Advanced Algorithms	27 November 2018

Lecture 11: Graph Sparsification II

Lecturer: Mohsen Ghaffari — *Scribe: Davin Choo*

In the previous lecture, we introduced graph sparsification as a way to obtain a subgraph with fewer edges but similar pairwise distances. In this lecture, we will look at *preserving cuts*.

1 Preserving cuts

Definition 1 (Cut and minimum cut)**.** *Consider a graph $G = (V, E)$.*

- *For $S \subseteq V, S \neq \emptyset, S \neq V$, $C_G(S, V \setminus S) = \{(u, v) : u \in S, v \in V \setminus S\}$ is a non-trivial cut in G*
- *Define cut size $E_G(S, V \setminus S) = \sum_{e \in C_G(S, V \setminus S)} w(e)$*
  For unweighted G, $w(e) = 1$ for all $e \in E$, so $E_G(S, V \setminus S) = |C_G(S, V \setminus S)|$
- *Minimum cut size of the graph G is denoted by $\mu(G) = \min_{S \subseteq V, S \neq \emptyset, S \neq V} E_G(S, V \setminus S)$*
- *A cut $C_G(S, V \setminus S)$ is said to be minimum if $E_G(S, V \setminus S) = \mu(G)$*

Given an undirected graph $G = (V, E)$, our goal in this lecture is to construct a weighted graph $H = (V, E')$ with $E' \subseteq E$ and weight function $w : E' \to \mathbb{R}^+$ such that

$$(1 - \epsilon) \cdot E_G(S, V \setminus S) \leq E_H(S, V \setminus S) \leq (1 + \epsilon) \cdot E_G(S, V \setminus S)$$

for every $S \subseteq V, S \neq 0, S \neq V$. Recall Karger's random contraction algorithm [Kar93][1]:

Algorithm 1 RandomContraction($G = (V, E)$)

```
while |V| > 2 do
    e ← Pick an edge uniformly at random from E
    G ← G/e                                        ▷ Contract edge e
end while
return The remaining cut                           ▷ This may be a multi-graph
```

Theorem 2. *For a fixed minimum cut S^* in the graph,* RandomContraction *returns it with probability $\geq 1/\binom{n}{2}$.*

Proof. Fix a minimum cut S^* in the graph. Suppose $|S^*| = k$. To successfully return S^*, *none* of the edges in S^* must be selected in the whole contraction process.

By construction, there will be $n - i$ vertices in the graph at step i of RandomContraction. Since $\mu(G) = k$, each vertex has degree $\geq k$ (otherwise that vertex itself gives a cut smaller than k), so there are $\geq (n - i)k/2$ edges in the graph. Thus,

$$\begin{aligned}
\Pr[\text{Success}] &\geq (1 - \tfrac{k}{nk/2}) \cdot (1 - \tfrac{k}{(n-1)k/2}) \cdot (1 - \tfrac{k}{(n-2)k/2}) \cdot \dots \cdot (1 - \tfrac{k}{4k/2}) \cdot (1 - \tfrac{k}{3k/2}) \\
&= (1 - \tfrac{2}{n}) \cdot (1 - \tfrac{2}{n-1}) \cdot (1 - \tfrac{2}{n-2}) \cdot \dots \cdot (1 - \tfrac{2}{4}) \cdot (1 - \tfrac{2}{3}) \\
&= (\tfrac{n-2}{n}) \cdot (\tfrac{n-3}{n-1}) \cdot (\tfrac{n-4}{n-2}) \cdot \dots \cdot (\tfrac{2}{4}) \cdot (\tfrac{1}{3}) \\
&= \tfrac{2}{n(n-1)} \\
&= 1/\tbinom{n}{2}
\end{aligned}$$

