Randomized Algorithms 2015A
Lecture 5 – Edge Sparsification for Cuts*

Robert Krauthgamer

1 Edge Sparsification for Cuts

Cut Sparsifier:

Input: a graph $G = (V, E)$, for simplicity we will sometimes assume unit capacities.

Goal: construct a sparse graph G' that has the same cut values, up to approximation factor $1 \pm \varepsilon$.

Applications: Smaller storage or communication (e.g., sending G to a smartphone, or receiving G_i's from different locations to compute on $G_1 + \cdots + G_k$), and potentially faster computation of min/max cut problems.

We shall actually allow parallel edges, i.e. let G be a multi-graph, and we can thus actually handle "small" weights. For two graphs on the same vertex-set, we write $G \leq G'$ if

$$\forall S \subset V, \qquad \mathrm{cap}_G(S, \bar{S}) \leq \mathrm{cap}_{G'}(S, \bar{S}).$$

Our goal will be to build G' such that $G' \in (1 \pm \varepsilon)G$, called a $(1 + \varepsilon)$-cut sparsifier.

Clique example: Suppose G is a clique on n vertices. Let G' be a random graph $G_{n,p}$ for p "sufficiently large" ($p \geq 100\varepsilon^{-2} \log n$). Prove that with high probability $G' \in (1 \pm \varepsilon)G$.

Hint: Consider the cuts $(S, \bar{S})$ by analyzing separately each $|S|$.

First attempt – subsampling:

Let's sample (i.e. keep) every edge independently with probability $p \in [0, 1]$. Denote the resulting graph $G' = (V, E')$. Consider a cut $(S, \bar{S})$, and suppose it's capacity in G is $c := \mathrm{cap}_G(S, \bar{S})$. Denote the capacity of the corresponding cut in G' by a random variable $c' := \mathrm{cap}_{G'}(S, \bar{S})$. Then

$$\mathbb{E}[c'] = pc.$$

So in expectation, cuts are preserved up to scaling by factor p. This can be "corrected" by giving every sampled edge capacity $1/p$. But is c' likely to be close to its expectation?

*These notes summarize the material covered in class, usually skipping proofs, details, examples and so forth, and possibly adding some remarks, or pointers. The exercises are for self-practice and need not be handed in. In the interest of brevity, most references and credits were omitted.

Analysis of subsampling: Using the Chernoff concentration bound,

$$\Pr[c' > (1+\varepsilon)\mathbb{E}c'] \le e^{-\varepsilon^2 pc/3}.$$

Suppose we make sure that $p \ge \frac{3d\ln n}{\varepsilon^2 \hat{c}} \ge \frac{3d\ln n}{\varepsilon^2 c}$ for some fixed $d > 0$ (say $d = 5$); then the RHS is $\le 1/n^d$. And since a similar bound applies to deviation in the other direction, we get

$$\Pr[c' \notin (1 \pm \varepsilon)\mathbb{E}c'] \le 2/n^d.$$

But is it possible to guarantee this approximation to *all* cuts? There are about 2^n such cuts....

The answer is yes, because the number of small cuts is not too large. We can then apply several "smaller" union bounds, each with number of events (cuts) that is inversely proportional to their probabilities.

Counting minimum cuts: We saw in previous weeks.

Theorem 1 [Karger 1993]: For every connected graph, the number of distinct cuts attaining the minimum value is $\le n(n-1)/2$.

Corollary 2: For every $\alpha \ge 1$ and every connected graph, the number of distinct cuts whose value is within factor α of the minimum is $\le n^{2\alpha}$.

Exer: prove corollary 2 by extending the proof of Theorem 1.

Theorem 3 [Karger]: Let G be a graph on n vertices and minimum cut capacity $\hat{c}$. Construct G' by including every edge from G with probability $1 \ge p \ge \frac{6(d+2)\ln n}{\varepsilon^2 \hat{c}}$. Then with probability $\ge 1 - O(1/n^d)$, every cut in G' has capacity within $1 \pm \varepsilon$ factor from its expectation.

Illustration: consider applying to a clique where $\hat{c} = \Theta(n^2)$, and to a graph with two cliques connected by one edge where $\hat{c} = 1$.

The proof was seen in class. The main idea is that by Corollary 2, the number of small cuts is not too large. We can then apply several "smaller" union bounds, each with number of events (cuts) that is inversely proportional to their probabilities.

Exer: Where did we use the fact that G is unweighted? What could happen with edge weights < 1? and > 1?

Exer: Let c_{max} denote the value of a maximum cut in the graph G. Prove that the expected number of edges in G' is $\Theta(\frac{1}{\varepsilon^2}\frac{c_{max}}{\hat{c}} \cdot \log n)$. (Hint: prove that $|E|/2 \le c_{max} \le |E|$ by considering a random bipartition.)

Here is a slightly more general version of this theorem.

