1 Overview

In this lecture we will study further applications of Karger’s algorithm for the minimum cut problem as well as cut sparsification. Afterwards, we will introduce symmetry breaking in the distributed setting and look at Luby’s algorithm for Maximal Independent Set.

2 Cut-Counting for $\alpha$-min-cuts

Let $G = (V, E)$ be an undirected, unweighted graph with $|V| = n$. Let $k$ be the size of a min-cut in $G$. Recall from last time that Karger’s min-cut algorithm (also known as the “contraction algorithm”) recovers a fixed min-cut with probability at least $\frac{1}{\binom{n}{2}}$. As a result, we observe that the number of min cuts in $G$ must be at most $\binom{n}{2}$.

**Lemma 1** The number of minimum cuts in $G$ is at most $\binom{n}{2}$.

**Proof** Let $C = \{C \mid C$ is a min-cut of $G\}$ be the set of minimum cuts. Let $E_C$ be the event that Karger’s algorithm outputs $C$. We have that $\Pr[E_C] \geq \frac{1}{\binom{n}{2}}$, thus
\[
1 \geq \Pr[\bigcup_{C \in C} E_C] = \sum_{C \in C} \Pr[E_C] \geq |C| \left(\frac{n}{2}\right)^{-1}
\]
implies the lemma, since the events $E_C$ are mutually exclusive. \[\blacksquare\]

In fact, $\binom{n}{2}$ is tight for the $n$-cycle graph, since every set of 2 edges induces a min cut (i.e. the number of min cuts in an $n$-cycle is at least $\binom{n}{2}$).

What can we say about approximately minimum cuts? We define an $\alpha$-min-cut as follows.

**Definition 2** Say that the minimum cut in $G$ has $k$ edges. An $\alpha$-min-cut of $G$ is a partition $(X, V \setminus X)$ such that the number of cut edges is at most $\alpha k$.

Now, consider an $\alpha$-min-cut $C$. We modify Karger’s min-cut algorithm to contract edges at random until there are $2\alpha$ vertices left. Then we choose a partition of the remaining $2\alpha$ vertices uniformly at random to obtain our cut $S$.

**Lemma 3** Given an $\alpha$-min-cut $C$, $\Pr[S = C] \geq \frac{1}{\binom{n}{2\alpha}2^{2\alpha}}$.

**Proof** Let $C$ be an $\alpha$-min-cut. So $|C| \leq \alpha k$. For $1 \leq i \leq n - 2\alpha$, let $B_i$ be the event that the edge contracted in round $i$ is in $\delta(C)$ (the set of edges in min-cut $C$), given that no edge in $\delta(C)$ has been contracted in a previous round. Then $\Pr[B_i] \leq \frac{\alpha k}{\binom{\alpha k + n - i}{2\alpha}} = \frac{2\alpha}{n-i+1}$. So the probability that our cut $S$
is the \( \alpha \)-min-cut \( C \) is
\[
Pr[S = C] \geq \frac{1}{2^{2\alpha}} \prod_{i=1}^{n-2\alpha} (1 - Pr[B_i]) \\
= \frac{1}{2^{2\alpha}} \prod_{i=1}^{n-2\alpha} \frac{n - i + 1 - 2\alpha}{n - i + 1} \\
= \frac{1}{2^{2\alpha}} \frac{(n - 2\alpha)!}{(2\alpha)!} \\
= \frac{1}{2^{2\alpha}} \binom{n}{2\alpha}
\]

As a corollary we get that the number of \( \alpha \)-min-cuts is bounded as follows.

**Corollary 4** The number of \( \alpha \)-min-cuts is at most \( n^{2\alpha} \)

### 3 Network Survivability

Let \( G = (V, E) \) be a connected graph and suppose we want to find a sparse subgraph that is still connected. A simple procedure for sparsifying the graph is to subsample the edges with some probability \( p_e \). What should \( p_e \) be in order to preserve connectedness?

**Theorem 5** Subsampling edges of \( G \) with probability \( p = O(\log n/k) \), where \( k \) is the size of the minimum cut leaves the graph connected with high probability.

**Proof** The trick will be to show that every cut has some number of edges crossing it in the subsampled graph. Let \( G' \) be the subsampled graph. Let \( S \) be an \( \alpha \)-min-cut and \( \delta_{G'}(S) \) be the edges crossing this cut. In expectation we have
\[
E[|\delta_{G'}(S)|] = \sum_{e \in \delta_{G}(S)} p_e = |\delta_{G}(S)| c \log(n)/k \leq c\alpha \log(n).
\]

Now applying Chernoff bounds we get that
\[
Pr[\delta_{G'}(S) \text{ disconnected}] \leq Pr[|\delta_{G'}(S)|] \leq \frac{1}{30} c\alpha \log(n) \leq \exp(-100c\alpha \log(n)) = n^{-d\alpha}
\]
for some constant \( d \). Now using the result of Corollary 4 and applying a union bound, we have:
\[
Pr[\exists \text{ disconnected } \alpha\text{-min-cut}] \leq n^{2\alpha} \cdot n^{-d\alpha} = n^{-(d-2)\alpha}
\]
Finally we take a union bound over different \( \alpha \)'s to show that \( G' \) is connected with high probability.
\[
Pr[G' \text{ disconnected}] \leq \sum_{\alpha \geq 1} n^{-(d-2)\alpha} = O(n^{-(d-2)}).
\]
So taking \( d \) large enough yields the result. \( \blacksquare \)
4 Cut Sparsifiers

Suppose that in addition to preserving connectedness, we also preserve the value of all cuts up to some constant factor. Now how should we set the edge sampling probability $p_e$?

