ORIE 6334 Spectral Graph Theory	September 20, 2016
Lecture 9	
Lecturer: David P. Williamson	*Scribe: Michael Roberts*

In this lecture, we develop and and analyze a randomized approximation algorithm for MAX CUT. Recall the MAX CUT problem: Given $G = (V, E)$, find $S \subset V$ that maximizes $\delta(S)$.

Definition 1 (Approximation algorithm) *A (randomized) α-approximation algorithm runs in (randomized) polynomial time and computes a solution with (expected) value within α of the value of an optimal solution.*

Note that there exists an easy randomized algorithm: Flip a coin for each $i \in V$ to decide whether or not $i \in S$. Then

$$\mathbb{E}\left[|\delta(S)|\right] = \sum_{(i,j)\in E} \Pr[(i,j) \in S] = \frac{1}{2}|E| \geq \frac{1}{2}\,\mathrm{OPT},$$

where OPT is the value of an optimal solution to **Max-Cut** on G.

Today, we will show a .529-approximation algorithm due to Trevisan using a combination of this naive randomized algorithm and Trevisan's Cheeger-like inequalities.

Recall from the previous lecture that we defined

$$\beta(S) = \min_{(L,R)\ a\ partition\ of\ S} \frac{2|E(L)| + 2|E(R)| + |\delta(S)|}{\mathrm{vol}(S)}$$

and

$$\beta(G) = \min_{\substack{S\subset V\\ S\neq\emptyset}} \beta(S).$$

Let β_n is the smallest eigenvalue of $I + \mathcal{A}$, where $\mathcal{A}$ is the normalized adjacency matrix of G. Last time we showed the following.

Theorem 1 (Trevisan 2009)

$$\frac{1}{2}\beta_n \leq \beta(G) \leq \sqrt{2\beta_n}.$$

We note that the proof was algorithmic; given the eigenvector corresponding to the eigenvalue β_n, the algorithm returns a set S and a partition of S into L and R such that $\beta(S) \leq \sqrt{2\beta_n}$.

1 Trevisan's Algorithm for MAX CUT

The main idea of this algorithm is to trade off between two cases:

- If OPT $< (1-\epsilon)|E|$, then we get an approximation ratio from the naive random algorithm that is better than 1/2.
- If OPT $\geq (1-\epsilon)|E|$, then we can use Trevisan's inequality to get a better bound.

For Max Cut S^*, let $S = V$, $L = S^*$, $R = V - S^*$. Suppose that OPT $\geq (1-\epsilon)|E|$. Then

$$\begin{aligned}
\beta(G) \leq \beta(S) &= \frac{2|E(S^*)| + 2|E(V-S^*)| + |\delta(V)|}{\mathrm{vol}(V)} = \frac{2(|E| - |\delta(S^*)|)}{2|E|} \\
&\leq \frac{2(|E| - (1-\epsilon)|E|)}{2|E|} \\
&= \epsilon.
\end{aligned}$$

[0]This lecture is derived from Lau, Lecture 4 `https://cs.uwaterloo.ca/~lapchi/cs798/notes/L04.pdf`.

Notice that in this case, then, we can infer that $\beta_n \leq 2\epsilon$.

So if $\beta_n > 2\epsilon$, then OPT $< (1-\epsilon)|E|$. So the naive randomized algorithm finds S such that

$$\mathbb{E}\left[\delta(S)\right] = \frac{1}{2}|E| \geq \frac{\text{OPT}}{2(1-\epsilon)}.$$

Thus in this case it is a $\frac{1}{2(1-\epsilon)}$- approximation algorithm.

Now suppose that $\beta_n \leq 2\epsilon$. We can run the algorithm to find a set S and a partition of S into L and R such that $\beta(S)$ is small, namely, at most $\sqrt{2\beta_n} \leq 2\sqrt{\epsilon}$.

Once we have this S, what should we do to find a large cut? In this case, we will attempt to improve our bounds by making some recursive calls. We recurse our Max-Cut algorithm on $V - S$, to find (L', R') that partition $S - V$.

Consider the following two possible cuts of G (presented as partitions on V):

- $(L \cup L', R \cup R')$
- $(L \cup R', R \cup L')$

Notice that every edge in $\delta(S)$ either "stays on the same side", going from L to L' or R to R', or else "crosses sides", going from L to R' or R to L'. That means that one of the above cuts must contain at least 1/2 the edges in $\delta(S)$. We choose that cut.

Call the size of the cut our algorithm finds on G, ALG(G), and the size of the maximum cut in G, OPT(G). Then:

$$\text{ALG}(G) \geq |\delta(L,R)| + 1/2\delta(S) + \text{ALG}(G-S),$$

and

$$\text{OPT}(G) \leq |E(L)| + |E(R)| + |\delta(L,R)| + |\delta(S)| + \text{OPT}(G-S).$$

Then

$$\frac{\text{ALG}(G)}{\text{OPT}(G)} \geq \min\left\{\frac{|\delta(L,R)| + 1/2\delta(S)}{|E(L)| + |E(R)| + |\delta(L,R)| + |\delta(S)|}, \frac{\text{ALG}(G-S)}{\text{OPT}(G-S)}\right\}.$$

