(c) Graph Sparsification

Undirected Graph G, say unweighted (again extends to weights). ↖ but requires a bit more work

Want a graph H, subgraph of G, and a scaling factor M, such that for every cut $S \subseteq V$,

$$M(1-\varepsilon) \leq \frac{|\partial_G(S)|}{|\partial_H(S)|} \leq M(1+\varepsilon) \qquad \forall S \subseteq V.$$

So cuts in H approximate cuts in G, upto a uniform scaling factor M and error $(1 \pm \varepsilon)$.

H is a ε-cut approximator for G. (if it is a cut apximator for some M).

- **Observe**: G is a cut approximator for G (duh).

  Want H to have few edges.

Thm 1: Suppose the min cut in G has λ edges.

then $\exists H$ with $O\left(\frac{m}{\lambda} \cdot \frac{\log n}{\varepsilon^2}\right)$ edges that ε-cut-approximates G.

Example: if $G = K_n$ then $m = \binom{n}{2}$. and $\lambda = (n-1)$.

then $\exists$ a ε-cut apximator for K_n with $O\left(\frac{n \log n}{\varepsilon^2}\right)$ edges.

Digression:

why do we care?

Sps want to find cuts in some large network (social network, e.g)

for data analysis

↳ small cuts, large cuts, low conductance cuts, etc.

working on H (sparser) $\tilde{O}(n)$ ~~time~~ size!

vs G (denser) $\tilde{O}(n^2)$ size maybe?

almost the same from the perspective of cuts!

May want other properties, of course:

So lots of interest in other kinds of sparsifiers

- distance sparsifiers — maintain all pairs distance (apx)
- congestion sparsifiers — maintain all flows.
- spectral sparsifiers — maintain algebraic quantities like eigenvalues.

this is just one example

— cut sparsifiers

OK: Back to Proof of Thm 1

Proof: define $p = \left(\frac{c \cdot \log n}{\lambda} \cdot \frac{1}{\varepsilon^2} \wedge 1\right)$ $\quad a \wedge b = \min(a,b)$

if $p=1$, nothing to prove.
so assume $p<1$.

Pick each edge w.p. p. independently into H.

$\Rightarrow$ fix any $S \subseteq V$. for each edge in $\partial_G S$, it belongs to $\partial_H S$ w.p. p.

$\Rightarrow$ if $X_e = 1$ if e picked (and so $E X_e = p$).

then $|\partial_H S| = \underbrace{\sum_{e \in \partial_G S} X_e}_{\text{indep } \{0,1\} \text{ r.v.s.}}$ $\quad \Rightarrow \quad E|\partial_H(S)| = p \cdot |\partial_G(S)|$.

$$\Rightarrow \Pr\Big[\underbrace{|\partial_H(S)| \notin (1 \pm \varepsilon) \cdot p \cdot |\partial_G(S)|}_{\sum X_e \notin (1\pm\varepsilon) \cdot E[\sum X_e].}\Big] \leq \exp\left(-\frac{\varepsilon^2}{3} \cdot p \cdot |\partial_G(S)|\right)$$

$$= \underbrace{\exp\left(-\frac{c}{3} \cdot \log n \frac{|\partial_G(S)|}{\lambda}\right)}_{(*)}$$

Great: but there are 2^{n-1} cuts to argue about ☹

How to take a union bound over them?

It's time to remember:

> Exercise from HW1:
>
> there are at most $n^{2\alpha}$ cuts of size $\leq \alpha \cdot \lambda$. α. times min cut. $\forall \alpha$.

$$\Pr[\exists \text{ a bad cut } S] \leq \sum_{\alpha \in \mathbb{Z}_{\geq 1}} \sum_{S : \partial_G(S)} \Pr\left(|\partial_H S| \notin (1\pm\varepsilon) \cdot p \cdot |\partial_G(S)|\right)$$

such that $\alpha\lambda \leq |\partial_G(S)| \leq (\alpha+1)\lambda$.

$$\leq \sum_{\alpha \in \mathbb{Z}_{\geq 1}} n^{2(\alpha+1)} \cdot \exp\left(-\frac{c}{3} \cdot \log n \cdot \alpha\right) \leq \sum_{\alpha \in \mathbb{Z}_{\geq 1}} n^{2(\alpha+1)} \cdot \frac{1}{n^{c\alpha/3}}$$

$= \frac{O(1)}{\text{poly}(n)}$ if $c \gg 3$.

