Succinct Graph Structures and their Applications **Spring 2018**

Lecture 8: May 31

Lecturer: Merav Parter

Cut Sparsifiers

A *cut* is a partition of the vertices in the graph into two groups S and $V \setminus S$. The *value* of the cut, denoted by $\delta_G(S)$ is the number of edges in $E_G(S, V \setminus S) = \{(u, v) \in E(G) \mid u \in S \text{ and } v \notin S\}$. For a weighted graph G, the cut value $\delta_G(S)$ is the sum of all edges weights in $E_G(S, V \setminus S)$. When the graph G is clear from the context, we may omit it and simply write $\delta(S)$. We sometimes abuse notation and when saying cut we actually refer to the edges that cross the cut $E_G(S, V \setminus S)$. The min-cut of G denoted by $\texttt{min-cut}(G)$ is the minimum cut value over all possible cuts, i.e., $\texttt{min-cut}(G) = \min_{S \subset V} \delta_G(S)$. For a vertex pair $u, v \in V$, $\texttt{min-cut}(u, v, G) = \min_{S \subset V, u \in S, v \notin S} \delta_G(S)$, the min-cut value that separates u and v in G.

The goal of this class is to compute a sparse *weighted* subgraph $H \subseteq G$ that *approximates* well (i.e., up to $(1 \pm \epsilon)$ factor) the value of *every* cut in G. The edges of this graph H will be taken from G, however, we will have the freedom to assign weights to the selected edges from G, and these weights will compensate for the removal of some edges in G.

Definition 8.1 ($(1+\epsilon)$ Cut Preservers) *Given an unweighted undirected graph $G = (V, E)$ and $\epsilon \in (0, 1)$, a weighted subgraph $H = (V, E', W)$ for $E' \subseteq E$ is an $(1 + \epsilon)$ cut preserver for G, if for every $S \subset V$,*

$$(1 - \epsilon)\delta_H(S) \leq \delta_G(S) \leq (1 + \epsilon)\delta_H(S).$$

For brevity, we write $\delta_H(S) \in (1 \pm \epsilon)\delta_G(S)$.

We will describe two constructions of cut sparsifiers. The first construction uses the *uniform sampling* approach and yields a preserver with $\widetilde{O}(|E_G|/c)$ edges where c is the value of the min-cut in G. The second construction uses the non-uniform sampling (i.e., where each edge $e \in G$ is sampled with probability p_e) and yields a preserver with $\widetilde{O}(n/\epsilon^2)$ edges.

Cut Sparsifier via Uniform Sampling

For a given probability value p and a graph G, let $G[p]$ be a subgraph of G obtained by sampling (independently) each edge $e \in E(G)$ with probability p. Throughout, we consider an n-vertex unweighted graph $G = (v, E)$ with min-cut $c = \texttt{min-cut}(G)$ where $c = \Omega(\log n)$.

The construction of the sparsifier is based on the following key lemma.

Lemma 8.2 ([Kar99]) *Given a graph G with min-cut c, let $p = \Theta(\frac{\log n}{\epsilon^2 \cdot c})$. Then, w.h.p., all cuts in $G[p]$ are concentrated within $(1 \pm \epsilon)$ factor of their expectation.*

Before proving the lemma, we first show how it is used to yield the sparsifier.

Corollary 8.3 *Let H be a subgraph obtained by sampling each edge with probability $p = \Theta(\frac{\log n}{\epsilon^2 \cdot c})$, and if sampled, its weight in H is $1/p$. Denote this subgraph by $H = 1/p \cdot G[p]$. Then, w.h.p., H is $(1 + \epsilon)$ cut sparsifier.*

Proof: By Lemma 8.2, w.h.p., for every $S \subset V$, it holds that $\delta_{G[p]}(S) \in (1 \pm \epsilon) \cdot p \cdot \delta_G(S)$. Thus, $\delta_H(S) \in (1 \pm \epsilon) \cdot 1/p \cdot p \cdot \delta_G(S)$, as required. ■

