CMPUT 675: Approximation Algorithms **Fall 2014**

Lecture 31 (Nov 28): Set Cover Hardness

Lecturer: Zachary Friggstad *Scribe: Zachary Friggstad*

31.1 Set Cover Hardness

In this lecture, we prove the following.

Theorem 1 *There is some constant $c > 0$ such that there is no $c \cdot \ln(|X|)$-approximation for* Set Cover *unless* $\mathbf{NP} \subseteq \mathbf{ZPTIME}(n^{O(\log \log n)})$ *where X is the set of items to be covered.*

That is, if such an approximation existed then we could solve every problem in **NP** with a randomized algorithm that always gives a correct answer and has expected running time $n^{O(\log \log n)}$. Assuming $\mathbf{NP} \not\subseteq \mathbf{ZPTIME}(n^{O(\log \log n)})$ is a stronger assumption than $\mathbf{P} \neq \mathbf{NP}$, but it is still open and it seems quite plausible. At the very least, it shows that getting a $o(\log n)$-approximation requires solving a major open problem in complexity theory in a way that many researchers believe is not possible.

In fact, an even stronger statement holds under the more common $\mathbf{P} \neq \mathbf{NP}$ assumption: there is no $c \cdot \ln(n)$-approximation for Set Cover for *any* constant $c < 1$ (see the discussion at the end of these notes).

At any rate, the hardness we will show is asymptotically tight since we already know of an $H_n = \ln(n) + O(1)$ approximation for Set Cover.

31.1.1 Label Cover

Our starting point is from the Label Cover problem.

Definition 1 *In an instance of the* Label Cover *problem, we are given a bipartite graph $G = (V; E)$ where V_L, V_R are the two sides of G. Additionally, we are given a finite set of* label sets Σ *and, for each edge $e = (u, v) \in E$ with $u \in V_L, b \in V_R$, a mapping $\pi_e : \Sigma \to \Sigma$. For a given* labelling $\sigma : V \to \Sigma$ *of labels to nodes, say edge e is satisfied if $\pi_{u,v}(\sigma(u)) = \sigma(v)$. The goal is to find a labelling that maximize the number of satisfied edges.*

The following hardness is known for Label Cover. The statement is quite precise, but the specific parameters are important in many reduction from Label Cover.

Theorem 2 *For any language $L \in \mathbf{NP}$ and any integer $\ell \geq 1$, given any instance x of the decision problem "$x \in L$?" we can construct a* Label Cover *instance, say $G = (V; E)$ with label set Σ and mappings $\{\pi_{u,v}\}_{(u,v)\in E}$ such that the following hold:*

- Completeness*: If $x \in L$, then there is some labelling σ that satisfies all $|E|$ edges.*
- Soundness*: If $x \notin L$, then no labelling satisfies more than $2^{-\ell} \cdot |E|$ edges.*

- Size of the Parameters*: $|\Sigma| = c^\ell$ for some constant c and $|V_L| = |V_R| = |x|^{O(\ell)}$.*
- Regularity*: All vertices of G have the same degree $d^{O(\ell)}$ for some constant d.*

Furthermore, the running time of the reduction is $|x|^{O(\ell)}$.

The Williamson and Shmoys text discusses how to obtain this hardness from the PCP theorem with only one (highly nontrivial and very interesting) detail left out. We will skip it for the sake of time. It is a very interesting read and I strongly encourage you to take a look at it.

Corollary 1 *There is no constant-factor approximation for* LABEL COVER *unless* $\mathbf{P} = \mathbf{NP}$.

Proof. Invoke the reduction from Theorem 2 with $\ell = \log_c \epsilon$. This produces a hardness gap of 1 vs. $c^\ell = \epsilon$. Since ϵ is a constant, then ℓ is as well so the running time of the reduction is polynomial. ■

However, we can get a much stronger hardness result under slightly stronger assumptions by choosing ℓ to be super-constant.

Corollary 2 *For any constant $1 > \epsilon > 0$, there is no $1/2^{\log^{1-\epsilon}(N)}$-approximation for* LABEL COVER *unless* $\mathbf{NP} \subseteq \mathbf{DTIME}(n^{O(\log^{1/\epsilon}(n))})$*. Here, N is the number of nodes in the label cover instance.*

The hardness ratio may seem strange, but for any constants $1 > \epsilon > 0, d > 0$ we have that $\log^d(n)$ grows slower than $2^{\log^{1-\epsilon}(n)}$ so this implies there is no polylogarithmic approximation for LABEL COVER.