⇒ with prob $(1 - 1/\text{poly}(n))$ we have that

$$\frac{|\partial_G(S)|}{|\partial_H(S)|} \in \frac{1}{p}\cdot(1\pm\varepsilon) \qquad \forall S\subseteq V.$$

So H is a ε-cut-approximator w.h.p.

Also: # Edges in H is pm in expectation

and also $(1\pm\varepsilon)pm$ with probability $\left(1-\exp\left(-\frac{\varepsilon^2}{3}\cdot pm\right)\right)$

$$= \left(1-\exp\left(-\frac{cm\log n}{3\lambda}\right)\right)$$

$$= 1 - 1/\text{poly}(n) \text{ for large enough } c>>3.$$

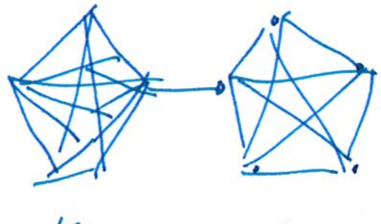
In summary: whp. H has $O\left(\frac{m\log n}{\varepsilon^2\cdot\lambda}\right)$ edges

and $|\partial_G(S)| \cong \frac{1}{p}\cdot|\partial_H(S)|$ ← up to $(1\pm\varepsilon)$ factors!

—— x ——

This is great .... ☺

... except it depends on λ. So if λ is small, say $\lambda=1$, then not useful ☹



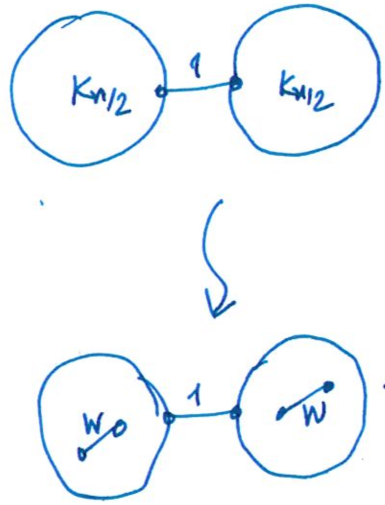
How to sparsify? ☹

$K_{n/2}$ $K_{n/2}$

$\lambda=1$.

Suppose we allow weights on the graph H edges.

then think of



sparse graph with $O\left(\frac{n \log n}{\varepsilon^2}\right)$ edges

but each of wt $\approx W = \frac{\varepsilon^2 n}{\log n}$

so that each cut is maintained up to $(1 \pm \varepsilon)$.

Replace unweighted graph G

by weighted graph H st $w(\partial_G S) \cong w(\partial_H S)$ $\forall S \subseteq V$

In fact previous construction can be thought of as setting weight $W = 1/p$ on each edge of H. (to scale it up "correctly").

- Idea of Benczúr–Karger

which started a large area of graph sparsification

(hypergraph)

(codes)

(CSP)

⋮

Graph Sparsification (Take 2)

exercise!

G is undirected (unweighted, but this can be made weighted).

want H weighted so that

$$w(\partial_H(S)) \in (1 \pm \varepsilon) \cdot w(\partial_G(S)) \qquad \forall S \subseteq V.$$

and $|E(H)|$ small.

Notation: H is an ε-cut-approximator of G.

Thm: $\forall G = (V,E)$, $\exists$ weighted graph $H = (V, E')$ and edge weights $w_e : e \in E'$

such that H is a (weighted) ε-cut approximator for G and.

$$|E(H)| \le O\left(\frac{n \log n}{\varepsilon^2}\right). \qquad \text{[Benczúr–Karger]}$$