We now turn to prove Lemma 8.2. To warm up, lets consider first the minimum cut of size $c = \Omega(\log n)$ in G. Let $C = E_G(S, V \setminus S)$ be the edge set that crossed the cut where $|C| = c$. The probability that no edge in C is sampled into $G[p]$ is $(1-p)^c \leq e^{-pc} = e^{-\Omega(\log n)}$. Hence, w.h.p. at least one edge from C is sampled. We next show that w.h.p. the number of sampled C edges in $G[p]$ is concentrated around its expectation. For every $e_i \in C$, let X_i be the indicator random variable for the event that e_i is sampled into $G[p]$ and let $X = \sum_{e_i \in C} X_i$. By Chernoff, we have that:

$$\Pr[X \notin (1 \pm \epsilon)\mathbb{E}(X)] \leq 2e^{-\epsilon^2 \mathbb{E}(X)/3} = 2e^{-\epsilon^2 p \cdot c/3} = 1/poly(n) \ . \tag{8.1}$$

Hence, by the above argument, we can show that for every cut S', w.h.p., the value of the cut of S' is concentrated around its expectation in H. The problem is that there are exponentially many cuts and hence we cannot simply apply a union bound over all cuts. That is, Chernoff bound provides a polynomially good success guarantee for a *single* cut but there are 2^n many cuts. To overcome this, we will divide the cuts in G into classes based on their value. As we are going to see, there are relatively few cuts (polynomially many) whose value is c (the min-cut). As the cut size increases, the number of cuts with that particular size increases, but so does the success guarantee provided by the Chernoff bound. This would allow us to obtain a high probability guarantee for each cut class, and by a union bound also for all classes.

Lemma 8.4 *[Key Lemma from [Kar93]] Let G be an n-vertex graph with min-cut c. For every $\alpha \geq 1$, the number of cuts of size at least $\alpha \cdot c$ is at most $n^{O(\alpha)}$.*

We now complete the proof of Lemma 8.2.

Proof: Consider one particular cut of size $\alpha \cdot c$. By applying Chernoff bound as above, the probability that the cut is *not* concentrated within $(1 \pm \epsilon)$ factor of its expectation is at most $e^{-\epsilon^2 \cdot \alpha \cdot c \cdot p} = 1/n^{\Theta(\alpha)}$, which holds by plugging $p = \Theta(\frac{\log n}{\epsilon^2 \cdot c})$. Since there are $n^{O(\alpha)}$ many cuts of size αc by Lemma 8.4, upon setting the constants in the big $O(.)$ notation, we get that w.h.p. (say $1 - 1/n^5$) all cuts of size $\alpha \cdot c$ are concentrated around their expectation. The total probability of deviation from expectation is then given by:

$$\int_1^\infty n^{O(\alpha)} \cdot e^{-\Theta(\alpha \cdot \log n)} d\alpha \leq \int_1^\infty 1/n^{10\alpha} d\alpha \leq 1/n^5.$$

■

Next, we state (without a proof) a stronger version of Lemma 8.2 that will become useful in our improved sparsifier construction.

Lemma 8.5 *For every edge $e \in G$, let p_e be a sampling probability for an edge e and let $G[p_e]$ be the subgraph of G obtained by sampling each $e \in G$ with probability p_e. Then, if for each cut C in G it holds that the expected number of sampled edges in $G[p_e]$ is $\sum_{e \in C} p_e = \Omega(\log n/\epsilon^2)$, then w.h.p.* all *cuts are concentrated within $(1 \pm \epsilon)$ factor of their expectation.*

Intuitively, if we sampled $\Omega(\log n/\epsilon^2)$ many edges in expectation from each cut into $G[p_e]$, then by Chernoff, we get that w.h.p. a given fixed cut is concentrated. So Chernoff is good for showing concentration for one fixed cut! The lemma then states that this high-probability guarantee (which holds for each individual cut), actually holds over all the (exponentially many) cuts in G. As we said before, this is highly non-trivial as we cannot just employ a union bound, but rather need to use this fine-grained counting of cuts similarly to the proof of Lemma 8.2.

