18.408 Topics in Theoretical Computer Science Fall 2022
Lecture 16

Dor Minzer

Today, we prove it is NP-hard to approximate the set-cover problem within any constant factor.

1 The Set-cover Problem

An instance of the set cover problem consists of a universe $\mathcal{U}$ as well as a collection subsets $S_1, \ldots, S_m \subseteq \mathcal{U}$. The goal is to find the smallest number of subsets, that is $I \subseteq \{1, \ldots, m\}$ of smallest size, such that $\{S_i\}_{i \in I}$ cover all of the universe $\mathcal{U}$, namely $\bigcup_{i \in I} S_i = \mathcal{U}$. Set cover is a classical NP-hard problem, and today we will study it via the approximation len.

1.1 An Approximation Algorithm for Set-cover: the Greedy Algorithm

The greedy algorithm for set cover is probably the first idea that comes to mind when first facing the problem. Starting with $I = \emptyset$ and maintaining $A = \mathcal{U} \setminus \bigcup_{i \in I} S_i$, the idea is that at each step the algorithm picks the set S_i that covers as many of elements from A as possible, add i to I and continue. The following result states the performance of the greedy algorithm:

Theorem 1.1. *Let $(\mathcal{U}, \{S_i\}_{i \in I})$ be a set cover instance whose smallest cover has size k. Then the above greedy algorithm finds a set cover of size at most $k \ln(|\mathcal{U}|)$.*

Proof. Let $t \in \mathbb{N}$ be a parameter representing the step in the algorithm, let A_t be the set of uncovered elements in step t and let i_t be the index of the set we chose at that time. Consider a step t in the algorithm; since there is a set cover of $\mathcal{U}$ consisting of k sets, there are k sets that cover A_t, so at least one of them covers at least $1/k$ fraction of the elements from A_t. Since we picked S_{i_t} greedily, it follows that $|S_{i_t} \cap A_t| \geqslant \frac{|A_t|}{k}$, hence

$$|A_{t+1}| = |A_t \setminus (A_t \cap S_{i_t})| = |A_t| - |A_t \cap S_{i_t}| \leqslant \left(1 - \frac{1}{k}\right) |A_t| .$$

Thus, $|A_t| \leqslant \left(1 - \frac{1}{k}\right)^t |\mathcal{U}|$ and taking $t = k \ln(|\mathcal{U}|)$ we get that $A_t = \emptyset$, hence we end up with a cover. □

In words, denoting $n = |\mathcal{U}|$ to be the size of the universe of the set cover instance, we have shown that there is a $\ln n$ approximation algorithm for set cover. Can one do better than this?

1.2 (ℓ, m)-system sets: A Gadget for Set-cover

The rest of this lecture is devoted to establishing hardness of approximation results for set cover. To do that, we introduce a general framework in hardness of approximation, in which we first design a gadget — which is a mini instance of the problem we want to prove hardness for — that has a very good intended solution, but any other solution to it is significantly worse. We use this instance in a way that the "intended solutions" encode satisfying assignments to a label cover instance, and use it to do a reduction from label cover.

Definition 1.2. *An (ℓ, m, n) set system consists of a universe $\mathcal{U}$ of size n and a collection of m sets $A_1, \ldots, A_m$ and their complements $B_1 = \overline{A_1}, \ldots, B_m = \overline{A_m}$.*

We say such collection forms an (ℓ, m, n) instance if any collection of subsets $\{A_i\}_{i \in I}$ and $\{B_i\}_{i \in I'}$ that covers $\mathcal{U}$ must contain a set and its complement, that is $I \cap I' \neq \emptyset$.

