Advanced Algorithms	09/19, 2017
Lecture 1	
Lecturer: Mohsen Ghaffari	*Scribe: Jakob von Kalle*

1 Graph Sparsification — Preserving Cuts

Given an undirected unweighted graph with n-nodes $G = (V, E)$, we would like to find a much sparser weighted subgraph G' which has roughly the same cut sizes, that is,

$$\forall S \subset V, S \neq \emptyset : E_G(S, V/S) \approx E_{G'}(S, V/S)$$

Here, $E_{G'}(S, V/S)$ denotes the summation of the weights of the edges in G' connecting S to $V \setminus S$, and similarly, $E_G(S, V/S)$ denotes the number of the edges in G connecting S to $V \setminus S$. In particular, during this lecture, we will see a method that sparsifies the graph into merely $O(n \log n/\epsilon^2)$ edges, while preserving each cut size up to a $(1 \pm \epsilon)$ factor. That is, amazingly, we can reduce the number of edges of any graph down to almost linear, while keeping the cut sizes essentially the same.

Warm up: If G was a complete graph K_n, how would we go about obtaining a sparser graph G' satisfying the above properties?

One simple way would be to use the standard Erdős-Rényi random graphs. In particular, define a sparsified graph $G' = G_{n,p}$ in which we keep each edge of $G = K_n$ with probability $p = \Omega\left(\frac{\log n}{n}\right)$, discard the rest, and weight each kept edge with $\frac{1}{p}$. One can see that this fulfills the above criteria with high probability —i.e., probability at least $1 - 1/n^c$ for a desirably-large fixed constant $c \geq 2$. We will see a stronger version of this in exercise 3 of this week's problem set.

Plan: Next, we describe a method for general graphs. This method is based a similar uniform random sampling and allows us to get a somewhat sparser graph, if the original graph has a large edge-connectivity. Later, we see how to extend this to a non-uniform sampling scheme which provides us with the sparsifer with near-linear number of edges, as claimed above.

1.1 Uniform Sampling

Theorem 1 (Karger [Kar93]). *: Given a graph G with min-cut size k, suppose we sample each edge with probability $p = \Omega(\frac{\log n}{k})$. Then, in the resulting graph, all cuts are within a $1\pm\epsilon$ factor of their expectation, with probability at least $1 - \frac{1}{n^5}$.*

Proof. As a warm up, let us start with one cut of size k. The probability that none of the edges remain is $(1-p)^k \leq e^{-kp} = e^{-\Omega(\log n)}$, which is pretty small. What is the the probability that the number of sampled edges is not in $(1\pm\epsilon)\,kp$? This probability can be upperbounded using the Chernoff Bound. Let X_i be the indicator random variable of whether the i^{th} edge in the cut is sampled or not. Then, the number of sampled edges is $X = \sum_i X_i$ and we can write

$$Pr\left(X = \sum_i x_i \notin (1\pm\epsilon)\,E[X]\right) \leq 2\exp\left(\frac{-\epsilon^2 E[X]}{3}\right) = 2\exp\left(-\epsilon^2 kp/3\right) \leq \exp\left(-\frac{c}{4}\log n\right),$$

where in the last inequality, we have assumed $p \geq \frac{c \log n}{k}$. This means that one min-cut is not likely to deviate from its expectation. But we want to argue something much stronger; we want to show that, with high probability, no cut will deviate from its expectation. For that, we need to somehow union bound over all cuts. A graph has potentially up to $2^n - 2$ non-trivial cuts. Thus, we cannot directly union bound over all cuts (why?).

One observation is that cuts that are larger have a much smaller probability of deviation, as then the expected sampled size is larger and Chernoff gives us a much smaller probability. Thus, we should try to handle cuts of different sizes differently. Here's where the following key fact comes in:

Claim 2. $\forall \alpha \geq 1$, *the number of cuts of size at most αk is at most $n^{O(\alpha)}$.*

This claim—the proof of which is based on Karger's random contraction algorithm and is discussed in exercise 1 of this week's problem set—allows us to union bound over all cuts of size αk. In particular, using a calculation as above, we see that for each cut of size αk, the probability that in the sampled graph the cut is not within a $1 \pm \epsilon$ of its expectation is at most

$$\exp\left(-\alpha \epsilon^2 kp\right) \leq \exp\left(-\alpha \frac{c}{4} \log n\right)$$

Now we can use a union bound over all cuts, while bundling cuts of the same size together. We get the following upper bound on the probability of deviation of any cut:

$$\int_1^\infty n^{O(\alpha)} \cdot \exp\left(-\alpha \frac{c}{4} \log n\right) \cdot d\alpha \leq \int_1^\infty \frac{1}{n^{10\alpha}} \cdot d\alpha \leq 1/n^5$$

That is, with probability at least $1 - 1/n^5$, all cuts are within $1 \pm \epsilon$ of their expectation. $\square$

One can prove a slighly stronger version of the above lemma where different edges are sampled with different probabilities using essentially the same proof. We will make use of this version in the next subsection. We next state this stronger version but leave its proof as an exercise.

Theorem 3. *[Karger]: Given a graph G and suppose that each edge e is sampled independently with some probability p_e. Suppose that for each cut, the expected number of sampled edges in this cut is $\Omega(\log n/\epsilon^2)$. Then, in the resulting graph, all cuts are within a $1 \pm \epsilon$ factor of their expectation, with probability at least $1 - \frac{1}{n^5}$.*

1.2 Non-Uniform Sampling

The scheme that we saw above allows us to sparsify a large roughly by a factor of $k/\log n$, where k denotes the min-cut size of the graph. But this is not always good enough. A graph might be very dense, while having a very small min-cut size. Think of the Dumbbell graph, which is made of two $n/2$-node cliques connected via a single edge. This graph has $\Omega(n^2)$ edges and min-cut size 1, which means that uniform sampling discussed above would not suffice for sparsifying it. On the other hand, this example suggests that we should treat different edges differently. In particular, edges in "well-connected parts" of the graph can be sampled with lower probabilities.