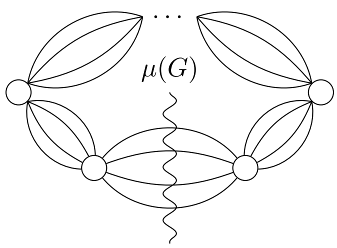
□

Corollary 3. *There are $\leq \binom{n}{2}$ minimum cuts in a graph.*

Proof. Since RandomContraction successfully produces *any given minimum cut* with probability $\geq 1/\binom{n}{2}$, there can be at most $\binom{n}{2}$ many minimum cuts. □

[1] Also, see https://en.wikipedia.org/wiki/Karger%27s_algorithm

Remark There exists (multi-)graphs with $\binom{n}{2}$ minimum cuts: Consider a cycle where there are $\frac{\mu(G)}{2}$ edges between every pair of adjacent vertices.



In general, we can bound the number of cuts that are of size at most $\alpha \cdot \mu(G)$ for $\alpha \geq 1$.

Theorem 4. *In an undirected graph, the number of α-minimum cuts is less than $n^{2\alpha}$.*

Proof. See Lemma 2.2 and Appendix A (in particular, Corollary A.7) of a version[2] of [Kar99]. □

1.1 Warm up: $G = K_n$

Consider the following procedure to construct H:

1. Let $p = \Omega(\frac{\log n}{n})$
2. Independently put each edge $e \in E$ into E' with probability p
3. Define $w(e) = \frac{1}{p}$ for each edge $e \in E'$

One can check[3] that this suffices for $G = K_n$.

1.2 Uniform edge sampling

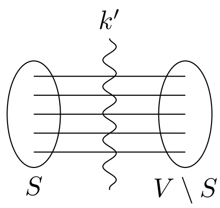
For a graph G with minimum cut size $\mu(G) = k$, consider the following procedure to construct H:

1. Set $p = \frac{c \log n}{\epsilon^2 k}$ for some constant c
2. Independently put each edge $e \in E$ into E' with probability p
3. Define $w(e) = \frac{1}{p}$ for each edge $e \in E'$

Theorem 5. *With high probability, for every $S \subseteq V, S \neq \emptyset, S \neq V$,*

$$(1-\epsilon) \cdot E_G(S, V \setminus S) \leq E_H(S, V \setminus S) \leq (1+\epsilon) \cdot E_G(S, V \setminus S)$$

Proof. Fix an arbitrary cut $(S, V \setminus S)$. Suppose $E_G(S, V \setminus S) = k' = \alpha \cdot k$ for some $\alpha \geq 1$.



Let X_e be the indicator for the edge $e \in C_G(S, V \setminus S)$ being selected into E'. By construction, $\mathbb{E}[X_i] = \Pr[X_i = 1] = p$. Then, by linearity of expectation, $\mathbb{E}[|C_H(S, V \setminus S)|] = \sum_{e \in C_G(S, V \setminus S)} \mathbb{E}[X_i] = k'p$. As we put $1/p$ weight on each edge in E', $\mathbb{E}[E_H(S, V \setminus S)] = k'$. Using Chernoff bound, for sufficiently large c, we get:

$$\begin{aligned}
& \Pr[\text{Cut } (S, V \setminus S) \text{ is badly estimated in } H] & \\
= \; & \Pr[|E_H(S, V \setminus S) - \mathbb{E}[E_H(S, V \setminus S)]| > \epsilon \cdot k'] & \text{What it means to be badly estimated} \\
\leq \; & 2e^{-\frac{\epsilon^2 k' p}{3}} & \text{Chernoff bound} \\
= \; & 2e^{-\frac{\epsilon^2 \alpha k p}{3}} & \text{Since } k' = \alpha k \\
\leq \; & n^{-10\alpha} & \text{For sufficiently large } c
\end{aligned}$$

[2]Version available at: http://people.csail.mit.edu/karger/Papers/skeleton-journal.ps

[3]Fix a cut, analyze, then take union bound.

