)$ where edge e has weight $\frac{1}{R_e}$. The potential vector ϕ for the electric flow and to the current vector i whose j -th component is the outgoing current at vertex j are related as follows,

$$L\phi = i \qquad L^+i = \phi \qquad (1)$$

The Laplacian linear system solver can be used to find the electrical flow for any i in time $\tilde{O}(m)$, by finding ϕ and then the current through each edge using Ohm's law.

Energy minimization: The energy dissipated in the network by flow f_e is $\sum_e f_e^2 R_e$, as a function of ϕ the energy is $E(\phi) = \sum_{u \sim v} \frac{(\phi_u - \phi_v)^2}{R_{uv}}$. Consider the problem of finding the minimum energy flow if some vertex potentials are fixed. The partial derivative of the energy with respect to free variables must be 0,

$$\frac{\partial E(\phi)}{\partial \phi_u} = \sum_{v \sim u} \frac{2(\phi_u - \phi_v)}{R_{uv}} = 2i_u \qquad (2)$$

The outflowing current at free vertices is 0 for the minimum energy flow that is the Kirchoff current laws are satisfied. The electrical flow is therefore the unique energy minimizing flow.

1.2 Experts for max flow

Binary search: Suppose we have an algorithm that constructs a flow of value f if one exists. The value of the max flow F^* can be found in $\log m$ iterations by binary search and the max flow can be found by running the algorithm with input F^* .

Flow game: The max flow game is played between the edge and flow players, the edge player plays edge e , the flow player plays flow f with value F and the gain for the edge player is f_e .

Suppose the flow player can guarantee an average loss $(1 + \epsilon)$ and a maximum loss of ρ against any strategy played by the edge player. This is formalized as an (ϵ, ρ) oracle,

$$\begin{aligned} \sum_e w_e f_e &\leq (1 + \epsilon) \sum_e w_e \\ f_e &\leq \rho \\ |f| &= F \end{aligned} \tag{3}$$

Then the following algorithm converges to a $(1 + O(\epsilon))$ approximate flow in $T = O(\frac{\rho \log n}{\epsilon^2})$ rounds.

1. The edge player follows the experts algorithm with initial weights $w(e) = 1$ and update rule $w_e^{t+1} = w_e^t (1 + \frac{\epsilon}{\rho} f_e^t)$.
2. The flow player plays the flow f_i output by the (ϵ, ρ) oracle against weights w_i .

CLAIM 1

The average flow $f^* = \frac{1}{T} \sum_{i \in T} f_i$ has value F and satisfies all the capacity constraints on edges within a factor of ϵ , for $T > \frac{\rho \ln n}{\epsilon^2}$.

PROOF: The oracle ensures that the gain for the edge player is at most $(1 + \epsilon)$ in each round. The gain of the best expert in retrospect against the flow f^* is $\max_e f^*(e)$. The analysis of the experts algorithm with gains yields,

$$(1 + \epsilon)T \geq G \geq \max_e f^*(e)T(1 - \epsilon) - \frac{\rho \ln m}{\epsilon} \tag{4}$$

The average flow f^* at the end of the procedure therefore satisfies all the capacity constraints up to a factor of $1 + O(\epsilon)$. \square

1.3 Implementing the oracle

The (ϵ, ρ) oracle can be implemented in time $\tilde{O}(m)$ by finding electrical flows. The running time of the ϵ algorithm is $\tilde{O}(\frac{m\rho}{\epsilon^2})$. We will first construct an oracle of width $\tilde{O}(m^{1/2})$ and then improve the width to $\tilde{O}(m^{1/3})$.

1.3.1 A width \sqrt{m} oracle

An oracle of width $\tilde{O}(m^{1/2})$ can be implemented by finding an electrical flow of value F from s to t with resistances as in the claim below,

CLAIM 2

The electrical flow with resistances $R_e = w_e + \frac{\epsilon W}{m}$ has average congestion $(1 + \epsilon)$ and maximum congestion $\tilde{O}(m^{1/2})$.

PROOF: We know that there is a feasible flow f' of value F such that $f'(e) \leq 1$ for all e . The energy of the electrical flow is bounded by,

$$\sum_e f_e^2 R_e \leq \sum_e (f'_e)^2 R_e \leq \sum_e R_e = (1 + \epsilon)W$$

The average congestion can be bounded using the Cauchy Schwarz inequality,

$$\begin{aligned} \left(\sum_e w_e f_e \right)^2 &\leq \left(\sum_e w_e f_e^2 \right) \left(\sum_e w_e \right) \\ &\leq \left(\sum_e R_e f_e^2 \right) W \leq (1 + \epsilon) W^2 \end{aligned} \quad (5)$$

Taking square roots, we have $\sum_e w_e f_e \leq \sqrt{1 + \epsilon} W < (1 + \epsilon) W$. To bound on the maximum congestion for the electrical flow, we note that the energy dissipated on an edge is at most the total energy of the electrical flow,

$$\frac{f_e^2 \epsilon W}{m} \leq (1 + \epsilon) W \Rightarrow f_e \leq \sqrt{\frac{m(1 + \epsilon)}{\epsilon}} \quad (6)$$

□

1.4 Improving the width of the oracle

The width of the oracle is improved by changing the algorithm to the following:

1. The edge player follows the experts algorithm with initial weights $w(e) = 1$ and update rule $w_e^{t+1} = w_e^t (1 + \frac{\epsilon}{\rho} f_e^t)$.
2. The flow player plays the flow f_i outputs the electrical flow with $R_e = w_e + \frac{\epsilon W}{m}$, if there is an edge with congestion more than ρ in f_i delete edge and repeat.