Theorem 4 [Karger]: Let H be an n-vertex graph and $X_e \in [0, M]$ for $e \in E(H)$ be independent random variables. Let $H(X_e)$ be a random graph obtained from G by placing edge weights equal to X_e, and denote by $\tilde{c}$ the minimum expected capacity over all cuts in $H(X_e)$. Then with probability $\ge 1 - O(1/n^d)$, we have every cut in $G(X_e)$ has capacity within $1 \pm \tilde{\varepsilon}$ factor of its expectation, for $\tilde{\varepsilon} = \sqrt{2(d+2)(M/\tilde{c})\ln n}$.

Example: Given a graph G and a desired accuracy ε, we can set $p = \frac{2(d+2)\ln n}{\hat{c}\varepsilon^2}$, and let $X_e = 1/p$ with probability p, and $X_e = 0$ (i.e., non-edge) otherwise. This way, the expected capacity of a cut is just its capacity in G, in particular $\tilde{c} = \hat{c}$. Applying this theorem with $M = 1/p$, we get approximation $\tilde{\varepsilon} = \sqrt{2(d+2)(M/\hat{c})\ln n} \approx \varepsilon$, as desired. In fact, this is just the graph from previous theorem scaled by factor $1/p$.

Exer: Prove this theorem (similarly to the previous one).

The downside of the above result is that the number of edges might not decrease at all. For instance, if the initial graph is two cliques connected by a single edge, we actually need to "sample down" each clique separately (perhaps at different rates, if they have different sizes), but not the entire graph at the same rate.

We now aim to overcome this.

Theorem 5 [Benczur-Karger, 1996]: For every weighted graph $G = (V, E)$ on n vertices and error parameter $\varepsilon > 0$, there is a weighted subgraph $G' = (V, E')$ with $O(\varepsilon^{-2} n \log n)$ edges such that $G' \in (1 \pm \varepsilon)G$. Moreover, G' can be constructed in $O(|E| \log^2 n)$ time.

Remark: The number of edges was later improved to $|E'| \le O(\varepsilon^{-2} n)$ by [Batson-Spielman-Srivastava, 2009], even deterministically (and even for spectral sparsifier) but not in near-linear time computation. There is a recent matching lower bound $\Omega(\varepsilon^{-2} n)$ by [Andoni-Krauthgamer-Woodruff].

Here: We will prove a slightly weaker version, for unweighted graphs, with another $\log^2 n$ factor, and without the near-linear time algorithm.

Example: Consider the following cut problem. The input is a graph $G = (V, E)$ with k vertices $t_1, \ldots, t_k \in V$, and the goal is to find a minimum capacity $F \subseteq E$ whose removal disconnects t_i from t_j for all $i \neq j$.

Exer: Does a $(1 + \varepsilon)$ cut sparsifier G' of G approximately preserve also the optimum for this problem?

Remark: Recall that by definition, the capacity of every cut $(S, \bar{S})$ is approximately the same in G and in G'. But the optimum F in the above problem might disconnect G into more than two connected components (you might want to show such G).

Main idea: Sample edges non-uniformly, each edge e with probability p_e that is inversely proportional to its "connectivity" c_e. So "dense" regions will be sampled with smaller probability, thereby reducing the number of edges there more aggresively.

Definitions of Connectivity:

A graph is *k-connected* if every cut in it has capacity $\geq k$.

A *k-strong component* is a maximal vertex-induced subgraph that is k-connected.

Example: Consider r cliques, connected by a cycle.

Exer: Prove there is a unique partitioning of the vertices into k-strong components. (Hint: If V_1 and V_2 are k-connected and have non-empty intersection, then also $V_1 \cup V_2$ is k-connected.)

It follows that the k-strong components partition the vertices of the graph. They can be obtained by removing (simultaneously) all cuts of capacity $< k$. Clearly, the $(k+1)$-strong components is a refinement of that partition. Indeed, start at $k = 1$ with the whole graph being 1-strong (assuming it is connected and unweighted); now gradually increase k, say to $k = 2$, "apply" cuts of value $< k$ (if any) to break the graph into 2-strong components, and so forth. In our example, each clique is 2-strong (but also 3-strong etc.).

The *strong connectivity of an edge* $e \in E$, denoted c_e, is the maximum value k such that it is contained in a k-strong component. An edge is called *k-strong* if its strong connectivity is at least k; otherwise *k-weak*.

Note that strong connectivity differs from the usual definition of edge-connectivity. (Example: n parallel paths between s, t.)

Construction of sparsifier G':

Set $q = q_\varepsilon := 4(d+2)\varepsilon^{-2} \ln n$, and sample every $e \in E$ with probability $p_e = \min\{q/c_e, 1\}$, in which case it is given weight $1/p_e$.

Lemma 6: With probability $\geq 1 - O(1/n^d)$, the resulting graph G' has $O(qn)$ edges.