Improved Sparsifier via Non-uniform Sampling [BK96]

So-far, we obtained a sparsifer with $\widetilde{O}(|E|/(\epsilon^2 c)$ edges for a graph $G = (V, E)$ with min-cut c. This result is unsatisfactory when c is small but the graph G is dense. Consider for example the dumbbell graph that consists of two $K_{n/2}$ cliques connected via an edge. Such a graph as min cut 1 but $\Omega(n^2)$ edges. To sparisfy the graph further the idea is to use a non-uniform sampling, where edges are sampled with different probabilities into the sparsifier, depending on the density of the region to which the edge belongs. That is, in the dumbbell graph, we would like to keep the single connecting edge between the cliques, but to sample edges much more aggressively inside each clique. We will define for each edge e, a parameter k_e which measures its importance. If e participates in sparse regions in the graph, k_e will be large and otherwise small. The sampling probability p_e will be set to $p_e = \Theta(\frac{\log n}{\epsilon^2 k_e})$. By sampling each edge e into the sparsifier H w.p. p_e, and taking its weight to be $1/p_e$, we guarantee that all cuts are preserved *in expectation*. The main goal would be to show that this holds w.h.p. as well. This approach is based on the notion of *strong connectivity*.

Definition 8.6 (Connectivity and Strong Connectivity) *A subgraph G' is k-connected if its min-cut value is at least k. A k-*strong component *of G is a maximal k-connected subgraph of G. The* strong *connectivity of an edge e denoted by k_e is the maximum value k such that there is a k-strong connected component containing e. Finally, the connectivity of $e = (u, v)$ (in the standard notion of connectivity) denoted by λ_e is the minimum cut vaue between u and v in G. I.e., $\lambda_e(G) = \texttt{min-cut}(u, v, G)$.*

We proceed by stating properties of the strong connectivity and its relation to the standard notion of connectivity.

Observation 8.7 *For every e, $k_e(G) \leq \lambda_e(G)$.*

Proof: Consider the maximal induced subgraph $G' = G[U]$ for some vertex-set $U \subseteq V$ such that G' is k_e strong component and $e \in G'$. By definition, every cut in G' has at least k_e edges, and thus:

$$\texttt{min-cut}(u, v, G) \geq \texttt{min-cut}(u, v, G') \geq k_e \ .$$

As illustrated in Fig. 8.1, the gap between these values, k_e and λ_e is $\Omega(n)$. ■

Another straightforward observation is that if an edge e belongs to a cut in G of size k, then $k_e \leq k$. This is because $k_e \leq \lambda_e \leq k$. The next lemma is the key structural property of *strong* connectivity, it would shed some light into why we use this particular definition.

Definition 8.8 (k-strong) *An edge e is k-strong if $k_e \geq k$ and otherwise it is k-weak.*

Lemma 8.9 *Let G' be the graph G obtained by removing all edges $e \in G$ which are k-weak. Then the strong-connectivity of all the k-strong edges in G' is unharmed. That is $k_e(G) = k_e(G')$ for every edge e that is k-strong.*

Proof: The proof is really immediate from the definition of strong connectivity. If $k_e(G) \geq k$, then e belongs to an induced subgraph $G[U]$ with connectivity at least k. By definition all edges of $G[U]$ are k-strong and hence the entire subgraph $G[U]$ is in G', the lemma follows. ■

Note that the above lemma does not hold for the standard notation of edge connectivity. Consider again Fig. 8.1. The edge connectivity $\lambda_{(s,t)}(G) = \Omega(n)$ while $\lambda_{e'}(G) = 2$ for any other edge $e' \neq (s, t) \in G$. Removing all edges of lower connectivity from G, would reduce the connectivity of (s, t) from $\Omega(n)$ to 1. Strong connectivity was defined in this particular way for that purpose.

Equipped with the definition of strong edge connectivity and with its structural properties, we are now ready to describe the construction of the sparsifier.

The Improved Construction: Sample each edge e into H with probability $p_e = q/k_e$ where $q = d \log n/\epsilon^2$ for some constant d, if e is taken, then set its weight to $w_e = 1/p_e$.

