Lecture 4

Definition 1. *Given $G = (V, E)$, the* Uniform Sparsest Cut Problem *is the problem of computing*

$$\beta(G) = \min_{\emptyset \neq S \subset V} \frac{|\delta(S, V \setminus S)|}{|S|\,|V \setminus S|}$$

where $\delta(A, B) = \{(u, v) \in E | u \in A, v \in B\}$.

Remark 1. *We are often interested not only in the value β, but also in the argument, S of the equation.*

Finding the Sparsest Cut is an NP-hard problem, but is also related to expansion. Recall:

$$\phi(G) = \min_{S \subset V, |S| \leq |V|/2} \frac{|\delta(S, V \setminus S)|}{|S|}.$$

Thus, $\frac{\phi(G)}{n} \leq \beta(G) \leq 2\frac{\phi(G)}{n}$.

So, if we can find an approximate solution for the sparsest cut problem, we can also find an approximation for expansion. For example, if we have a family of r-regular expanders, then we know that $\beta(G) = \Theta(1)$.

Before we introduce the non-uniform version of the sparsest cut problem, we define the concurrent multicommodity flow problem.

Definition 2. The Concurrent Multi-Commodity Flow Problem*: We are given $G = (V, E, c)$ where $c : E \to \mathbb{R}^+$ and k "commodities" such that $\forall i \in [k]$, we have source/destination pairs $(s_i, t_i) \in V \times V$ and demand D_i. The goal is to maximize α such that $\forall i$ we can simultaneously send αD_i units between s_i and t_i with respect to the capacities c. Let $\alpha(G)$ denote the maximum such value.*

Example: Using the complete bipartite graph $K_{2,3}$, we set $k = 4$ and set $\forall e : c(e) = 1$ and $\forall i \in [4] : D_i = 1$. We set our sources and sinks as shown in Figure 1. For this example, we see a valid concurrent flow in Figure 2 with $\alpha = 3/4$.

In our example, each path between source and destination for any commodity has length 2, so αD_i uses 2α units of capacity. As we have 4 units of demand and a total capacity of 6 in the graph, we see that $8\alpha \leq 6 \Rightarrow \alpha \leq \frac{3}{4}$. Thus, for our instance, $\alpha(G) = \frac{3}{4}$.

To get an upper bound on $\alpha(G)$, we can look at every cut and the requirement through this cut. This leads to the following (non-uniform) Sparsest Cut problem:

Definition 3. *Let*

$$\beta(G) = \min_S \frac{\sum_{e \in \delta(S, V \setminus S)} c(e)}{\sum_{s_i, t_i : s_i \in S \text{ XOR } t_i \in S} D_i}.$$

Observation 1. *It follows that $\alpha(G) \leq \beta(G)$ since in the cut $\delta(S, V \setminus S)$ we would need $\sum_{s_i, t_i : s_i \in S \text{ XOR } t_i \in S} D_i$ if we were to fully satisfy the demands. In our example, $\alpha(G) = \frac{3}{4} < \beta(G) = 1$.*

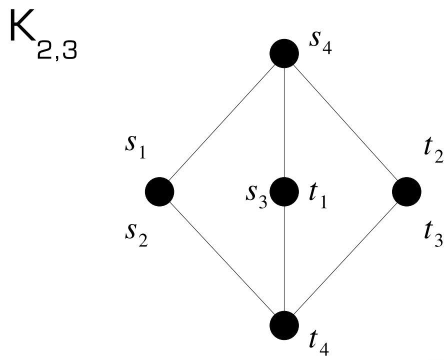

Figure 1: The complete bipartite graph with sizes 2 and 3.

Observation 2. *For another example, we look again at constant-degree expanders. From before we saw that the expansion $\phi(G)$ is constant, and thus $\beta(G) \geq \phi(G)/n = \Omega(1/n)$.*

In order to calculate α, we note that in an expander graph, $\sum_{e \in E} c(e) = \frac{rn}{2} = \Theta(n)$ and for most s_i, t_i pairs, the path length between them will be $\Omega(\log n)$. So, $\alpha(G) = \frac{\Theta(n)}{\Omega(n^2)\Omega(\log n)} = O(\frac{1}{n \log n})$. Therefore, $\frac{\beta(G)}{\alpha(G)} = \Theta(\log n)$.