To formalize this intuition, we first need a definition:

Definition 4 (Strong Connectivity)**.** *A k-strong component is a maximal induced subgraph that is k-edge-connected. For each edge e, the strength k_e of e is the maximum value k' such that there exists a k'-strong component that contains e.*

Lemma 5. *The above definition satisfies the following three properties*

(1) for each edge e, it's strength k_e is uniquely-defined

(2) For any two values k_1, k_2 such that $k_2 \geq k_1$, we have that k_2-strong components are a refinement of k_1-strong components. That is, each k_2-strong component is completely inside one of the k_1 strong components.

(3) $\sum_e \frac{1}{k_e} \leq n - 1$

Proof. The proofs of the first two properties are left as exercises, with the hint that given two intersecting k'-edge connected induced subgraphs, their union is also k'-edge connected.

We now argue about the third property. Consider a minimum cut of graph G. It has k edges and each edge e in this cut has strength $k_e \geq k$. Hence, the contribution of the edges on this cut to the summation $\sum_e \frac{1}{k_e}$ is at most $k \cdot 1/k = 1$. Let us remove all of these edges, now we remain with a graph with (at least) two components. We repeat a similar process. Each time, we pick one of the connected components of the remaining graph, and a min-cut of it, say with k' edges. Then, each of the edges in this cut has strength at least k' in the base graph, because it is in a k'-connected induced subgraph. Hence, the summation contributed by the edges of this cut is at most $k' \cdot 1/k' = 1$; we then remove these edges. Each time we remove a cut, we remove a value of at most 1 from the summation and we increase the number of connected components by at least 1. The process stops once we reach n components, as then we have an empty graph, and we started with 1 component. Hence, $\sum_e \frac{1}{k_e} \leq n - 1$. $\square$

The Sparsification Algorithm of Benczur and Karger [BK96]: Sample each edge e with $p_e = \frac{q}{k_e}$ where $q = \frac{C \log n}{\epsilon^2}$. If sampled, keep the edge with weight $\frac{1}{p_e}$.

Lemma 6. *The sampled graph satisfies the following two properties with high probability: (1) it has $O(n \log n/\epsilon^2)$ edges, (2) the size of each cut in the sampled graph is within a $(1 \pm \epsilon)$ factor of its size in the base graph G.*

Proof. We start with proving property (1). In the sampled graph, the expected number of edges is $O(nq) = O\left(\frac{n \log n}{\epsilon^2}\right)$. Therefore, by the Chernoff bound, the number of sampled edges is in $O(\frac{n \log n}{\epsilon^2})$.

Next, we argue about property (2). Let $G_w \rightarrow$ be the same as G, except that each edge e has weight $\frac{1}{P_e} = \frac{K_e}{q}$. We want to analyze the graph G_w under the random sampling process. However, analyzing the sampling process on G_w directly is difficult, because different edges in G_w have different weights and thus we cannot use standard concentration bounds to analyze the cut size of the resulting graph. Instead, we rewrite G_w as the summation of a number of uniformly-weighted graphs, as follows. Let $k_1 < k_2 < \ldots < k_{m'}$ be all the edge strengths values in graph G. Notice that $m' \leq m$ as each edge has one value. Define subgraph F_i as the spanning subgraph with all the edges of G that have strength at least k_i. Then, we can rewrite graph G_w as the following sum of graphs

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

To see why, just note that an edge e with strength $k_e = k_i$ has weight k_e/q on the left hand side, and weight $(k_i - k_{i-1}) + (k_{i-1} - k_{i-2}) + \ldots + (k_2 - k_1) + k_1 = k_i$ also on the right hand side.

Next, we analyze the result of the sampling process on each of graphs F_i separately. Notice that the sampling process on G_w directly translates to a sampling on graphs F_i. In particular, for each edge $e \in G_w$, we toss one coin (with the appropriate probability) and if it comes out head, then we keep the edge in G_w as well as in all F_i-s that contain e. Otherwise, none of them keeps e. Of course now the sampling processes in different graphs F_i are not independent of each other. But, in each F_i, different edges are sampled independently.

Consider one fixed F_i. We next argue that, during the sampled process, each cut in F_i is concentrated around its expectation. Consider one component C of the graph F_i. We argue that each cut in this component has expected size at least q, and therefore, we can apply Theorem 3 to analyze it and infer that each cut is concentrated around its expectation. Consider an arbitrary cut of component C, and suppose that it has k' edges. Then, the strength of each of the edges in this cut with respect to graph C is at most k'. We can then use Lemma 5(2) to infer that the strength of each of these edges in the original graph G is also at most k'. Hence, each of these edges is sampled with probability at least q/k', which implies that the expected number of sampled edges across this cut is at least q. Now, since each cut in C has expected sampled size at least q, we can apply lemma Theorem 3 to infer that after the sampling, each cut in C, and thus also similarly each cut in F_i, is within a $1 \pm \epsilon$ of its expectation.

By a union bound over all graphs F_i over different values of i, we can conclude that each of these graphs remains within a $1 \pm \epsilon$ factor of its expectation. Therefore, G_w is also within a $1 \pm \epsilon$ factor of its expectation. ☐

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.