CS 354: Unfulfilled Algorithmic Fantasies April 29, 2019

Lecture 9: Unique Games Hardness of Vertex Cover

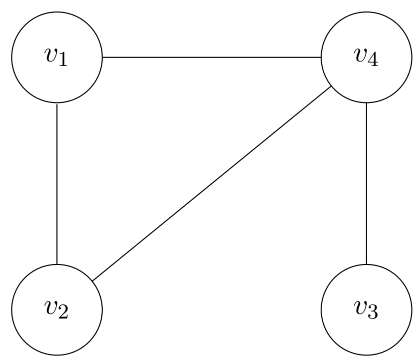
Lecturer: Aviad Rubinstein *Scribe: William Marshall*

In this lecture, we prove the conditional hardness of the Vertex Cover problem by relating it to the Independent Set problem and reducing Unique Games to Independent Set.

1 Minimum Vertex Cover

Definition 1.1. Let $G = V, E$ be a graph. A *vertex cover* is a set $S \subseteq V$ such that for all edges $(u, v) \in E$, $u \in S$ or $v \in S$ (or both).

In the Minimum Vertex Cover problem (often shortened to Vertex Cover), the objective is to find the smallest vertex cover for a given graph. For example, in the following graph the size of the minimum vertex cover is 2 since no single vertex is incident to every edge, but the set $\{v_2, v_4\}$ contains vertices which combined are incident to every edge and so form a vertex cover.



The Vertex Cover problem is NP-hard, so we can not hope to solve it exactly. However, we can easily achieve a 2-approximation by iteratively choosing an uncovered edge and adding both endpoints of the edge to the vertex cover. At least one vertex from each of these pairs must be included in any valid vertex cover, so the result is at worst off by a factor of 2. It is natural to ask if it is possible to beat the benchmark set by this trivial approach. The purpose of this lecture is to show that assuming the Unique Games Conjecture the answer is "no".

Theorem 1.2 ([KR08])**.** *Assuming the Unique Games Conjecture, it is impossible to approximate Vertex Cover within a factor of* $2 - \epsilon$.

2 Maximum Independent Set

Instead of reducing Unique Games directly to Vertex Cover, we will instead go through the intermediate Independent Set problem.

Definition 2.1. Let $G = (V, E)$ be a graph. A set $I \subseteq V$ is a *independent set* of G if for all $u, v \in I$, $(u, v) \notin E$.

In the Independent Set problem, the objective is to find the largest possible independent set. This is related to the Vertex Cover problem: notice that in the example above, the complement of the vertex cover, $\{v_1, v_3\}$ forms a maximum independent set. This is not a coincidence: in general, there can not be an edge between vertices not in a vertex cover, otherwise the edge would not be covered. Therefore, the vertices not in a vertex cover form an independent set. Indeed, it is not hard to see that if I and S are the maximum independent set and minimum vertex cover of some graph on n vertices, then $|S| + |I| = n$.

Theorem 2.2. *Assuming the Unique Games Conjecture, it is hard to distinguish between*

- *(YES) The size of the maximum independent set is at least* $n/2 - \epsilon$*,*
- *(NO) The size of the maximum independent set is at most* ϵn*.*

Theorem 2.2 implies Theorem 1.2 since by the relation between independent sets and vertex covers if we could distinguish between vertex cover of size close to n and vertex cover close to $n/2$ (i.e. if we had better than a 2-approximation to vertex cover), then by taking the compliments of the vertex covers found we could distinguish between independent set of size close to 0 and independent set of size close to $n/2$. Therefore, we will spend the majority of our effort proving theorem 2.2.

3 Reduction Sketch

Just as with the Max Cut problem, for each vertex of Unique Games, we create a vertex cloud in the Independent Set instance corresponding to $\{0, 1\}^\Sigma$ (where Σ is the alphabet for the Unique Games instance).

The edges are constructed differently: for vertices $u_x, v_y \in V$, we have $(u_x, v_y) \in E$ if and only if $x \cap \pi_{u,v}(y) = \emptyset$ where we view x and y as sets and $\pi_{u,v}$ is the permutation/constraint between u and v. We will soon see why this is the case.

Finally, we use a special weighting on the vertices, corresponding to the *biased long code.*

4 Comparison to Max Cut

From the above high level overview, we see that this reduction will share many similarities with the reduction to Max Cut, so we begin by comparing the two problems to gain intuition.

	Max Cut	Independent Set
constraint	$f(x) \neq f(y)$	$\neg(f(x) \wedge f(y))$
satisfaction requirement	satisfy as many constraints as possible	must satisfy all constraints
test	$f(x) \neq f(-x + \text{noise})$	$\exists i \in \Sigma$ s.t. $x_i = y_i = 1 \implies \neg(f(x) \wedge f(y))$

To start with, in the Independent Set problem we require that we do not include both endpoints of an edge in the independent set, so the constraint takes a different form. More consequential, however, are the requirements for how many constraints must be satisfied. In the Max Cut reduction, we were satisfied with satisfying some large fraction of the constraints, but in the Independent

Set problem, the output must be a valid independent set, so we must satisfy all the constraints. This matters when considering the test used. In max cut, adding noise to the inputs allowed us to circumvent possible "bad" solutions in the analysis of soundness. However, since we need to satisfy all the constraints for the Independent Set problem, this randomness is hard to control for our purposes.