Since $\beta_n \leq 2\epsilon$, using Trevisan's inequalities we bound:

$$\begin{aligned}
2\sqrt{\epsilon} &\geq \frac{2|E(L)| + 2|E(R)| + |\delta(S)|}{\text{vol}(S)} \\
&= \frac{2|E(L)| + 2|E(R)| + |\delta(S)|}{2|E(L)| + 2|E(R)| + |\delta(S)| + 2|\delta(L,R)|} \\
&= 1 - \frac{|\delta(L,R)|}{|E(L)| + |E(R)| + 1/2|\delta(S)| + |\delta(L,R)|}.
\end{aligned}$$

Thus

$$\frac{|\delta(L,R)| + 1/2\delta(S)}{|E(L)| + |E(R)| + |\delta(L,R)| + |\delta(S)|} \leq \frac{|\delta(L,R)|}{|E(L)| + |E(R)| + |\delta(L,R)| + 1/2|\delta(S)|} \leq 1 - 2\sqrt{\epsilon}.$$

So, we can conclude that

$$\frac{\text{ALG}(G)}{\text{OPT}(G)} \geq \min\left\{1 - 2\sqrt{\epsilon}, \frac{\text{ALG}(G-S)}{\text{OPT}(G-S)}\right\}.$$

The same must hold true for $G-S$ recursively. But note that for some subgraph of G we consider in some recursive step, it may be possible that $\beta_n \geq 2\epsilon$. Thus we conclude that:

$$\frac{\text{ALG}(G)}{\text{OPT}(G)} \geq \min\left\{1 - 2\sqrt{\epsilon}, \frac{1}{2(1-\epsilon)}\right\}.$$

These two expressions are equal for $\epsilon \approx .0554$, at which point the ratio is about .529. So this is a .529-approximation algorithm.[1]

Better analyses were given in Trevisan 2009, which improved the bound to .531, and in Soto 2015, which improved it to .614.

2 Discussion

Goemans, W (1995) gave a .878-approximation algorithm for MAX CUT by using semidefinite programming (SDP). So why do we care about Trevisan's spectral algorithm?

- Computing eigenvectors is a lot easier than solving SDP. (Although, Trevisan's algorithm makes recursive calls that require recomputing new vectors).
- This method may be more powerful than LP. Chan, Lee, Raghavendra, Steurer FOCS '13 shows you need superpolynomial-sized LPs to do better than a 1/2-approximation algorithm. In a forthcoming paper Kothari, Meka and Raghavedra, this result is improved to showing that exponentially-sized LPs are required to get better than a 1/2-approximation algorithm.

These observations raise some research questions:

- The current bound on the algorithm's performance doesn't seem tight - is it?
- Is there a"one-shot" spectral algorithm, one that doesn't require recursive calls? The recursion makes it hard to analyze the algorithm, and forces recomputation of eigenvectors.
- Can we apply this algorithm to other problems with a similar structure (called 2-CSP)? For instance, the MAX DICUT problem (MAX CUT in directed graphs) and the MAX 2SAT problem have this structure. In the MAX 2SAT problem, we are given n boolean variables $x_1, \ldots, x_n$, and some number of clauses with at most two variables (e.g. $\bar{x}_1$, $x_2 \vee \bar{x}_3$, etc.) The goal is to find a setting of the variables to true or false so as to maximize the total number of satisfied clauses.

Some progress has been made on this last question.

Definition 2 (Balanced MAX E2SAT) *Balanced MAX E2SAT is a subclass of MAX 2SAT instances such that each clause has exactly two literals in it (i.e. variables or their negations) and for all i, the number of clauses in which x_i appears is exactly equal to the number of clauses in which $\bar{x}_i$ appears.*

Paul, Poloczek, W (2016) use Trevisan's algorithm to obtain a .81-approximation algorithm for Balanced MAX E2SAT, which is better than a .75-approximation algorithm that can be obtained via a naive randomized algorithm.

3 Other Cheeger-Like Inequalities

We previously claimed that $\lambda_k(L_G) = 0$ iff G has at least k connected components, and made a similar clame for $\lambda_k(\mathcal{L})$. So, we may be interested in Cheeger-like inequalities for λ other than λ_2. We define the *k-way conductance* of a graph G as

$$\phi_k(G) = \min_{\substack{S_1, S_2 \ldots S_n \subset V \\ All\ S_i\ disjoint}} \max_i \phi(S_i).$$

Some relatively recent papers have proved a bound on the k-way conductance via λ_k. These inequalities are called *higher-order Cheeger inequalities.*

[1] Lau, in his lecture notes, attributes this analysis to Nick Harvey.

Theorem 2 (Lee, Oveis Gharan, Trevisan '12)

$$\frac{\lambda_k}{2} \leq \phi_k(G) \leq O(k^2)\sqrt{\lambda_k}.$$

The following has also been shown, in which the dependence on k is improved, but the eigenvalue in the inequality is weakened to be λ_{2k} rather than λ_k.

Theorem 3 (Lee et al '12; Louis, Raghavedra, Tetali, Vempala '12)

$$\phi_k(G) = O(polylog(k))\sqrt{\lambda_{2k}}.$$

But it is an open question whether or not one can have both things at once; that is, whether one can show that

$$\phi_k(G) = O(polylog(k))\sqrt{\lambda_k}.$$