CMPUT 675: Approximation Algorithms **Fall 2011**

Lecture 23 (Dec. 6, 2011): Unique Games Conjecture and Consequences

Lecturer: Mohammad R. Salavatipour *Scribe: Xida Chen and Kyrylo Shegeda*

23.1 Unique Games Conjecture

In a Constraint Satisfaction Problem (CSP) we are given n variables $x_1, \ldots, x_n$ over a finite set of values U. We also have m constraints over these variables where each is essentially a function mapping some subsets of variables to 0 or 1. If all constraints are over k variables then each function has the form $f : U^k \rightarrow \{0, 1\}$ specifying which assignment of values to variables of the constraint satisfy the constraint and which don't. The goal is to find an assignment to variables to maximize the number of constraints that are satisfied. Many of the problems we have see so far (e.g. Max-SAT, Max-Cut, etc) are special forms of CSP's.

The unique-games is a special form of CSP defined in the following way. We have a constraint graph $G = (V, E)$, and $[k] = \{0, ..., k-1\}$ is the set of labels that vertices can take. For each edge $e \in E$, we have a permutation $\prod_e : [k] \rightarrow [k]$. If u is assigned label i and v is assigned label j, then we say that edge uv is satisfied if $\prod_{uv}(i) = j$. The goal is to assign labels (from $[k]$) to the vertices of the graph to maximize the number of satisfied edges.

A natural question comes up: can we satisfy all the edges? The answer is yes. And this can be done in polynomial time. However, it appears that if the answer is no then it's not easy to find a good approximate solution.

Theorem 1 ***Unique Games Conjecture(Khot'02)*** *For any ϵ, δ, there is a $k = k(\epsilon, \delta) > 0$ such that for any unique games with label size of k, it is NP-hard to distinguish the following two things:*

1. *at least $(1 - \epsilon)$-fraction of edges can be satisfied.*
2. *at most δ-fraction of edges can be satisfied.*

The unique games conjecture is still open, we are not sure whether it is true or not. Recent evidence shows that it might not be true. In the following, we will call the first one yes case and the second one no case.

23.2 Consequences

We now introduce some theorems which clearly show the consequence of unique games conjecture.

Theorem 2 (Khot and Regev'08,) *Assuming that unique game conjecture is true, then there is no $(2-\epsilon)$-approximation algorithm for vertex cover for any $\epsilon > 0$.*

Theorem 3 (Khot et al.'07) *Assuming that unique game conjecture is true, the Max-cut problem has no α-approximation algorithm for any constant $\alpha < \alpha_{GW}$.*

There are semi-definite programming algorithm for unique games with the following property: if it is a yes case, then the algorithm satisfies $1 - O(\sqrt{\epsilon \log k})$-fraction of edges.

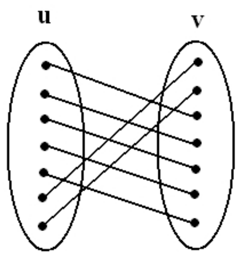

Figure 23.1: $MAX - 2LIN(k)$ permutation

Theorem 4 *Assuming that unique game conjecture is true, for any $\epsilon > 0$, there is a $k = k(\epsilon) > 0$ such that it is NP-hard to distinguish the following two instance:*

1. $(1 - \epsilon)$*-fraction are satisfiable.*
2. $(1 - \sqrt{\frac{2}{\pi}}\epsilon \log k + o(1))$*-fraction are satisfiable.*

We can see that the consequence of unique games conjecture being true is basically saying that semi-definite programming relaxation is the strongest relaxation that we can get. If unique games conjecture is not true, then the consequence is that we would be able to get better relaxation thus a lot of lower bound will not be true any more.

Now let's recall the problem of multi-cut. For multi-cut problem, we are given a graph $G = (V, E)$ and k pairs of source and destination nodes denoted as $\{s_i, t_i\}_{1 \le i \le k}$. Our goal is to remove a set of edges with minimum cost so that all the k pairs are separated. There are $O(\log k)$-approximation algorithm for multi-cut problem.