In words, an (ℓ, m, n) set system is an instance of set cover that has a cover of size 2 (by taking a set and its complement), and any other cover (possibly much larger) must contain such cover. We have the following lemma proving the existence of (ℓ, m, n) systems

Lemma 1.3. *For all $\ell \in \mathbb{N}$, there is an $(\ell, 2\ell, 2^\ell)$ set system, and furthermore this system can be constructed in time $2^{O(\ell)}$.*

Proof. Take $\mathcal{U} = \{0, 1\}^\ell$, and define

$$A_i = \{ x \in \mathcal{U} \mid x_i = 0 \} , \qquad B_i = \{ x \in \mathcal{U} \mid x_i = 1 \} .$$

We leave it to the reader to verify this is an $(\ell, 2\ell, 2^\ell)$ set system. $\square$

2 A Reduction form Label-cover to Set-cover

We need the PCP theorem proved in previous lectures with an additional assumption of regularity. A bipartite graph $G = (L \cup R, E)$ is called bi-regular if all of the vertices in L have the same degree, and all of the vertices in R have the same degree.[1]

Theorem 2.1. *For all $\varepsilon > 0$, there is $k \in \mathbb{N}$ such that the problem gap-Label-Cover$[1, \varepsilon]$ is NP-hard on instances with alphabet size at most k and bi-regular constraint graphs.*

We use Theorem 2.1 to prove a strong hardness of approximation result for set cover. It will be easier for us to work with weighted version of the set-cover problem; in the problem set, you will see that hardness for weighted set-cover can be converted to hardness for standard instances of set cover.

A weighted set cover instance is composed of a universe $\mathcal{U}$, a collection of sets $S_1, \ldots, S_m \subseteq \mathcal{U}$ as well as a weight function $w\colon \{1, \ldots, m\} \to [0, \infty)$ indicating, for each set S_i, its weight. The problem is to find the minimum weight set cover of $\mathcal{U}$, that is find $I \subseteq \{1, \ldots, m\}$ such that $(S_i)_{i \in I}$ cover all of the universe $\mathcal{U}$, and $\sum\limits_{i \in I} w(i)$ is as small as possible. We note that the standard set-cover problem corresponds to the case that the weight function w is the constant 1 function.

For $a, b \in \mathbb{N}$, we denote by gap-Weighted-set-cover$[a, b]$ the problem in which one is given an instance of weighted set-cover that either has a cover of weight at most a, else all covers have weight at least b.

Theorem 2.2. *For all $\varepsilon > 0$, there is $\ell \in \mathbb{N}$ such that the problem gap-Weighted-set-cover$[\ell, \frac{\ell}{\varepsilon}]$ is NP-hard.*

The rest of this section is devoted to the proof of Theorem 2.2.

2.1 The Reduction from Label-cover to Set-cover

We show a reduction from gap-Label-cover$[1, \varepsilon]$ and use Theorem 2.1 to finish off the proof. Namely, we show a polynomial time reduction that maps an instance $\Psi = (G = (L \cup R, E), \Sigma_L, \Sigma_R, \{\Phi_e\}_{e \in E})$ of Label-cover to an instance $(\mathcal{U}, \{S_i\}_{i \in \mathcal{I}}, w)$ of weighted set cover such that:

[1] We remark that the additional bi-regularity condition in 2.1 can be quite easily ensured elementary transformations.

1. If Ψ is satisfiable, then $(\mathcal{U}, \{S_i\}_{i\in\mathcal{I}}, w)$ has a set cover of weight $a = 2\,|L|$.

2. If Ψ is at most ε-satisfiable, then $(\mathcal{U}, \{S_i\}_{i\in\mathcal{I}})$ has no set-cover of weight $b = \frac{1}{8\sqrt{\varepsilon}}\,|L|$.

Choose $\ell = |\Sigma_R|$, and take an $(\ell, 2\ell, 2^\ell)$ set system $(A_1, \ldots, A_\ell, B_1, \ldots, B_\ell)$ with universe U as in Lemma 1.3. Re-labeling the indices, we think of the sets $A_1, \ldots, A_\ell$ and $B_1, \ldots, B_\ell$ as being indexed by Σ_R (in other words, we identify $\{1, \ldots, \ell\}$ with Σ_R).