Remark 2. *In some cases, $\alpha(G) = \beta(G)$. The first is simple: $k = 1$. Equality follows directly from the equivalence of Max-Flow and Min-Cut. T.C. Hu proved that the two are also equal when $k = 2$ in 1963.*

Now, we relate these values to embeddings and their distortions. We will first derive two very similar looking expressions for $\beta(G)$ and $\alpha(G)$.

Lemma 1. $\beta(G) = \min_{(V,d)\ l_1-\text{embeddable}} \frac{\sum_{e \in E} c(e) \cdot d(e)}{\sum_{i \in [k]} D_i d(s_i, t_i)}$.

Proof. Use the cut metric induced by S: $d(u, v) = 1 \iff u \in S\ XOR\ v \in S$. Then

$$\beta(G) = \min_{\text{cut metrics } d} \frac{\sum_{e \in E} c(e) \cdot d(e)}{\sum_i D_i d(s_i, t_i)} = \min_{d \in \mathrm{CUT}_n} \frac{\sum_{e \in E} c(e) \cdot d(e)}{\sum_i D_i d(s_i, t_i)}.$$

The last equality follows from the fact that CUT_n is precisely the cone of all the cut metrics and that $\beta(G)$ is the ratio of two linear functions. More precisely, if we let a_i and b_i be the numerator and denominator for cut i, we obtain $\min_i \left\{ \frac{a_i}{b_i} \right\}$ when we optimize over the cuts while we get

$$\min_{\lambda_i \geq 0\ \forall i} \frac{\sum_i \lambda_i a_i}{\sum_i \lambda_i b_i}$$

when we optimize over CUT_n. The two quantities are equal for all choices of a_i, b_i.

Recall from Lecture 1 (see Lemma 4 there) that CUT_n is exactly the set of all l_1-embeddable (semi)-metrics, and this completes the proof of the lemma. $\square$

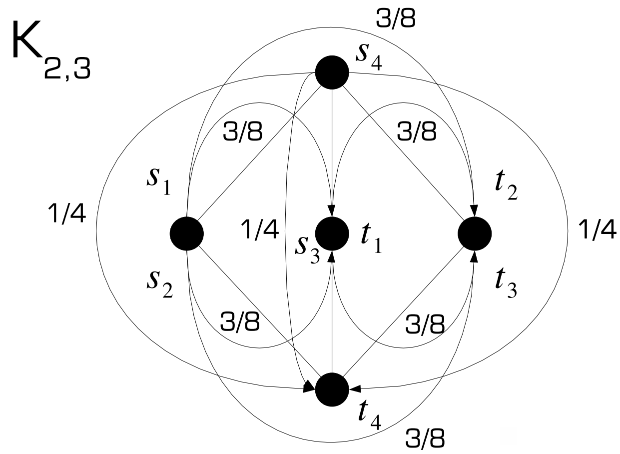

Figure 2: A solution providing as much concurrent flow as possible.

Lemma 2. $\alpha(G) = \min_{(V,d)\ l_\infty\text{-}embeddable} \frac{\sum_{e \in E} c(e) \cdot d(e)}{\sum_i D_i d(s_i, t_i)}$.

Proof. Formulate $\alpha(G)$ as a linear program. We will do this in a seemingly straightforward and stupid way. For all i, we enumerate all paths between s_i and t_i. So, P_{ij} is the j^{th} such path between s_i and t_i and x_{ij} is the amount of flow on P_{ij}. Then,

$$\begin{aligned}
\alpha(G) \quad = \quad & \max \alpha \\
s.t. \quad & \\
& \alpha D_i - \sum_j x_{ij} \leq 0 & i = 1, \cdots, k \qquad (1) \\
& \sum_{i=1}^{k} \sum_{j: e \in P_{ij}} x_{ij} \leq c(e) & e \in E \qquad (2) \\
& \alpha \geq 0 & \\
& x_{ij} \geq 0 & \forall i, j
\end{aligned}$$

By strong duality, we know that $\alpha(G)$ also equals to (using dual variables h_i for constraints (1) and l_e for constraints (2)):

$$\begin{aligned}
\alpha(G) &= \min \sum_{e \in E} c(e) \\
s.t. & \\
& \sum_i D_i h_i \geq 1 && && (3)\\
& \sum_{e \in P_{ij}} l_e - h_i \geq 0 && \forall i, j && (4)\\
& h_i \geq 0 && \forall i \\
& l_e \geq 0 && \forall e
\end{aligned}$$

Observe that we can assume that (3) is an equality (by decreasing some of the h_i's without violating any other constraint), and similarly that for every i, h_i equals the shortest path length with respect to l_e between s_i and t_i (i.e. for every i, there exists j with (4) an equality).