Note that in the previous construction, $p_e = p = q/c$ for every e. We proceed by analyzing the size of the sparsifier.

Lemma 8.10 (Size) *W.h.p., H has $\widetilde{O}(n/\epsilon^2)$ edges.*

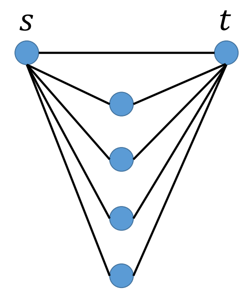

Figure 8.1: An example where $\lambda_e(G) = \Omega(n)$ but $k_e(G) = 2$ for $e = (s,t)$.

Proof: We first show that $\sum_e 1/k_e \leq n-1$. To bound the sum of k_e values, we consider an iterative process where in every step j, we show that $\sum_{e \in E_j} 1/k_e \leq 1$ for a *disjoint* subset of edges $E_j \subseteq E$ (where $E_j \cap E_{j'} = \emptyset$). We will then argue that this process has at most $n-1$ many steps, thus yielding the claim.

We start by picking E_1 to be the edges of the min-cut C in G. For every $e \in C$, it holds that $k_e = \lambda_e = |C|$. Hence, summing over all edges in C, we get that $\sum_{e \in C} 1/k_e = 1$. Next, we remove all the edges of C from G and do as follows for every connected component G' in $G \setminus C$ (since C was a min-cut, we actually have only two connected components when removing C). Let C' be the edges that cross the min-cut in G' where $|C'| = k'$. The key observation is that G' is a k'-strong connected component. The reason is that there is no edge of C that has both its endpoints in G'. In other words, when removing edges of a min-cut $C = E(S, V \setminus S)$, one connected component is fully contained in S, while the other component is fully contained in $V \setminus S$. We thus have that $k_e \geq k'$ for every $e \in C'$, which implies that $\sum_{e \in C'} 1/k_e \leq 1$. We continue in this way, each time removing the edges of the current min-cut in the given connected component (which is in fact an induced subgraph of G), and increase the "counter" by at most 1. Since in each such step, the number of connected components grows by 1, this can be repeated for at most $n-1$ steps, and hence $\sum_e 1/k_e \leq n-1$. Finally, since H has $\sum_e p_e$ edges in expectation, by Chernoff, we get that H has $\widetilde{O}(q \cdot n)$ edges w.h.p. ■

We proceed by showing that H is an $(1+\epsilon)$ sparsifier. For that purpose, we would like to use Lemma 8.5 which handles graphs obtained by a non-uniform sampling. If all edges of H were given the same weight, say, s, then by showing that the cuts of $G[p_e]$ are concentrated, we would get immediately that all cuts in $H = s \cdot G[p_e]$ are concentrated. Unfortunately, the edge weights of H might be highly non-uniform. In particular, to preserve the value of all cuts in expectation, we had to scale each edge e by assigning it a weight weight $1/p_e$. To overcome this non-uniformity of the edge weights, the analysis decomposes H into several subgraphs of uniform weights.

Let $G_w = (V(G), E(G), W)$ be a weighted graph consisting of all G edges, where each edge e has a weight $w_e = 1/p_e$. Hence, our sparsifier H is exactly $G_w[p_e]$. We now write G_w as a summation of uniformly weighted graphs as follows. Let $k_1 < k_2 < \ldots < k_{m'}$ be the sorted values of the strong edge connectivities in G. For every $i \in \{1, \ldots, m'\}$, define $F_i \subseteq G$ as the set of all edges e with strong connectivity $k_e \geq k_i$. That is, F_i consists of all the k_i-strong edges of G. We can then write G_w as the following sum of graphs:

$$G_w = \sum_i \frac{k_i - k_{i-1}}{q} \cdot F_i \ .$$

(In this notation, we treat G as a vector of length $|E(G)|$, where the j^{th} coordinate contains the weight of the j^{th} edge in G.) We next show that this equality indeed holds. Consider an edge e with $k_e = k_i$. Such an edge appears in $F_1, \ldots, F_i$ and thus, its total weight in the subgraph of the right hand-side is:

