Advanced Approximation Algorithms **(CMU 18-854B, Spring 2008)**

Lecture 16: Gaps for Max-Cut

Mar 6, 2008

Lecturer: Ryan O'Donnell *Scribe: Ravishankar Krishnaswamy*

1 Outline

In this lecture, we will discuss algorithmic (Goemans-Williamson) and integrality gaps for the Max-Cut problem. We begin by recalling the semidefinite programming formulation for Max-Cut.

$$\max \sum_{(u,v)\in E} w_{uv}(\frac{1}{2} - \frac{1}{2}\vec{u}\cdot\vec{v})$$
$$\text{subject to } \vec{v}\cdot\vec{v} = 1 \ \forall v \in V$$

Let us denote the optimal value of the SDP by $\mathsf{Sdp}(G)$. Also, let $\mathsf{Opt}(G)$ be the value of the maximum cut in G, and $\mathsf{Alg}_{GW}(G)$ be the expected value of a cut which the Goemans Williamson algorithm returns.

For any given graph G, we know the following to be true:

$$\mathsf{Sdp}(G) \geq \mathsf{Opt}(G) \geq \mathsf{Alg}_{GW}(G)$$

Also, we know the following as well,

Theorem 1.1. *[2] For all graphs G, $\mathsf{Alg}_{GW}(G) \geq \alpha_{GW}\mathsf{Sdp}(G)$, where α_{GW} is a constant ≈ 0.878.*

But this only gives us a ratio comparing $\mathsf{Sdp}(G)$ and $\mathsf{Alg}_{GW}(G)$, and is thus a worry in the following sense: If the graph is such that $OPT(G) \approx 0.5$, the Goemans Williamson algorithm could actually return a cut of size ≤ 0.5 (if $\mathsf{Sdp}(G)$ were also nearly 0.5), performing worse than any random cut. This leads us to the following questions:

1. How much more can $\mathsf{Sdp}(G)$ be when compared to $\mathsf{Opt}(G)$?
2. Is the analysis of Goemans and Williamson actually tight ? Is there an instance where $\mathsf{Alg}_{GW}(G) = \alpha_{GW}\mathsf{Opt}(G)$?

Let's first recap their algorithm: After solving the SDP, they generate a random hyperplane, and partition the vertices based on which side of the hyperplane their associated vectors lie. Therefore, $\mathrm{P}[(u,v) \text{ is cut}] = \arccos(\rho_{uv})/\pi$, whereas the contribution to $\mathsf{Sdp}(G)$ by the edge (u,v) is $\frac{1}{2} - \frac{1}{2}\rho_{uv}$. Here, ρ_{uv} is the dot product $\vec{u}\cdot\vec{v}$. A comparison of the contribution of an edge (u,v) (such that an embedding has $\vec{u} \cdot \vec{v} = \rho$) towards the SDP with the expected contribution to the goemans williamson cut is shown in figure 1.

Hence, if a non-negligible fraction of the edges (u, v) have their corresponding end-points' dot products ρ_{uv} distant from $\rho^* = -0.69$ (see figure 1), we notice that it is possible to obtain better guarantees on the approximation factor.