Idea of Proof: The idea is that "regions" with with high connectivity will have many edges and vice versa, so these things balance out. We saw in class shows that

$$\mathbb{E}[|E(G')|] = \sum_{e \in E} p_e \leq q \sum_{e \in E} (1/c_e).$$

The high-probability bound then follows by a Chernoff bound.

Exer: Complete the high-probability bound using Chernoff.

Lemma 7: With high probability, $G' \in (1 \pm \varepsilon \log |E|)G$.

Note that Lemmas 6,7 together indeed prove (a weaker version of) Theorem 5, by simply using a smaller value $\varepsilon' = \varepsilon / \log |E|$.

Idea of Proof: The proof seen in class uses uniform sampling (Theorem 4). The idea is to describe the same algorithm in a different way, as if we divide the sampling process into phases, and phase $i = 0, 1, \ldots$ flips the coins only for edges e with $2^i \leq c_e < 2^{i+1}$. The analysis then applies Theorem 6 separately on each 2^i-strong component, which means that we basically "remove" edges with $c_e < 2^i$. Inside each such component, the edges from level i are chosen at random, but all edges with $c_e \geq 2^{i+1}$ are kept (deterministically).

As seen in class, decompose G into edge-disjoint graphs G_i for $i = 0, 1, \ldots$, where $e \in G_i$ if $2^i \leq c_e < 2^{i+1}$. For sake of analysis, at each phase i we actually consider the graph $G_{\geq i} = \cup_{j \geq i} G_j$; notice it consists (exactly) of all 2^i-strong components.

At phase i, we sample edges of $G_{\geq i}$ as follows:

$$X_e^{(i)} = \begin{cases} 1/p_e \text{ w.p. } p_e, \text{ and } 0 \text{ otherwise} & \text{if } e \in G_i; \\ 1 & \text{otherwise (i.e. } e \in G_{\geq i+1}) \end{cases}$$

Recall that $p_e = \min\{q/c_e, 1\}$. (Edges of levels lower than i are not touched or considered at all.)

For each 2^i-strong component H, we apply Theorem 4 to $H(X_e^{(i)})$, and obtain that WHP

$$H(X_e^{(i)}) \in (1 \pm \sqrt{2(d+2)(2/q)\ln n})H = (1 \pm \varepsilon)H.$$

(If $M = 1$ then $\frac{q}{c_e} \geq \frac{q}{2^{i+1}} > 1$, hence $p_e = 1$ and surely $H(X_e^{(i)}) = H$, which is even stronger.) By a union bound over the disjoint 2^i-strong components H, WHP $G_{\geq i}(X_e^{(i)}) \in (1 \pm \varepsilon)G_{\geq i}$.

Finally, we consider the entire graph G, incurring an error of ε at each level i (notice all $1 \leq c_e \leq |E|$):

$$\begin{aligned} G' = \sum_{i=0}^{\log|E|} G_i(X_e^{(i)}) &= \sum_i \big(G_{\geq i}(X_e^{(i)}) - G_{\geq i+1}\big) \\ &\in \sum_i \big((1\pm\varepsilon)G_{\geq i} - G_{\geq i+1}\big) = \sum_i \big((1\pm\varepsilon)G_i + (1\pm\varepsilon-1)G_{\geq i+1}\big) \\ &\in (1 \pm \varepsilon \log|E|)G. \end{aligned}$$

Exer: It is sometimes easier/faster to compute an approximation to c_e. So suppose we use in p_e an approximation to c_e, say within factor 3, i.e., values $c'_e \in [c_e, 3c_e]$. Explain how the theorem and analysis shown in class would extend.

Extensions and open problems:

Hypergraphs: The analysis above can be extended to hypergraphs, giving an upper bound of $O(n^2)$ [Kogan-Krauthgamer]. The lower bound we know is $\Omega(n)$, so there is a gap. (In fact, there are also other ways to "extend" the definition from graphs to hypergraphs.)

Uncut: Can we sparsify G and maintain (for every cut) the number of uncut edges? This problem is equivalent to a system of Boolean linear equations $x_i \oplus x_j = 0$.

2-SAT: Can we sparsify a 2-SAT formula to $\tilde{O}(n)$ clauses, such that the value under *every* truth assignment is maintained within approximation factor $1 \pm \varepsilon$?

High-probability answers: Consider a weaker requirement, for a graph G' such that for every cut $(S, \bar{S})$, with high probability

$$\text{cap}_{G'}(S, \bar{S}) \in (1 \pm \varepsilon)\,\text{cap}_G(S, \bar{S}).$$

Notice the difference that here "for each cut, with high probability ..." rather than "with a high probability, for all cuts ..." (called "for each" instead of "for all" in compressive sensing literature). Is it possible to obtain G' with $\tilde{O}(n/\varepsilon)$ edges? A positive evidence (but not a graph G') is given by [Andoni-Krauthgamer-Woodruff].