The universe of the set cover instance. The universe of the set-cover instance we construct is tuples of edges from Ψ and universe elements of the set system, namely $\mathcal{U} = E \times U$.

The sets in the set cover instance. Recall that Ψ is a projection label cover, meaning that for every each $e = (u, v) \in E$ we have a map $\phi_e \colon \Sigma_L \to \Sigma_R$ such that $\Phi_e = \{\,(\sigma, \phi_e(\sigma)) \mid \sigma \in \Sigma_L\}$. We define a set in our system, S_{u,σ_u} for each vertex $u \in L$ and $\sigma_u \in \Sigma_L$, as well as a set S_{v,σ_v} for each $v \in R$ and $\sigma_v \in \Sigma_R$. For $v \in R$ and $\sigma_v \in \Sigma_R$ we take

$$S_{v,\sigma_v} = \bigcup_{u:(u,v)\in E} \{e\} \times B_{\sigma_v}.$$

In words, we pick the B-set from our set system corresponding to σ_v, take several copies of it and attach to each one of them a name, which is the edge in the graph we associate it with.

For $u \in L$ and $\sigma_u \in \Sigma_L$, we define

$$S_{u,\sigma_u} = \bigcup_{v:(u,v)\in E} \{e\} \times A_{\phi_{u,v}(\sigma_u)}.$$

In words, for each u and label for it σ_u, we go over the neighbours v of u in G, and consider the A-set in our set system corresponding to $\phi_{u,v}(\sigma_u)$. We take a union over these, but also attach a name to each copy representing the edge it came from.

The weight function. Finally, we describe the weight function. If G was a regular graph (as opposed to bi-regular), we could have picked the weight function to be the constant 1 function, but this is not necessarily the case. Tracing back the construction of the label cover instance, we expect the size of L to be much larger than the size of R, hence there are many more sets corresponding to L than to R, and the weight function we define aims at balancing this out. Specifically, we define $w(S_{v,\sigma_v}) = \frac{|L|}{|R|}$ for each $v \in R$ and $\sigma_v \in \Sigma_R$ and $w(S_{u,\sigma_u}) = 1$ for $u \in L$ and $\sigma_u \in \Sigma_L$.

2.2 High Level Idea of the Analysis

Before proceeding to the formal analysis of the reduction, we explain the high level idea of it, and for that it is best to assume that $|L| = |R|$ so that the weight function can be ignored. Let us inspect an edge $e \in E$ in Ψ, and consider ways to cover the universe elements associated with it. For that, writing $e = (u, v)$, we can only pick sets generated either by u or by v. Further, note that in the definition of the S_u-sets we picked A's and in the definition of S_v-sets we picked B's, hence we may try to cover the universe element using the inteded cover in the set system. Inspecting, to do that we must pick S_{v,σ_v} and S_{u,σ_u} such that $\phi_{u,v}(\sigma_u) = \sigma_v$, namely pick up a pair of sets that were generated by a satisfying assignment of the edge e. Thus, the intended solution for our gadget set system can be utilized towards constructing a set cover (provided that we have a satisfying assignment of the edge).

However, by properties of our set system any pair collection of S_v and S_u sets that cover all universe elements from $\{e\} \times U$ must follows this strategy to an extent. Indeed, by the properties of the set system, if we are forbidden from picking a pair S_{v,σ_v} and S_{u,σ_u} corresponding to satisfying assignment, we would not be able to cover all of the elements of U, and hence not all of the elements in $\{e\} \times U$.

With this in mind, the punchline is that the satisfying-assignment based cover can be executed so long as we have a satisfying assignment for Ψ, which then takes care of the completeness of the reduction. As for the soundness of the reduction, since Ψ has no good single assignment we cannot pick one global assignment for Ψ that would allow us to cover all edges. In fact, any collection of sets that comes from a global assignment will completely cover the elements of $\{e\} \times U$ only for very few edges $e \in E$ (since the label cover has small soundness). Thus, typically for a vertex $u \in L$ and $v \in R$ we would need to pick many of the sets it generated to get a complete cover.