Theorem 5 *Assuming that unique game conjecture is true, there is no α-approximation algorithm for multi-cut problem for any constant α.*

Next we introduce another problem which is called $MAX - 2LIN(k)$, it is also called linear unique games. It is a kind of unique games with an additional constraint. Each permutation $\prod_{uv}$ is given by a constant $C_{uv} \in \{0, ..., k-1\}$. And $\prod_{uv}(x_u) = x_v$, $x_u - x_v = C_{uv}(mod)k$. The diagram in Figure 23.1 shows the above constraint.

In the rest we want to show that linear unique games conjecture is equivalent to unique games conjecture, which is: if $MAX - 2LIN(k)$ is true $\Leftrightarrow$ unique games conjecture is true.

Suppose we are given a graph $G = (V, E)$, and a set of labels $L = \{0, ..., k-1\}$. A permutation $\prod_{uv}$ is an instance of $MAX - 2LIN(k)$. We build an instance $G' = (V', E')$ for multi-cut as shown in Figure 23.2. We can see that the vertices $V' = V \times L$. The edges are defined as follows: $((u, i), (v, j)) \in E'$ if and only if $uv \in E$ and $i - j = C_{uv}(mod)k$. Also, $(u, i), (u, j)$ is a pair for each $u \in V$ and $i, j \in [k], i \neq j$. Also we assume that each edge has cost 1. From the construction of G' we see that $|V'| = k|V|$ and $|E'| = k|E|$.

To prove the theorem we use the following two lemmas.

Lemma 1 *For any $0 \le \epsilon \le 1$, give a feasible solution to MAX-2LIN(k) instance with value of $(1 - \epsilon)|E|$, there is a feasible solution to the multicut (in graph G') of cost at most $\epsilon |E'|$*

Proof. Suppose we have a good labeling X of G which satisfies $(1 - \epsilon)|E|$ of edges. Let $V'_0, ..., V'_{k-1}$ be a partitioning of vertices in G', where $V'_c = \{(u, x_u + c(\mod k))\}$ for all $u \in V$. Clearly, (u, i) and (u, j) are

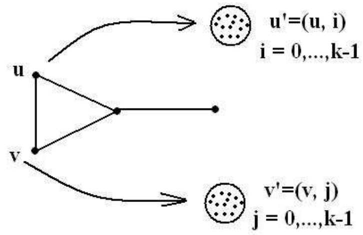

Figure 23.2: Construction of multi-cut instance $G' = (V', E')$

separated for $i \neq j$. Hence, we have a multicut for graph G'. Consider any edge $((u,i),(u,j)) \in E'$ such that it is in the cut.

Claim 1 *(u, v) must not be satisfied by X*

Proof. $i - j = c_{uv} (\mod k)$(by construction of E'). Suppose $(u,i) \in V'_c$ and $(v,j) \in V'_{c'}$. Then we have that $i = x_u + c(\mod k)$ and $j = x_v + c'(\mod k)$, which implies $c_{uv} = i - j(\mod k) = (x_u + c) - (x_v + c')(\mod k) = (x_u - x_v) + (c - c')(\mod k)$. Since $c \neq c'$ we have that $x_u - x_v \neq c_{uv}(\mod k)$ which means that (u, v) is not satisfied. ■

Thus the edge $(u, v) \in E$ is not satisfied in the solution to MAX2LIN(k). Since each edge (u, v) in the MAX2LIN(k) instance corresponds to k edges in the multicut, the total number of edges cut in G' is at most k times the number of unsatisfied edges of G which is $\epsilon |E|$. Thus, the size of the cut is at most $k\epsilon |E| = \epsilon |E'|$ ■

Lemma 2 *For any $0 \leq \epsilon \leq 1$, if G' has a multicut of size at most $\epsilon |E'|$ then there is a solution to MAX2LIN(k) instance of value at least $(1 - 2\epsilon)|E|$.*

