0.1 Sparsification matching Spielman bounds

In this subsection, we describe a sparsification algorithm that involves flipping a fair coin on each edge, to determine wehther we keep it in the sparsifier. We iterate this $\log \frac{m}{n}$ times.

This is known **TIMOTHY: is it?** . Our algorithm for degree-preserving sparsification on unit weight graphs will use the same technique.

0.1.1 Algorithm Description

1. Take the graph G, and find all edges with effective resistance less than $2(n-1)/m$ (twice the average effective resistance).
2. Toss a fair coin on those each of those edges, to determine whether to either double their weight or remove them from the graph.
3. Iterate on the resulting graph for $O(\log \frac{m}{n})$ rounds.

0.1.2 Proving this algorithm works

Lemma 0.1. *At least $\frac{1}{2}$ of the edges in the graph have effective resistance less than $\frac{2(n-1)}{m}$.*

Proof. The average effective resistance of edges in a graph is $\frac{n-1}{m}$, via trace arguments. The rest follows from Markov's inequality. □

First, we state Matrix Bernstein, and a direct lemma of Matrix Bernstein **TIMOTHY: which may or may not be Matrix Azuma.**

Theorem 0.2. *(Matrix Bernstein) Let $X = \sum_{i=1}^{s} S_i$. The norm of the total variance of X is:*

$$\sigma^2 = \left|\left| \sum_{i=1}^{s} \mathbb{E}_{S_i}\left[S_i^2\right] \right|\right| \tag{1}$$

Then $Pr\left[||X|| \geq t\right] \leq d \cdot e^{\frac{-t^2}{\sigma^2 + Lt/3}}$ where d is the dimension of X and $L = max||S_i||$. Here, $||M||$ represents the spectral norm of M.

*(***TIMOTHY: Check if d is defined the right way***)*

Lemma 0.3. *Let $X = \sum_{i=1}^{s} S_i$ and $S_i := \mu_i M_i$ where M_i is a fixed PSd matrix and μ_i is a Rademacher random variable, then $Pr\left[||X|| \geq t\right] \leq d \cdot e^{\frac{-t^2}{L \cdot \mathbb{E}_{S_i}[X] + Lt/3}}$ where d is the dimension of X and $L = max||S_i||$. Here, $||Y||$ represents the spectral norm of Y.*

Proof. This follows from

$$\mathbb{E}_{S_i}\left[S_i^2\right] \preceq \max_i ||M_i|| \sum M_i \tag{2}$$

when $M_i \succeq 0$.

TIMOTHY: I'd much rather define a matrix absolute value and use $abs(S_i)$ rather than M_i □

Lemma 0.4. *Let G' be the output of the algorithm stated in 0.1.1. Then G' is with $\frac{1}{n^{O(1)}}$ probability an $1 + O\left(\sqrt{\frac{n \log n}{m}}\right)$ sparsifier, with high probability.*

Proof. Consider the Laplacian L_e corresponding to each edge of the graph G. Let E' be the set of edges in G with effective resistance less than $\frac{2(n-1)}{m}$.

Let S_e be the set of independent random variables such that: it is L_e with probability $\frac{1}{2}$ and $-L_e$ with probability $\frac{1}{2}$. Recall G' is the output of the algorithm in 0.1.1. Then

$$L_{G'} := \left(\sum_{e \in E} L_e\right) + \left(\sum_{einE'} S_e\right). \tag{3}$$

by construction. If we left and right multiply this equation by $L_G^{\dagger/2}$ and substitute $X_H := L_G^{\dagger/2} L_H L_G^{\dagger/2}$ for any graph H, and let $T_e := L_G^{\dagger/2} S_e L_G^{\dagger/2}$ then we get:

$$X_{G'} := \left(\sum_{e \in E} X_e\right) + \left(\sum_{e \in E'} T_e\right). \tag{4}$$

Note that $\mathbb{E}_{T_e}[X_{G'}] = I$ and T_e are independent random variables with expected value zero satisfying (by construction):

$$||T_e|| \leq \frac{2(n-1)}{m} \tag{5}$$

and

$$\sum_{e \in E'} L_e \preceq I \tag{6}$$

then we can apply lemma 0.3 to get:

$$\Pr\left[||X_{G'}|| \geq t\right] \leq n \cdot e^{\frac{-t^2}{\frac{2(n-1)}{m} + \frac{2(n-1)t}{m}}} \tag{7}$$

Note that if $t = O\left(\sqrt{\frac{n \log n}{m}}\right)$ (and $t \leq 1$), then

$$n \cdot e^{\frac{-t^2}{\frac{2(n-1)}{m} + \frac{2(n-1)t}{m}}} = \frac{1}{n^{O(1)}} \tag{8}$$

thereby proving that

$$||X_{G'} - I|| \leq O\left(\sqrt{\frac{n \log n}{m}}\right) \tag{9}$$

with polynomially high probability. which proves that $L_{G'}$ is a $1 + O\left(\sqrt{\frac{n\log n}{m}}\right)$ sparsifier of L_G with high probability. $\square$

TIMOTHY: I will do a major cleanup on this part... Some words about how chaining this lemma we just proved, with log n iteration, causes error to blow up geometrically each time, with factor $\sqrt{2}$ which is fine, making the total error ϵ thanks to convergence of geometric series.

Likewise, the run time of a single iteration of the algorithm described in 0.1.1 is also nearly linear in m, and is thus dominated by the first iteration (as edges decreases by a $\frac{1}{2}$ factor each time.

0.2 Obtaining a degree-preserving sparsifier of size $O\left(\frac{nl\log n}{\varepsilon^2}\right)$ in X time, given an $(O(n), l)$ short-cycle decoposition

TIMOTHY: This is a very very rough shoddy writeup. The key part is that I believe you only pick up an l factor in space. Note that I think you can do better, in that I think you can pick up some sublinear function of l because I naively bounded the effective resistance of a cycle to be the sum of the effective resistances of the edges, and also did some matrix absolute value shenanigans under the hood when bounding the variance parameter. Both if these assumptions seem a little silly.

Cycles change nothing except the effective resistance calculation (and now we sparsify down to $\frac{3}{4}$ as many edges, in each step of te algorithm, which ultimately changes nothing).

This throws in an extra l factor into the variance squared, **TIMOTHY: not an l^2 factor, which is what I initially expected**.

Therefore, t is now $= O\left(\sqrt{\frac{nl\log n}{m}}\right)$ (and $t \leq 1$), and so the final number of edges m' we can sparsify to while incurring an ϵ^2 error, satisfies:

$$\varepsilon = O\left(\sqrt{\frac{nl\log n}{m'}}\right) \tag{10}$$

or

$$\varepsilon^2 = O\left(\frac{nl\log n}{m'}\right) \tag{11}$$

or

$$m' = O\left(\frac{nl\log n}{\varepsilon^2}\right) \tag{12}$$

The runtime is more than linear in m, so since the number of edges decreases geometrically at each iteration of our coin-tossing algorithm, the total run time is a constant factor of the runtime of the first iteration.

That is to say, the total run time is the time it takes to run a short cycle decomposition and estimate effective resistances on each cycle.

TIMOTHY: I gave some thought into how degree preserving property might help us get a better sparsfier than the naive l factor in space we pick up, where l is the length of the short cycles in a short-cycle decomposition. I had no idea how to proceed, for a couple good reasons I will not go into here.