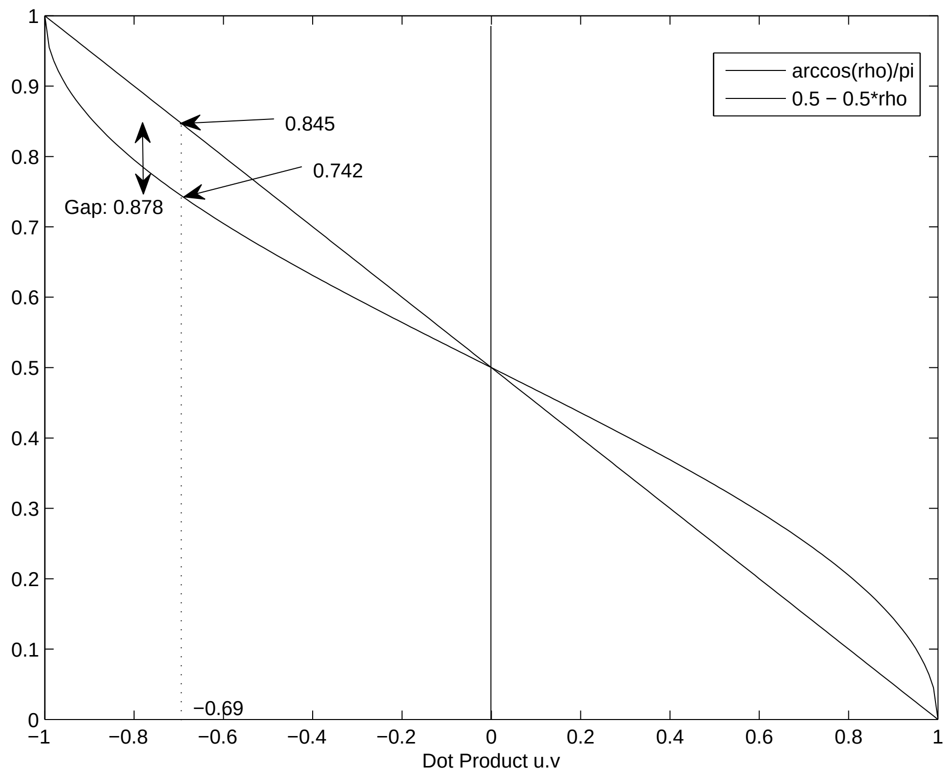

Figure 1: The SDP vs GW Algorithm Values Comparison Curve

2 Integrality Gap

Getting back to the questions we had in mind, the first one was resolved by Feige and Schehtman [1] where they established a tight integrality gap instance for this SDP formulation.

Theorem 2.1. [Feige and Schechtman [1]] *For every constant ϵ, there exists a graph G such that* $\mathsf{Opt}(G) \leq (\alpha_{GW} + \epsilon)\mathsf{Sdp}(G)$

Note 2.2. *On such graphs G, the Goemans Williamson algorithm actually finds the optimal cut !*

We shall now give a sketch of the construction used in [1] to obtain the integrality gap instance. The idea is to embed a graph on a sphere (thereby giving a feasible SDP solution — this will lower bound $\mathsf{Sdp}(G)$), and then prove that $\mathsf{Opt}(G)$ on this instance is not high. Here are some useful notations.

Definition. An "embedded graph" is a weighted graph $G = (V, E)$ where $V \subseteq \mathcal{S}^{d-1}$ for some $d \in \mathbb{N}$, where $\mathcal{S}^{d-1}$ is the surface of a d dimensional sphere, i.e $\mathcal{S}^{d-1} = \{x \in \mathbb{R}^d : \|x\| = 1\}$.

Definition. Given such an embedded graph, define $\mathsf{Obj}(G)$ to be $\sum_{(u,v)\in E} w_{uv}(\frac{1}{2} - \frac{1}{2}\vec{u}\cdot\vec{v})$.

It is simple to see the following fact (as the embedding itself is one feasible solution to the SDP).

Fact 2.3. *For all embedded graphs G, $\mathsf{Obj}(G) \leq \mathsf{Sdp}(G)$.*

Let us denote the value of the dot product ρ which actually minimizes $\frac{\frac{\arccos\rho}{\pi}}{\frac{1}{2}-\frac{1}{2}\rho}$ be denoted by ρ^*. As marked in the figure 1, $\rho^* \approx -0.69$.