Proof. Consider a multicut solution of G'. Suppose G' is cut into l parts $(l \geq k)$. Let $V'_1, ..., V'_l$ be a random ordering of this parts. For each vertex $u \in V$ there is a least index c such that $(u, i) \in V'_c$(for some i) and no other $(u, j) \in V'_{c'}$, for $c' < c$. Because the partition is given by multicut, we know that there can't be any other $(u, j) \in V'_c$ for $j \neq i$. We assign label i to u and we say that V'_c defines u. Consider an edge $e = uv \in E$, let ϵ_{uv} be the fraction of k edges in E'(corresponding to edge e) that are cut. Then for $(1 - \epsilon_{uv})$ fraction of these edges, both (u, i) and (v, j) are in the same part of the partition. Suppose that some part V'_c contains both endpoints (u, i) and $(v, j))$ of an edge $((u, i), (v, j))$. We call such part a good part. Let's also suppose that V'_c is defining u and v. Then we label u with i and v with j. Then the labeling of u and v satisfies $(u, v) \in E$ since then u is labeled with i, v is labeled with j, and the edge $((u, i), (v, j))$ implies that $i - j = c_{uv}(\mod k)$. There are $(1 - \epsilon_{uv})k$ good parts of the partition. We now want to bound the probability that the good part defines both u and v.

There are $\epsilon_{uv}k$ edges that are in multicut, hence we have at most $2\epsilon_{uv}k$ parts with one of the following properties holding (bad parts):

- it contains a vertex (u, i), but not (v, j)
- it contains a vertex (v, j), but not (u, i)
- it contains both (u, i) and (v, j) but doesn't define one of them

If any such part is ordered before a good part then the good part doesn't define the labels for u or v. Suppose there are $b \leq 2\epsilon_{uv}k$ bad parts. Thus the probability that edge $(u, v) \in E$ is not satisfied by the labeling is at most the probability that of the $b + (1 - \epsilon_{uv})k$ total good and bad parts, one of the bad ones is ordered first. This is at most $\frac{b}{b+(1-\epsilon_{uv})k} \leq \frac{2\epsilon_{uv}k}{2\epsilon_{uv}k+(1-\epsilon_{uv})k} = \frac{2\epsilon_{uv}}{1+\epsilon_{uv}} \leq 2\epsilon_{uv}$. Hence, the overall expected number of edges that are not satisfied by the random labeling is at most $2 \sum\limits_{uv \in E} \epsilon_{uv}$. By the definition of ϵ_{uv}, we have $k \sum\limits_{uv \in E} \epsilon_{uv}$ of E' in the multicut. The size of multicost is at most $k \sum\limits_{uv \in E} \epsilon_{uv} \leq \epsilon |E'| = k\epsilon |E|$. Then the expected number of edges not satisfied is at most $2\epsilon |E|$ and thus the expected number of satisfied edges is at least $(1 - 2\epsilon) |E|$. ■

Now let's assume that the above two lemmas are correct, we would like to prove that multi-cut problem has no α-approximation where α is a constant.

Proof. We prove it by contradiction. Assuming that there is an α-approximation algorithm for multi-cut problem. We choose ϵ and δ such that $\epsilon < \frac{1-\delta}{2\alpha}$. Given a graph G for $MAX - 2LIN(k)$ we build graph G' for multi-cut as described, and run α-approximation algorithm on G'. Then we will get the following two cases.

(1). If $(1 - \epsilon)|E|$-fraction of edges of G can be satisfied, then the solution of G' is $\leq \epsilon|E'|$, which means the solution can return a solution that is $\leq \alpha\epsilon|E'|$. According to lemma 2, we can get a solution for G of value $(1 - 2\alpha\epsilon)|E|$.

(2). If at most $\delta|E|$-fraction of edges of G can be satisfied, then we can satisfy no more than $\delta|E|$.

Since we choose $\epsilon < \frac{1-\delta}{2\alpha}$, then by simple manipulation we can get $(1 - 2\alpha\epsilon) > \delta$. It means that we can distinguish the above two case, which is a contradiction. Therefore, we can conclude that multi-cut problem has no α-approximation algorithm for any constant α. ■