2.3 The Completeness of the Reduction.

Suppose Ψ is satisfiable and let $A_L \colon L \to \Sigma_L$ and $A_R \colon R \to \Sigma_R$ be assignments that satisfy Ψ. We choose the sets $|L| + |R|$ sets $\{S_{u,A_L(u)}\}_{u \in L}$ and $\{S_{v,A_R(v)}\}_{v \in R}$ and note that they form a set cover. Indeed, consider any element of the form $(e,x) \in \mathcal{U}$ and write $e = (u,v)$. Then we have picked the sets $S_{u,A_L(u)}$ which contains $\{e\} \times A_{\phi_{u,v}(A_L(u))} = \{e\} \times A_{A_R(v)}$ (where we used the fact that $\phi_{u,v}(A_L(u)) = A_R(v)$ since (u,v) is satisfied) and $S_{v,A_R(v)}$ which contains $\{e\} \times B_{A_R(v)}$, and since $A_{A_R(v)} \cup B_{A_R(v)} = U$, at least one of these sets contains (e,x).

By the definition of the weight function, it follows we have a set cover of weight $2\,|L|$.

2.4 The Soundness of the Reduction

Next, we show the soundness of the reduction. Towards this end, assume that Ψ is at most ε satisfiable and that our set cover instance has cover $\mathcal{C}$ of weight at most $\beta\,|L|$. For $u \in L$ and $v \in R$ define

$$\mathsf{Labels}(u) = \{\sigma_u \in \Sigma_R \mid S_{u,\sigma_u} \in \mathcal{C}\}, \qquad \mathsf{Labels}(v) = \{\sigma_v \in \Sigma_R \mid S_{v,\sigma_v} \in \mathcal{C}\}.$$

In words, for each vertex in the graph we define the set of labels that are associated with sets that are in our set cover $\mathcal{C}$. Then the total weight of the set cover instance is

$$\sum_{u \in L} |\mathsf{Labels}(u)| + \frac{|L|}{|R|} \sum_{v \in R} |\mathsf{Labels}(v)|,$$

and by assumption this is at most $\beta\,|L|$. Thus, we get that $\sum\limits_{u \in L} |\mathsf{Labels}(u)| \leqslant \beta\,|L|$ hence by an averaging argument for at least $3/4$ of $u \in L$ we have that $|\mathsf{Labels}(u)| \leqslant 4\beta$, and we refer to such vertices as good. Also, $\frac{|L|}{|R|} \sum\limits_{v \in R} |\mathsf{Labels}(v)| \leqslant 4\beta\,|L|$ hence $\frac{1}{|R|} \sum\limits_{v \in R} |\mathsf{Labels}(v)| \leqslant 4\beta$ so by an averaging argument for at least $3/4$ of $v \in R$ we have that $|\mathsf{Labels}(v)| \leqslant 4\beta$; we also refer to such vertices as good.

Note that sampling $e \in E$ and writing $e = (u,v)$, by the bi-regularity of G, the vertex u is distributed uniformly in L and hence is good expect with probability $1/4$, and v is distributed uniformly in R and hence is good expect with probability $1/4$. Thus, both endpoints of e are good with probability at least $1/2$, and we denote the set of these edges by $E' \subseteq E$. We will show an assignment that satisfies many of these edges.

The following claim says that if for every edge $e = (u,v) \in E$, the label sets $\mathsf{Labels}(u)$ and $\mathsf{Labels}(v)$ contain a pair of assignments that satisfy the constraint on e.