**Definition 6** $G'$ is a cut sparsifier of $G$ if they are both defined on the same vertex set $V$ and for all $X \subseteq V$,

$$\frac{w(\delta_{G'}(X))}{|\delta_G(X)|} \in [1-\epsilon, 1+\epsilon].$$

Where $w(C)$ is the total weight of the edges in set $C \subseteq E(G')$.

**Theorem 7** Subsampling edges of $G$ with probability $p_e = O((\log(n)/\epsilon^2 k)$ and including it in $G'$ with weight $w_e = 1/p_e$ yields a cut sparsifier with high probability. $k$ is again the size of a minimum cut in $G$.

**Proof** The proof is essentially the same as the connectedness proof, but we use both sides of the Chernoff bound to show that the number of sampled edges in every cut is close to its expectation with high probability. Let $p_e = 3c \log(n) \epsilon^2$. Fix an $\alpha$-min-cut $S$ of $G$, the expected number of sampled edges of this cut is $\alpha^2 k p_e$. Thus by Chernoff bound we have,

$$\Pr[|\delta_{G'}(S)| \notin (1 \pm \epsilon) \alpha k p] \leq 2 \exp(-\epsilon^2 \alpha^2 k p/3) = 2 \exp(-c \alpha \log(n)) = n^{-\alpha c}$$

Now we apply the same two steps of union bounds as before to get that the number of sampled edges is preserved with high probability. Using this we can show that the weight of the cut in $G'$ is approximately the same as the size of the cut in $G$. With high probability, we have that $|\delta_{G'}(S)| \in (1 \pm \epsilon) p_e |\delta_G(S)|$ with high probability. Now we have that

$$w(\delta_{G'}(S)) = \sum_{e \in \delta_{G'}(S)} w_e = \frac{1}{p_e} |\delta_{G'}(S)| \frac{1}{p_e} (1 \pm \epsilon) p_e |\delta_G(S)| = (1 \pm \epsilon)|\delta_G(S)|.$$  

Thus $G'$ is a cut sparsifier with high probability.

5 Cut Sparsification with Strength

The above results implicitly assume that the minimum cut has size $\omega(\log n)$. In many cases, the minimum cut can be very small. For example consider two copies of the complete graph $K_n$ connected by a single edge. In this case the above sampling probability will not work since we need to ensure that we keep the single edge connecting the two complete graphs. Intuitively we need to take into account how important each edge is. The notion of strength will handle this problem.

**Definition 8** Let $e$ be an edge of graph $G$. The strength of $e$, $\text{str}(e)$, is the maximum connectivity of a subgraph of $G$ that $e$ belongs to.

The following theorem is shown by Benczur and Karger [1].

**Theorem 9** Subsampling each edge $e$ of $G$ with probability $p_e = O(\frac{\log(n)}{\epsilon^2 \text{str}(e)})$ and including it in $G'$ with weight $w_e = 1/p_e$ yields a cut sparsifier with $O(n \log n/\epsilon^2)$ edges with high probability.

We refer the reader to Benczur and Karger’s paper for the full proof. Some of the techniques are the same as the previous results we have shown, but there is more work to be done. In these notes we will show the bound on the expected number of edges. The expected number of edges in $G'$ is as follows.
\[ E[|E'(G)|] = \sum_{e \in E(G)} p_e = \sum_{e \in E(G)} \frac{c \log(n)}{\epsilon^2 \text{str}(e)} = \frac{c \log(n)}{\epsilon^2} \sum_{e \in E(G)} \frac{1}{\text{str}(e)}. \]

The following lemma implies the bound.

**Lemma 10** For any graph \(G\), \(\sum_{e \in E(G)} \frac{1}{\text{str}(e)} \leq n - 1\)

**Proof** Fix a cut \(C\) with \(k\) edges. For each edge \(e \in C\), we have \(\text{str}(e) \geq k\), thus we have \(\sum_{e \in C} \frac{1}{\text{str}(e)} \leq k \frac{1}{k} = 1\). Removing these edges increases the number of connected components in \(G\) by 1. Repeating this \(n - 1\) times yields a graph with no edges, thus accounting for all edges of the graph and giving the bound of \(n - 1\).

Thus the expected number of sampled edges is \(O(n \log n)\). This concludes our discussion of Cut Sparsification. We note that in the last decade, new methods of sparsifying graphs have been found. They use spectral techniques and are much more time efficient [2].