5 Dictatorship vs Majority

How good is our new test? As we discussed last time, it is not enough that the tests verifies the UG constraints between colors encoded as our error correcting code, the Long Code (we'll later see it does a good job with that). In order for the soundness analysis to go through, we also need to make sure that the function/independent set actually corresponds to a valid encoding of a color. Any function that doesn't look at all like a valid codeword should correspond to a much smaller independent set.

For reference, we begin with considering the size of independent set corresponding to valid Long Code codewords, aka "dictatorships" of the form $f_i(x) := x_i$. First, notice that any such f satisfies all the constraints. The (relative) size of the independent set is given by $\Pr_x[f_i(x)] = 1/2$, which is roughly what we wanted for our completeness.

Let's contrast that with "majority" $Maj(x) := sign\left(\sum x_i - |\Sigma|/2\right)$. Assuming $|\Sigma|$ is odd, this function also satisfies all the constraints (verify!). Also, we still have that the independent set has size $\Pr_x[Maj(x)] = 1/2$. But this function is really bad for our purposes because it is symmetric with respect to Σ and so doesn't tell us anything about the UG instance.

6 p-biased Long Code

Motivated by the above discussion, we want to significantly decrease the size of the majority independent set, without significantly affecting the dictatorship independent sets. Our graph already has all the constraints/edges it could possibly have without breaking any of the valid independent sets. Since we can't add any more edges (and removing them certainly won't help), we should look at the vertices instead...

This motivates a slight modification to the long code. For $p \in [0, 1]$, define

$$w(x) := p^{\sum x_i} \cdot (1-p)^{\sum(1-x_i)}.$$

We think of this w as a weight on each vertex. Alternatively, w corresponds to the probability of picking a random $x \in \{0,1\}^{\Sigma}$, where each $x_i = 1$ with probability p. Under this weighting scheme, the majority function now induces an independent set that is exponentially small. As with Max Cut, we can generalize beyond the majority function. This is made formal in the following lemma.

Lemma 6.1. *For all $\epsilon, s > 0$ and all $p \in (0, 1-\epsilon)$ and all monotone functions $f : \{0,1\}^{\Sigma} \to \{0,1\}$, there exists $q \in [p, p+\epsilon]$ for which there exists a $c(\epsilon, \delta, p)$-junta $\hat{f}$ such that*

$$Pr_{x \sim q}[f(x) \neq \hat{f}(x)] < \delta.$$

To put it a bit imprecisely, under some weighting close to that created by p, every monotone function can be approximated by a junta of some constant size. The proof of this lemma is beyond the scope of our class.

7 Analysis of Reduction

We are now ready to fill in the final details of the reduction and analyze it. As described above, given an instance of the Unique Games problem we create a cloud for every vertex and add an edge between two vertices u_x and v_y if and only if $x \cap \pi_{u,v}(y) = \emptyset$. The reduction also has parameter $p = \frac{1}{2} - \epsilon$ that defines the vertex weighting.

7.1 Completeness

We actually consider a reduction from a variant of the UGC where in the YES case, there is an assignment σ satisfies *all* the constraints on $(1-\epsilon)$ fraction of the variables. (This variant is equivalent to the UGC; we will skip the proof.)

Let $I = \{u_x : f_{\sigma(u)}(x) = x_{\sigma(u)} = 1\}$ where $f_{\sigma(u)}$ is the long code encoding of $\sigma(u)$. For UG-vertices u not assigned a color by Σ we do not add any IS-vertices to I.) We claim that I is an independent set since

$$\begin{aligned} \text{constraint } (u,v) \text{ satisfied} &\Leftrightarrow \sigma(u) = \pi_{u,v}(\sigma(v)) \\ &\Leftrightarrow \forall u_x, v_y \in I, \sigma(u) \in x \cap \pi_{u,v}(y) \\ &\Leftrightarrow (u_x, v_y) \notin E \end{aligned}$$

7.2 Soundness

For some independent set I, suppose $w(I) > \epsilon$. For all y, let $f_u(x) = 1$ if and only if $u_x \in I$. Let u, v such that $|I \cap u|, |I \cap v| > \epsilon/2$ fraction of the cloud size. By Markov's inequality, this is the case for roughly $\geq \epsilon/2$-fraction of $u \in V_{UG}$.

Let $f = f_u$ and $g = \pi \circ f_v$. The fact that there are no edges in $(I \cap u) \times (I \cap v)$ implies that for all x, $f(x)g(x) = 0$. Since $w_p(I) > \epsilon$, then there exists q close to p such that $w_q(I) > \epsilon$, and so there exist juntas $\hat{f}$ and $\hat{g}$ such that $\hat{f} \approx_q f$ and $\hat{g} \approx_q g$. Intuitively, these juntas cannot be supported only on coordinates disjoint to each other[1]. Thus if we let $\sigma(u)$ be a random color from the junta, then $\Pr(\sigma(u) = \pi(\sigma(v))) \geq \frac{1}{|\text{junta}|^2}$.

Hence, we obtain that a non-trivial fraction (roughly $\frac{\epsilon^2}{|\text{junta}|^2}$) of the UG constraints can be satisfied, which completes the soundness analysis.

References

[KR08] Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within 2-epsilon. *J. Comput. Syst. Sci.*, 74(3):335–349, 2008.

[1] This is not technically true, but there is an analog of this statement which is true and "morally" gives the same implication.