Thus, from the optimum solution l, h of the above dual program, we can consruct the shortest path metric d corresponding to the lengths l_e, and we have that

$$\alpha(G) = \frac{\sum_{e \in E} c(e) \cdot d(e)}{\sum_i D_i d(s_i, t_i)}.$$

Vice versa, if we have an l_∞=embeddable metric d which minimizes $\frac{\sum_{e \in E} c(e) \cdot d(e)}{\sum_i D_i d(s_i, t_i)}$, without loss of generality (without increasing the ratio), we can assume that this metric is the shortest path metric corresponding to d_e for $e \in E$ (i.e. the distance separating the endpoins of a non-edge is indeed given by the shortest path). Thus, it leads to a feasible solution of the above dual program and this proves equality and completes the proof. $\square$

Linial, London and Rabinovich, '95 and Aumann and Rabani '95 proposed to solve the linear program to get the value α and the l_∞-embeddable metric d that optimizes it. Now, embed d into the l_1=metric l with distortion γ (with γ chosen as small as possible); thus we have that $l(u, v) \leq d(u, v) \leq \gamma l(u, v)$ for all u, v. Then

$$\beta(G) \leq \frac{\sum c(e) l(e)}{\sum D_i l(s_i, t_i)} \leq \gamma \frac{\sum c(e) d(e)}{D_i d(s_i, t_i)} = \gamma \alpha(G),$$

showing that $\frac{\beta(G)}{\alpha(G)} \leq \gamma$. In addition, the l_1-embeddable metric l can be decomposed into at most $\binom{n}{2}$ cut metrics (see Lemma 4 from Lecture 1), and the best of these gives a cut S with

$$\frac{\sum_{e \in \delta(S, V \setminus S)} c(e)}{\sum_{s_i, t_i : s_i \in S \ XOR \ t_i \in S} D_i} \leq \gamma \alpha(G) \leq \gamma \beta(G),$$

and therefore approximates the sparsest cut within a ratio of γ.

What remains is to establish how small γ can be. In general, Bourgain proved that γ can be $O(\log n)$ for any metric. (We will see a proof of this in the next lecture.) This implies that for all G, k, c, D : $\frac{\beta(G)}{\alpha(G)} \leq O(\log n)$. This logarithmic gap between sparsest cut and concurrent flow was already known for the uniform case; this was established (without embeddings of metric spaces) by Leighton and Rao '88.

If the graph G is planar, Rao has shown that for any planar graph metric d, we have $c_1(d) \leq c_2(d) = O(\sqrt{\log n})$. Thus for every planar graph independently of the number of commodities), we have that $\frac{\beta(G)}{\alpha(G)} = O(\sqrt{\log n})$.

Last time we noted that a regular expander graph metric d satisfies $c_2(d) = \Omega(\log n)$. Today we saw that $\frac{\beta(G)}{\alpha(G)} = \Omega(\log n)$ for expander graphs, implying that $c_1(d) = \Theta(\log n)$. Thus, Bourgain's result cannot be improved for general graphs.

We have two closing remarks.

Remark 3. *The embedding of any metric into l_1 with distortion $O(\log n)$ can done efficiently. This implies that the Linial-London-Rabinovich result is algorithmic.*

Remark 4. *: Suppose we don't care about concurrent flow, and just finding an approximation to the sparsest cut. Instead of optimizing over l_∞ metrics (which was convenient wince we could use linear programming), we can try instead to optimize over a more restrictive class of metrics which nevertheless includes all cut metrics (so that the resulting bound is a lower bound on $\beta(G)$).*

One class for which this has been successful is the class of negative type metrics. A metric (X, d) is a it negative-type metric if $(X, \sqrt{d})$ is iso-metrically embeddable into l_2.

This only begs more questions. One question is: can we optimize in polynomial time over negative-type metrics? It turns out that we can—at least, approximately—using Semi-Definite Programming. The clear follow up to this is: what distortion do we need to embed negative type metrics into l_1? According to Arora, Lee and Naor (2005), that distortion is $O(\sqrt{\log n} \log \log n)$.