### 6 Symmetry Breaking and Luby’s Algorithm

We now introduce the idea of symmetry breaking with the unique ID’s problem. Say that we are given \(n\) objects and we want to give a simple way to assign unique ID’s to the objects without much coordination between the objects. The following simple random procedure works with high probability, just have each object chooses a value uniformly at random from the set \(\{1, 2, \ldots, n^c\}\) as its ID.

**Theorem 11** For each object \(a \in [n]\) choose \(ID(a)\) uniformly at random from \(\{1, 2, \ldots, n^c\}\). Then the ID’s are unique with high probability.

**Proof** Fix a pair \(a, b \in [n]\). Then \(\Pr[ID(a) = ID(b)] = \frac{1}{n^c}\). Applying a union bound over all pairs, we get

\[ \Pr[\text{ID’s not unique}] \leq \binom{n}{2} \frac{1}{n^c} = \frac{1}{n^{c-2}}. \]

We note that the probability \(\frac{1}{n^{c-2}}\) is roughly necessary due to the birthday problem. The above procedure is useful in distributed setting since no coordination between machines is needed to determine the unique ID’s. We now introduce a distributed model.

#### 6.1 Distributed Model

Let \(G = (V, E)\) be graph. The nodes of the graph are machines that can communicate with their neighbors. A computation in this setting generally operates in synchronous rounds and follows a certain pattern:

- For \(T\) rounds:
  1. Nodes send messages to their neighbors
  2. Nodes receive messages from their neighbors
  3. Nodes to local computations with the information communicated so far

The general goal is to compute something about the underlying graph \(G\) in as few rounds as possible and communicating as little information as possible. We will now study Luby’s algorithm for computing Maximal Independent Sets in this model.
6.2 Luby’s Algorithm

Let $G = (V, E)$ be a graph on $n$ vertices and $m$ edges. A set $I \subseteq V$ is independent if no two vertices of $I$ are connected by an edge of $G$. A set $I$ is a maximal independent set (MIS) if $I$ is independent and no superset of $I$ is independent. The following result was first given by Luby [3].

**Theorem 12** There is a randomized $O(\log(n))$-round distributed algorithm that outputs an MIS that succeeds with high probability.

A surprising fact about this result is that an MIS can be computed without having all nodes communicate with each other, only very local information is necessary to succeed. The algorithm is very simple and is as follows. For notation let $\Gamma(X)$ be the set of neighbors of vertices in $X$ for $X \subseteq V$.

```
procedure MIS(G)
    X ← ∅
    while G is not empty do
        Each node chooses a uniform random priority in [n^3]
        If node v is a local maximum, then X ← X ∪ {v}
        Delete all nodes in X ∪ Γ(X)
    Output X
```

Since we delete all nodes in $X \cup \Gamma(X)$ at each step, it follows that any $v$ we choose in step 2 of the loop is independent of the previous chosen nodes in $X$. Thus the main difficulty is showing that only $O(\log(n))$ rounds are needed to finish running the algorithm.

There are many things we might try to bound in order to show that $O(\log n)$ rounds suffice to compute an MIS. We might try to show that nodes or edges get removed with constant probability in each round, or we might try to show that a constant fraction of nodes get removed in each round. It turns out that all of these approaches will fail. The approach that succeeds is to show that a constant fraction of edges are deleted in a round.

**Lemma 13** Let $R$ be the set of edges that remain in $G$ after one round of the algorithm. Then $E|R| \leq m/2$

Once we have this lemma it follows that the expected number of rounds is at most $O(\log(m)) = O(\log(n))$. We can get high probability rounds by applying concentration inequalities.

To prove this lemma we need to carefully define the right random variables.

**Definition 14** (Strong Deletion) We say that $u$ strongly deletes $v$ if (1) $u$ is in the MIS $X$ and (2) $u$ has the largest priority in $\Gamma(v) \cap X$. We denote this event by $X \rightarrow v$

**Claim 15** $Pr[X \rightarrow v] \geq \frac{1}{d(u) + d(v)}$, where $d(v)$ is the degree of $v$.

**Proof** $u$ needs to be the local maximum in both $\Gamma(v)$ and in $\Gamma(u)$, this happens with probability at least $1/(d(u) + d(v))$. ■

**Definition 16** (Strong Edge Deletion) We say that edge $e$ is strongly deleted by $u, v$ if $X \rightarrow v$ and $e$ has $v$ as an endpoint. We denote this event by $X \rightarrow v$.

**Claim 17** Every deleted edge is strongly deleted over all $u, v$ at least once and at most twice.

**Lemma 18** The expected number of strong edge deletions is at least $m$ and the expected number of edge deletions is at least $m/2$. 2-5
**Proof** Let $S$ be the number of strong deletions and $D$ the number of deletions in a single round. Then we have

$$E[S] = \sum_{uv \in E} (\Pr[X_u \rightarrow v]d(v) + \Pr[X_v \rightarrow u]d(u)) \geq \sum_{e \in E} \frac{d(u) + d(v)}{d(u) + d(v)} = m.$$ 

By Claim 17 we have that $D \geq S/2$, thus $E[D] \geq E[S]/2 \geq m/2$.

The proof of Lemma 13 now follows since $E[R] \leq m - E[D] \leq m/2$

**References**