$$k_1/q + (k_2 - k_1)/q + \ldots + (k_i - k_{i-1})/q = k_i/q = w_e \ .$$

Our goal now is to show that all cuts in $H = G_w[p_e]$ are concentrated around their expectation. To do that, we will translate the sampling procedure in G_w to a sampling procedure in the F_i subgraphs. We sample each edge e in G_w with probability p_e. If the edge is taken (the coin flip came up head), then e is included in all subgraphs $F_1, \ldots, F_i$ where $k_e = k_i$. We denote all the sampled edges in F_i by $F_i[p_e]$. Note that in each F_i the edges are sampled independently, however, there are dependencies between different F_i subgraphs. We will now fix one particular F_i subgraphs and show that w.h.p. all the cuts in $F_i[p_e]$ are concentrated within $(1 \pm \epsilon)$ factor of their expectation. Then, by employing a union bound over all F_i subgrpahs, we will get that w.h.p. all cuts in all the $F_i[p_e]$ subgraphs are concentrated. Finally, since $H = G_w[p_e]$ is a linear combination of these subgraphs, all cuts in H are concentrated around their expected value as well.

To show that all cuts in $F_i[p_e]$ are concentrated, by Lemma 8.5 it is sufficient to show that for every cut C in F_i, its expected number of edges in $F_i[p_e]$ is large enough, i.e., $\sum_{e \in C} p_e = \Omega(\log n/\epsilon^2)$. Consider a cut C in some connected component of F_i (F_i is not necessarily connected). To bound the expected number of sampled edges, $\sum_{e \in C} p_e$, we use the following inequalities. First observe that by Lemma 8.9, $k_e(F_i) = k_e(G)$ (recalling that removing k_i weak edges, does not change the connectivity of the k_i strong edges). In addition, by definition, for each $e \in C$, $\lambda_e(F_i) \leq |C|$. Finally, by Obs. 8.7, we get that $k_e(F_i) \leq \lambda_e(F_i)$. Combining all these inequalities, we get that $k_e(G) = k_e(F_i) \leq \lambda_e(F_i) \leq |C|$. Thus,

$$\sum_{e \in C} p_e = \sum_{e \in C} q/k_e \geq q = \Omega(\log n/\epsilon^2) \ .$$

We get by Lemma 8.5 that all cuts in F_i are within $(1 + \epsilon)$ of their expectation w.h.p.. Applying the union bound over all F_i provides the correctness for H.

To digest this analysis, lets consider for simplicity that we have a graph G with only two values of strong edge connectivities, $k_1 < k_2$. Then, F_1 consists of all G edges (with strong connectivity at least k_1), and F_2 consists of all the edges e with $k_e = k_2$. Consider a cut $C = E(S, V \setminus S)$ in G. Let $C_1 = C \setminus F_2$ be all the edges in C with strong connectivity k_1 and let $C_2 = C \cap F_2$ by the cut edges with strong connectivity k_2. Hence, $C = C_1 \cup C_2$. We would like to show that w.h.p. $\delta_H(S) \in (1 \pm \epsilon)\delta_G(S)$. By applying Lemma 8.5 on the subgraph $F_1[p_e]$, we have that:

$$\delta_S(F_1[p_e]) = |C \cap F_1[p_e]| \in (1 \pm \epsilon) \sum_{e \in C} p_e.$$

By applying Lemma 8.5 on the subgraph $F_2[p_e]$, we have that:

$$\delta_S(F_2[p_e]) = |C_2 \cap F_2[p_e]| \in (1 \pm \epsilon) \sum_{e \in C_2} p_e.$$

Hence,

$$\begin{aligned} \delta_H(S) &\in (1 \pm \epsilon)\left(k_1/q \cdot \sum_{e \in C} p_e + (k_2 - k_1)/q \cdot \sum_{e \in C_2} p_e\right) \\ &\in (1 \pm \epsilon)\left(k_1/q \cdot \sum_{e \in C_1} p_e + k_2/q \sum_{e \in C_2} p_e\right) \in (1 \pm \epsilon)\delta_G(S) \ . \end{aligned}$$

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. 1993.

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