$$\le O(n/\varepsilon^2) \qquad \text{[Batson–Spielman–Srivastava].}$$

First result uses randomization and simple proof

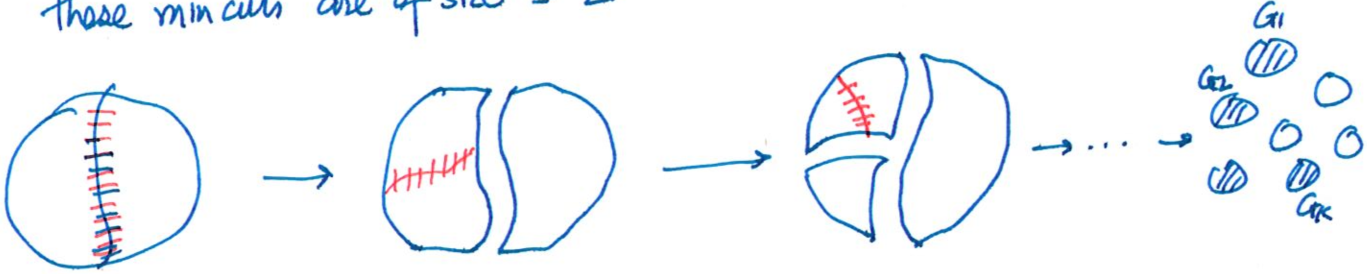
Second result deterministic and more involved.

Today: let's see a much simpler result that shows.

$$|E(H)| = O($$

Simple Idea: [Khanna-Putterman-Sudan]

Given graph, find ~~...~~ non-trivial min cuts repeatedly as long as those mincuts are of size $\leq Z$



(I) Each time # components increases, $\Rightarrow$ at most $(n-1)$ steps.

Each step delete $\leq Z$ edges $\Rightarrow$ delete $\leq nZ$ edges.

Now: on the final graphs G_i, which have only mincuts of size $\geq Z$,

use Karger sampling (as above) to get sparser graphs

(scale them by $\frac{1}{p} = \left(\frac{c \log n}{\varepsilon^2 Z}\right)^{-1}$ to get the graphs H_i)

$\boxed{\text{Know}: w(\partial_{H_i}(S)) \approx w(\partial_{G_i}(S)) = |\partial_{G_i}(S)|.}$ $\forall S$ whp.

Sparsifier = All red edges deleted in Step 1. $(\leq \boxed{nZ})$

+ all edges in H_i's $\left(\leq O(m \cdot p) = \boxed{O\left(\frac{m}{Z} \frac{\log n}{\varepsilon^2}\right)}\right)$

choose $Z = O\left(\sqrt{\frac{m}{n} \log n / \varepsilon^2}\right)$ to get

graph with $O\left(\sqrt{mn \log n / \varepsilon^2}\right)$ edges

and cuts maintained upto $(1+\varepsilon)$!

$\boxed{\text{call this } H}$

Not quite $m \to \frac{n \log n}{\varepsilon^2}$ but $m \to \sqrt{mn}$

if $m = \Theta(n^2)$ $\Rightarrow$ get $n^2 \to n^{1.5}$.

looks great! could recurse

$$\overset{G}{n\cdot(m/n)} \to \overset{H}{n\cdot(m/n)^{1/2}} \to \overset{H'}{n(m/n)^{1/4}} \to \overset{H''}{n(m/n)^{1/8}} \dots$$

for $O(\log\log(m/n))$ levels

losing $(1+\varepsilon)$ in each level to get

$O(n \log n/\varepsilon^2)$ edges with $(1+\varepsilon)^{\log\log n}$ loss.

Now choose $\varepsilon = \frac{\delta}{\log\log n}$ to get

$O(n \log n (\log\log n)^2/\delta^2)$ edges with $(1+\delta)$ loss.

Or do we?

Slight problem:– H is a weighted graph.

And we've done sparsification only for unweighted graphs.

Solution:

Right now: did G (m edges) $\longrightarrow$ A (small cuts) + B (rest)

A $\xrightarrow{\text{retain}}$ H $\xleftarrow{\text{sample + scale up edges}}$ B

H (weighted) $\tilde{O}(\sqrt{mn})$ edges

$\rightsquigarrow$???

Instead: $G \longrightarrow A + B$

A (small cuts) $\qquad$ B (rest / large mincuts)

$A \downarrow$ recurse $\downarrow$ $\hat{A}$ sparse

$B \downarrow$ ~~recurse~~ subsample ~~& scale up~~. $\downarrow$ recurse on sampled graph $\downarrow$ $\hat{B}$

$\hat{A}, \hat{B} \xrightarrow{\text{scale up}} \; \downarrow \; \hat{H}$ weighted + sparse $O\left(n \frac{\text{polylog } n}{\varepsilon^2}\right)$ edges.

See notes or paper of Khanna Putterman Sudan

More sophisticated ideas in

1. Benczúr - Karger
2. Spielman-Srivastava (spectral sparsifiers & Matrix Chernoff)
3. Batson-Spielman-Srivastava (deterministic optimal $O(m/\varepsilon^2)$ edges)

⋮