Using Theorem 4 and union bound over all possible cuts in G,

$$\begin{array}{lll} & \Pr[\textit{Any} \text{ cut is badly estimated in } H] & \\ \leq & \int_1^\infty n^{2\alpha} \cdot \frac{1}{n^{-10\alpha}} d\alpha & \text{From Theorem 4 and above} \\ \leq & n^{-5} & \text{Loose upper bound} \end{array}$$

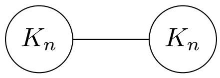
□

Theorem 6. *[Kar94] For a graph G, consider sampling every edge independently with probability p_e into E', and assign weights $1/p_e$ to each edge $e \in E'$. Let $H = (V, E')$ be the sampled graph and suppose $\mu(H) \geq \frac{c \log n}{\epsilon^2}$, for some constant c. Then, with high probability, every weighted cut size in H is (well-estimated) within $(1 \pm \epsilon)$ of the original cut size in G.*

Theorem 6 can be proved by using a variant of the earlier proof. Interested readers can see Theorem 2.1 of [Kar94].

1.3 Non-uniform edge sampling

Unfortunately, uniform sampling does not work well on graphs with small minimum cut.



Running the uniform edge sampling will not sparsify the above dumbbell graph as $\mu(G) = 1$ leads to large sampling probability p.

Before we describe a non-uniform edge sampling process [BK96], we first define *k-strong components*.

Definition 7 (k-connected)**.** *A graph is k-connected if the value of each cut of G is at least k.*

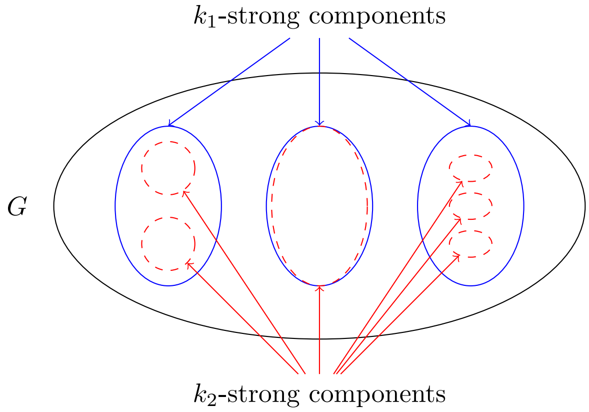
Definition 8 (k-strong component)**.** *A k-strong component is a maximal k-connected vertex-induced subgraph. For an edge e, define its strong connectivity / strength k_e as the maximum k such that e is in a k-strong component. We say an edge is k-strong if $k_e \geq k$.*

Remark The (standard) connectivity of an edge e is the minimum cut size that separates its endpoints. In particular, an edge's strong connectivity is no more than the edge's (standard) connectivity since a cut size of k implies there is no k-connected component containing both endpoints.

Lemma 9. *The following holds for k-strong components:*

1. *k_e is uniquely defined for every edge e*
2. *For any k, the k-strong components are disjoint.*
3. *For any 2 values k_1, k_2 $(k_1 < k_2)$, k_2-strong components are a* refinement *of k_1-string components*
4. *$\sum_{e \in E} \frac{1}{k_e} \leq n - 1$*
   Intuition: If there are a lot of edges, then many of them have high strength.

Proof.



1. By definition of maximum
2. Suppose, for a contradiction, there are two intersecting k-strong components. Since their union is also k-strong, this contradicts the fact that they were maximal.
3. For $k_1 < k_2$, a k_2-strong component is also k_1-strong, so it is a subset of some k_1-strong component.
4. Consider a minimum cut $C_G(S, V \setminus S)$. Since $k_e \geq \mu(G), \forall e \in C_G(S, V \setminus S)$, these edges contribute $\leq \mu(G) \cdot \frac{1}{k_e} \leq \mu(G) \cdot \frac{1}{\mu(G)} = 1$ to the summation. Remove these edges from G and repeat the argument on any remaining connected components. Since each cut removal contributes at most 1 to the summation and the process stops when we reach n components, $\sum_{e \in E} \frac{1}{k_e} \leq n - 1$.