To get our instance, we try to find an embedding such that $\mathsf{Obj}(G) \gg \mathsf{Opt}(G)$. Since we need the gap between the Goemans Williamson Algorithm (on the integrality gap instance we hope to create, $\mathsf{Alg}_{GW}(G) = \mathsf{Opt}(G)$) and the SDP value to be 0.878, we want all our edges (u, v) included in the embedded graph to satisfy the property $\vec{u}\cdot\vec{v} \approx \rho^*$. If we accomplish this, we would ensure that $\mathsf{Sdp}(G) \geq \mathsf{Obj}(G) \geq \frac{1}{2} - \frac{1}{2}\rho^*$, while the expected value of the Goemans Williamson algorithm when run on our embedding is $\approx \frac{\arccos\rho^*}{\pi}$. We would then be done if we show that $\mathsf{Opt}(G)$ is no more than this quantity.

The following will be a sketch of the instance we construct. It will be an infinite graph $G^* = (V, E)$ with the vertices being all points on the surface of the d dimensional sphere (i.e $V = S^{d-1}$). And the idea suggested above was to put in all possible edges whose embedded vertices have a dot product of ρ^* between them. As the instance we are creating is an infinite (the vertex set is, in fact, even uncountable) graph it becomes necessary to think of the edges as being a probability distribution over pairs of points on the sphere (symmetric distribution over $V \times V$). We therefore slightly redefine our notions of Obj and Opt as follows:

$$\mathsf{Obj}(G^*) = \mathop{\mathbf{E}}_{(u,v)\sim E}\left[\frac{1}{2} - \frac{1}{2}\vec{u}\cdot\vec{v}\right]$$

and

$$\mathsf{Opt}(G^*) = \max_{f:V\to\{-1,1\}} \mathop{\mathbf{Pr}}_{(u,v)\sim E}[f(u) \neq f(v)]$$ [1]

Now we actually define the edge distribution. E is the distribution such that a random draw from E amounts to picking u and v randomly and independently, but conditioned on $\vec{u}\cdot\vec{v}$ being at most ρ^*. We now present a fact about the tail bounds on this random distribution.

Fact 2.4. *For large values of d, $\mathbf{E}[\vec{u}\cdot\vec{v} | \vec{u}\cdot\vec{v} \leq \rho^*] \cong \rho^*$.*

[1] We are overlooking a technical issue about the usage of $\max$ here as we are not discussing the existence of the maximum value over our domain

In fact, a lower bound of $\rho^* - O(\frac{1}{\sqrt{d}})$ can be shown.

By virtue of the way we have defined the distribution E, $\mathsf{Obj}(G^*) \geq \frac{1}{2} - \frac{1}{2}\rho^*$. What about $\mathsf{Opt}(G^*)$?

Let the optimal cut of the surface of the sphere be (A, A^C) where $A \subseteq S^{d-1}$ has a fractional surface area a (this is the measure of A). The notion of a measure of a set is the fraction of vertices it has, and the function $\mu_\rho(A)$ (on the product measure space) represents the fraction of the edges crossing the set A. That is, $\mu_\rho(A) = Pr_{(u,v)\sim E}[A(u) \neq A(v)]$, where $A(u) = 1$ if and only if $u \in A$. The hardest part of the integrality gap proof is the following theorem.

Theorem 2.5. [Theorem 4, [1]] *Fix an a between 0 and 1 and a ρ between 0 and -1. Then the maximum of $\mu_\rho(A)$ where A ranges over all (measurable) subsets of S^{d-1} of measure a is attained for a(ny) cap of measure a.*