The running time of this algorithm is $O(\frac{\rho m \log m}{\epsilon^2} + km)$, to analyze the algorithm we need to bound k the number of edges removed and argue that the max flow does not change significantly due to removal of the edges.

1.5 Effective Resistance Lemma

The analysis relies on a lemma that bounds the change in effective resistance when an edge contributing to a β fraction of the energy of the electrical flow is removed.

LEMMA 3

Let $R(r)$ be the effective resistance between (s, t) when edges have resistances r_e ,

(i) If $r_e > r'_e$ then $R(r) \geq R(r')$.

(ii) Suppose f is an electrical flow and e an edge such that $f_e^2 r_e \geq \beta E_r(f)$. The effective resistance on removing e is at least $\frac{R}{1 - \beta}$.

PROOF: (i) For all potential vectors ϕ such that $\phi(s) = 1$ and $\phi(t) = 0$ the energy of the electrical flow corresponding to resistances r is less than the energy for resistances r' .

$$E_r(\phi) = \sum_{u \sim v} \frac{(\phi_u - \phi_v)^2}{r_{uv}} \leq \sum_{u \sim v} \frac{(\phi_u - \phi_v)^2}{r'_{uv}} = E_{r'}(\phi) \quad (7)$$

Taking the minima we have $\frac{1}{R(r)} = \min_{\phi} E_r(\phi) \leq \min_{\phi} E_{r'}(\phi) = \frac{1}{R'(r)}$.

(ii) Let ϕ be the potential vector corresponding to the electrical flow with potential difference 1 across s and t .

$$\frac{1}{R} \geq \sum_{e \in E(G) \setminus h} \frac{(\phi_u - \phi_v)^2}{r_e} + \frac{\beta}{R} \quad (8)$$

The energy of ϕ with respect to resistances r'_e is at least $\frac{1}{R'}$, therefore $\frac{(1-\beta)}{R} \geq \frac{1}{R'}$.

□

1.6 Bounding the number of edges removed

THEOREM 4

For $\rho = \tilde{O}(m^{1/3})$, the number of edges removed by the algorithm is $\tilde{O}(m^{1/3})$, the total capacity of the edges removed is $O(\epsilon F)$.

PROOF: We analyze the change in the effective resistance between (s, t) over the course of the algorithm. The effective resistance R is the energy of the electrical flow of value 1.

The capacity of the (s, t) min-cut is equal to F^* by the max-flow min-cut theorem. There must be an edge with flow $\frac{1}{F^*}$ across the min-cut so the initial energy $R(0) \geq \frac{1}{(F^*)^2}$.

The energy dissipated of an edge with congestion ρ is at least $\rho^2 R_e \geq \frac{\epsilon \rho^2 W}{m}$, while the total energy is at most $(1 + \epsilon)W$. If k edges get removed during the course of the algorithm, by part (ii) of the effective resistance lemma,

$$R(T) \geq R(0) \left(1 - \frac{\epsilon \rho^2}{m(1 + \epsilon)}\right)^{-k}$$

The energy of the flow of value F is at most $(1 + \epsilon)W(T)$, hence we have an upper bound on $R(T)$,

$$\frac{(1 + \epsilon)W(T)}{F^2} \geq R(T)$$

The increase in weight over a single iteration is bounded as follows,

$$W(t+1) = \sum_e w_e(t) \left(1 + \frac{\epsilon}{\rho} f_e^t\right) \leq W(t) \left(1 + \frac{\epsilon(1 + \epsilon)}{\rho}\right) \quad (9)$$

The weight $W(T)$ after $T = \frac{\rho \ln n}{\epsilon^2}$ rounds can be at most $e^{2 \ln n / \epsilon}$.

$$(1 + \epsilon)e^{2 \ln n / \epsilon} \geq \frac{F^2}{(F^*)^2} \left(1 - \frac{\epsilon \rho^2}{m(1 + \epsilon)}\right)^{-k} \quad (10)$$

The ratio between F^* and F can be at most m , taking logarithms we have $k \leq \frac{2 \ln m + 2 \ln n / \epsilon}{-\ln(1 - \frac{\epsilon \rho^2}{m(1 + \epsilon)})} = O\left(\frac{m \ln m}{\epsilon^2 \rho^2}\right)$.

Choosing $\rho = O(m^{1/3}(\ln m)^{1/3}/\epsilon)$ we find that at most $O((m \ln m)^{1/3})$ edges get removed. The congestion ρ on an edge can be at most F [for the unit capacity case $F/\rho > 1$], so the total capacity of the edges removed is $\leq O\left(\frac{m \ln m F}{\epsilon^2 \rho^3}\right) = O(\epsilon F)$. □

1.7 The Cut Algorithm

First a bound on the sum of potential differences across edges for an electrical flow s.t $\phi(s) - \phi(t) = 1$ using Cauchy Schwartz,

$$\sum_{e \in E} \phi(e) \leq \left(\left(\sum_e \frac{\phi_e^2}{r_e} \right) \left(\sum_e r_e \right) \right)^{1/2} = \sqrt{\frac{R}{R_{eff}}}$$

References

- [1] P. Christiano, J.A. Kelner, A. Madry, D.A. Spielman, and S.H. Teng. Electrical Flows, Laplacian Systems, and Faster Approximation of Maximum Flow in Undirected Graphs. *Arxiv preprint arXiv:1010.2921*, 2010.