□

For a graph G with minimum cut size $\mu(G) = k$, consider the following procedure to construct H:

1. Set $q = \frac{c \log n}{\epsilon^2}$ for some constant c
2. Independently put each edge $e \in E$ into E' with probability $p_e = \frac{q}{k_e}$
3. Define $w(e) = \frac{1}{p_e} = \frac{k_e}{q}$ for each edge $e \in E'$

Lemma 10. $\mathbb{E}[|E'|] \leq \mathcal{O}(\frac{n \log n}{\epsilon^2})$

Proof. Let X_e be the indicator whether edge e was selected into E'. By construction, $\mathbb{E}[X_e] = \Pr[X_e = 1] = p_e$. Then,

$$\begin{array}{rcll} \mathbb{E}[|E'|] & = & \mathbb{E}[\sum_{e \in E} X_e] & \text{By definition} \\ & = & \sum_{e \in E} \mathbb{E}[X_e] & \text{Linearity of expectation} \\ & = & \sum_{e \in E} p_e & \text{Since } \mathbb{E}[X_e] = \Pr[X_e = 1] = p_e \\ & = & \sum_{e \in E} \frac{q}{k_e} & \text{Since } p_e = \frac{q}{k_e} \\ & = & q(n-1) & \text{Since } \sum_{e \in E} \frac{1}{k_e} \leq n - 1 \\ & \in & \mathcal{O}(\frac{n \log n}{\epsilon^2}) & \text{Since } q = \frac{c \log n}{\epsilon^2} \text{ for some constant } c \end{array}$$

□

Remark One can apply Chernoff bounds to argue that $|E'|$ is highly concentrated around its expectation.

Theorem 11. *With high probability, for every $S \subseteq V, S \neq \emptyset, S \neq V$,*

$$(1 - \epsilon) \cdot E_G(S, V \setminus S) \leq E_H(S, V \setminus S) \leq (1 + \epsilon) \cdot E_G(S, V \setminus S)$$

Proof. Let $k_1 < k_2 < \cdots < k_s$ be all possible strength values in the graph. Consider G as a weighted graph with edge weights $\frac{k_e}{q}$ for each edge $e \in E$, and a family of unweighted graphs $F_1, \ldots, F_s$ where $F_i = (V, E_i)$ where $E_i = \{e \in E : k_e \geq k_i\}$. Observe that:

- $s \leq |E|$ since each edge has only 1 strength value
- By construction of F_i's, if an edge e has strength i in F_i, $k_e = i$ in G
- $F_1 = G$
- For each i, F_{i+1} is a subgraph of F_i
- By defining $k_0 = 0$, one can write $G = \sum_{i=1}^{s} \frac{k_i - k_{i-1}}{q} F_i$. This is because an edge with strength k_i will appear in $F_i, F_{i-1}, \ldots, F_1$ and the terms will telescope to yield a weight of $\frac{k_i}{q}$.

The sampling process in G directly translates to a sampling process in each graph in $\{F_i\}_{i \in [s]}$ — When we add an edge e into E', we also add it to the edge sets of $F_{k_e}, \ldots, F_s$. For each $i \in [s]$, Theorem 6 tells us that every cut in F_i is well-estimated with high probability. Then, a union bound over $\{F_i\}_{i \in [s]}$ will tell us that any cut in G is well-estimated with high probability. □

References

[BK96] András A Benczúr and David R Karger. Approximating st minimum cuts in õ (n 2) time. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 47–55. ACM, 1996.

[Kar93] David R Karger. Global min-cuts in rnc, and other ramifications of a simple min-cut algorithm. In *SODA*, volume 93, pages 21–30, 1993.

[Kar94] David R Karger. Random sampling in cut, flow, and network design problems. In *Proc. of the Symp. on Theory of Comp. (STOC)*, pages 648–657, 1994.

[Kar99] David R Karger. Random sampling in cut, flow, and network design problems. *Mathematics of Operations Research*, 24(2):383–413, 1999.