Essentially, it proves that the best cut is a cap (a cap is the portion of the sphere's surface above some hyperplane; see figure 2). In fact, Feige and Schechtman go on to prove that the best cap is a hemisphere (where the hyperplane defining the cap passes through the origin). Here is the intuition as to why. For any cap C, if we consider H to be the hemisphere centered at the same point as the center of C, we can observe that for point any p in $H \notin C$, the measure of $(\{p\}, H^C)$ (the edges from p to H^C) is at least as much as the measure of $(\{p\}, C)$ (by symmetry of H with respect of C, for every edge from p to some point in C, there is a mirror image in H^C to which p will have an edge). Thus, including all such p into C would not decrease the measure of the new set. Hence, the hemisphere has at least as much measure as that of the cap. And by symmetry of G^*, all hemispheres will have the same value. This means that the value of the optimal cut on the graph is at most the expected value of the Goemans Williamson algorithm (since all hyperplanes passing through the origin have the same cut measure) when run on our embedding ! Since we have the expected dot product $\vec{u} \cdot \vec{v} \approx \rho^*$ when $(u, v) \sim E$, the expected value of the Goemans Williamson cut is approximately $\frac{\arccos \rho^*}{\pi}$. Therefore,

$$\mathsf{Opt}(G) \leq \frac{\arccos \rho^*}{\pi} + o(1)$$

Hence,

$$\frac{\mathsf{Opt}(G^*)}{\mathsf{Sdp}(G^*)} \leq 0.878 + o(1)$$

This establishes the integrality gap result.[2] We can then get a finite graph from this by discretizing the sphere into small portions of size $O(\epsilon)$, and choosing the small portions on the surface as vertices. We would get the integrality gap at $0.878 + \epsilon$.

[2]The reason why we don't define the edge distribution to be such that the dot product is exactly ρ^* is that the argument that establishes any cap to be the optimal cut breaks down.

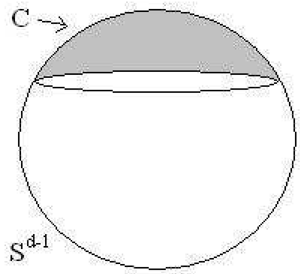

Figure 2: C is the cap on the d dimensional sphere

3 Algorithmic Gap

The following section is based on a result by Karloff [3]. We now focus on the second question — that of the algorithmic gap: Are there instances G where $\mathsf{Alg}_{GW}(G) \leq (\alpha_{GW} + \epsilon)\mathsf{Opt}(G)$ (for any constant ϵ)? Note that such instances would imply that $\mathsf{Sdp}(G) = \mathsf{Opt}(G)$. But a point to note here is that $\mathsf{Alg}_{GW}(G)$ is a quantity which actually depends on the particular SDP embedding. Should we therefore strive to show $\mathsf{Alg}_{GW}(G) \leq 0.878\mathsf{Sdp}(G)$ for all optimal SDP embeddings ? No — as the algorithm would perform optimally on the magic hidden SDP embedding of the integral cut solution ! Therefore, we just want to get *some* embedding whose value is $\mathsf{Sdp}(G)$, such that the Goemans Williamson algorithm performs poorly on that. Then we could argue that the SDP could have unfortunately output that optimal embedding, and the algorithm would get a much smaller cut than the optimal. Let's give such an instance.

Recalling the LP integrality instance we created for Max-Coverage, we again use the notion of binary strings corresponding to vertices. The graph G has vertex set $V = \{-\frac{1}{\sqrt{k}}, \frac{1}{\sqrt{k}}\}^k$, a discrete cube embedded on the sphere of dimension k. Note that on these vertices, the quantity $\frac{1}{2} - \frac{1}{2}\vec{u} \cdot \vec{v}$ is equal to the fractional hamming distance between the vectors $\vec{u}$ and $\vec{v}$.

Now for the edges, we present a random experiment which defines the probability distribution over the edges (which will be our weighted distribution of our edges): pick a vertex $\vec{u} \in \{-\frac{1}{\sqrt{k}}, \frac{1}{\sqrt{k}}\}^k$ uniformly at random, and form $\vec{v}$ by negating each coordinate of $\vec{u}$ with probability $\frac{1}{2} - \frac{1}{2}\rho^*$, and set (u, v) to be an edge[3].

Note that the expected dot product $\mathbf{E}_{(u,v)\sim E}[\vec{u} \cdot \vec{v}] = \rho^*$!