Claim 2.3. *Let $e \in E$ be any edge, and write $e = (u, v)$. Then there are pairs of labels $\sigma_u \in \mathsf{Labels}(u)$ and $\sigma_v \in \mathsf{Labels}(v)$ that satisfy Φ_e, namely such that $(\sigma_u, \sigma_v) \in \Phi_e$.*

Proof. Otherwise, looking at the universe elements of the form (e, x), only the sets $S_{v,\sigma}$ and $S_{u,\sigma'}$ may cover them, and if there is no pair such as in the claim, then all of these sets from $\mathcal{C}$ give us elements of the form $\{e\} \times A_i$ for $i \in I$ and $\{e\} \times B_j$ for $j \in J$ where $I \cap J = \emptyset$. By the properties of our set system, $(A_i)_{i \in I}, (B_j)_{j \in J}$ do not form a cover of U, hence there is some $x \in U$ not covered by them, and then $(e, x) \in \mathcal{U}$ is not be covered. $\square$

The list decoding assignment. We are now ready to describe a good assignment for Ψ by an idea called list decoding. The idea that for edges $e = (u, v) \in E'$, the list of labels of u and v are short, and by Claim 2.3 contain a pair of satisfying assignment. Thus, if we pick a label for each vertex uniformly from its list, the probability that an edge $e = (u, v) \in E'$ is satisfied is at least 1 over the product of the sizes of the lists of u and v, which is significant (as these lists are short).

More precisely, define A_L and A_R in a randomized manner;: for each $u \in L$ independently, pick $A_L(u) \in \mathsf{Labels}(u)$ uniformly, and for each $v \in R$ pick $A_R(v) \in \mathsf{Labels}(v)$ uniformly. We analyze the expected fraction of edges that A_L and A_R satisfy. Fix $e \in E'$ and write $e = (u, v)$. Then by Claim 2.3 there is a pair of labels $\sigma_u^\star \in \mathsf{Labels}(u)$ and $\sigma_v^\star \in \mathsf{Labels}(v)$ that satisfy the constraint between u and v, and we note that $\Pr\left[A_L(u) = \sigma_u^\star\right] = \frac{1}{|\mathsf{Labels}(u)|} \geqslant \frac{1}{4\beta}$ since u is good. Similarly, $\Pr\left[A_R(v) = \sigma_v^\star\right] \geqslant \frac{1}{4\beta}$, and since these events are independent it follows that

$$\Pr\left[A_L, A_R \text{ satisfy } e\right] \geqslant \Pr\left[A_L(u) = \sigma_u^\star\right]\Pr\left[A_R(v) = \sigma_v^\star\right] \geqslant \frac{1}{16\beta^2}$$

Hence, by linearity of expectation, the expected number of constraints that A_L and A_R satisfy is at least $\frac{|E'|}{16\beta^2} = \frac{|E|}{32\beta^2}$, in particular it follows that there is an assignment to Ψ satisfying at least $1/32\beta^2$ fraction of constraints. As $\mathsf{val}(\Psi) \leqslant \varepsilon$, it follows that $\beta \geqslant \frac{1}{8\sqrt{\varepsilon}}$. This finishes the soundness analysis.

2.5 Reflecting on Theorem 2.2

One immediate corollary of Theorem 2.2 is that it is NP-hard to approximate the set-cover problem within any constant factor. In fact, it turns out that one can prove super-constant hardness of approximation results for set cover using this approach, and even get hardness of approximation result of up to factor $\Theta(\log n)$. To do that, one needs to make sure that ℓ is a not-too-large-growing function of the instance size (which is the alphabet size of Ψ), and that the soundness of the label-cover instance is vanishing with the instance size at sufficient rate (say $\varepsilon = 1/\log n$). This is an example where PCPs with sub-constant soundness come in handy in hardness of approximation results, but we will not elaborate on this further.

We remark that by now it is known that it is NP-hard to approximate Set-cover within factor $(1 - o(1)) \ln n$, which is optimal by the greedy algorithm above.