[3]This actually sets a tiny probability on self loops at vertices, which we neglect in the proof (but this issue can be fixed)

Also,

$$\begin{aligned}
\mathsf{Obj}(G) &= \mathop{\mathbf{E}}_{(u,v)\sim E}\left[\frac{1}{2}-\frac{1}{2}\vec{u}\cdot\vec{v}\right] \\
&= \frac{1}{2}-\frac{1}{2}\mathop{\mathbf{E}}_{(u,v)\sim E}[\vec{u}\cdot\vec{v}] \\
&= \frac{1}{2}-\frac{1}{2}\mathop{\mathbf{E}}_{(u,v)\sim E}\left[\sum_{i=1}^{k}u_iv_i\right] \\
&= \frac{1}{2}-\frac{1}{2}\rho^*
\end{aligned}$$

Further, since our edge distribution is an independent experiment on pairs of vertices with the expected value of $\vec{u}\cdot\vec{v}=\rho^*$, the random edge from this distribution will have its dot product concentrated (we can use chernoff bounds to bound the tails on both sides of the mean) about the mean (i.e $\vec{u}\cdot\vec{v}\approx\rho^*+o(1)$), where the $o(1)$ term decreases with higher k. Hence, the Goemans Williamson algorithm (on this embedding) would return a cut of size at most $\frac{\arccos\rho^*}{\pi}+o(1)$. We therefore need to show: (1) that there exists an integral cut of value $\mathsf{Obj}(G)$ (this would establish $\mathsf{Opt}(G)\geq\mathsf{Obj}(G)$), and (2) that the embedding we have produced is actually optimal to the SDP! (i.e we need to show that $\mathsf{Sdp}(G)\leq\mathsf{Obj}(G)$).

If we establish these, we would have proved that there is a graph G where $\mathsf{Opt}(G)=\mathsf{Sdp}(G)=\mathsf{Obj}(G)=\frac{1}{2}-\frac{1}{2}\rho^*$, and that there is an SDP optimal embedding of G on which the Goemans Williamson algorithm fares poorly. We would then be done !

We discuss the first part in this lecture, and leave the next part to after spring break. To show that there are solutions with value $\mathsf{Obj}(G)$, we revert back to *dictator cuts* ! That is, we cut the cube into 2 parts, based on any coordinate (say i). Let's denote this cut by $f_i:V\rightarrow\{-1,1\}$. The value of this cut is,

$$\begin{aligned}
Val &= \mathop{\mathbf{Pr}}_{(u,v)\sim E}[f_i(u)\neq f_i(v)] \\
&= \mathop{\mathbf{Pr}}_{(u,v)\sim E}[u_i\neq v_i] \\
&= \frac{1}{2}-\frac{1}{2}\rho^*
\end{aligned}$$

where the last step follows by the definition of our edge distribution. This actually shows that there are k such good cuts, but unfortunately, starting from the embedding we've constructed, the Goemans Williamson algorithm performs poorly.

It remains to show that our embedding is actually optimal, that is, the SDP optimizer could unfortunately output it.

We now conclude this lecture by describing what we shall show in future lectures, after introducing fourier analysis.

$$\forall\ F:\{-\frac{1}{\sqrt{k}},\frac{1}{\sqrt{k}}\}^k\rightarrow S^{d-1}\ ,\ \mathop{\mathbf{E}}_{(u,v)\sim E}\left[\frac{1}{2}-\frac{1}{2}Fu.Fv\right]\leq\frac{1}{2}-\frac{1}{2}\rho^*$$

References

[1] U. Feige and G. Schechtman. On the optimality of the random hyperplane rounding technique for max cut. *Random Struct. Algorithms*, 20(3):403–440, 2002.

[2] M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, 1995.

[3] H. Karloff. How good is the goemans–williamson max cut algorithm? *SIAM J. Comput.*, 29(1):336–350, 2000.