Proof. Invoke Theorem 2 with $\ell = \alpha \cdot \log_2^{1/\epsilon} |x|$ for some constant α. The running time of this reduction is $|x|^{O(\ell)}$.

If $x \notin L$, then at most $2^{-\ell} \cdot |E|$ edges can be satisfied by any assignment. For an appropriate choice of constant α and recalling $N \leq |x|^{O(\ell)}$, one can verify that $2^{\log_2^{1-\epsilon}(N)} \leq 2^\ell$. ■

31.2 From Label Cover to Set Cover

31.2.1 A Helpful Gadget

Consider a *set system* $(U; C_1, \overline{C}_1, \ldots, C_m, \overline{C}_m)$ where U_m is a finite set and for each i we have $C_i \subseteq U$ and $\overline{C}_i = U - C_i$. We are particularly interested in such a set system when the only covers of U that use few of the listed subsets $C_1, \overline{C}_1, C_2, \overline{C}_2, \ldots, C_m\overline{C}_m$ is by choosing some complementary pair $C_i, \overline{C}_i$.

Lemma 1 *For positive integers m, k, there is set system $(U; C_1, \overline{C}_1, \ldots, C_m, \overline{C}_m)$ with $|U| = \text{poly}(m, 2^k)$ such that for any $\mathcal{S} \subseteq \{C_1, \overline{C}_1, \ldots, C_m, \overline{C}_m\}$ such that $\mathcal{S}$ covers U and $|\mathcal{S}| \leq k$ we must have $C_i, \overline{C}_i \in \mathcal{S}$ for some $1 \leq i \leq m$.*

Furthermore, such a set system can be constructed in $\text{poly}(m, 2^k)$ time.

While proving this lemma with a deterministic construction is a bit tricky, there is a nice and simple randomized construction of this set.

Lemma 2 *There is a randomized algorithm with running time* $\text{poly}(m, 2^\ell)$ *that constructs such a set system with probability at least* $1/2$.

Proof. Let U be a set of size $\ln(2m) \cdot 2k \cdot 2^k$. For each $1 \leq i \leq m$ form each C_i by adding each $x \in U$ to C_i independently with probability $1/2$.

Let $\mathcal{S}$ consist of k subsets among $\{C_1, \overline{C}_1, \ldots, C_m, \overline{C}_m\}$ such that $|\mathcal{S}' \cap \{C_i, \overline{C}_i\}| \leq 1$ for each $1 \leq i \leq m$. Because the sets in $\mathcal{S}$ were formed independently (as $\mathcal{S}$ does not contain a complementary pair), then for each $x \in U$ we have $\Pr[x \text{ is not covered by } \mathcal{S}] = 2^{-k}$. So, the probability that $\mathcal{S}$ covers U is exactly $(1 - 2^{-k})^{|U|}$.

The number of different size-k subsets of $\mathcal{S}$ is at most $(2m)^k$. By the union bound, the probability that U is covered by some collection $\mathcal{S}$ of size k that includes no complementary pair is at most

$$\begin{aligned}
(2m)^k \cdot (1 - 2^{-k})^{|U|} &= (2m)^k \cdot (1 - 2^{-k})^{\ln(2m) \cdot 2k \cdot 2^k} \\
&\leq (2m)^k \cdot e^{-\ln(2m) \cdot 2k} \\
&= (2m)^k \cdot \frac{1}{(2m)^{2k}} \\
&\leq 1/2
\end{aligned}$$

■

31.2.2 The Construction

We prove Theorem 1 by first going through LABEL COVER. Specifically, let $L \in \mathbf{NP}$ and let x be an instance of the decision problem $x \in L?$.

For the remainder of this proof, we consider the following parameters:

- $n := |x|$
- $\ell = \Theta(\log \log n)$
- $k = \Theta(\ell \cdot k)$

To reduce the amount of notation, the leading constants in the $\Theta(\cdot)$ terms above will not be explicitly described and the arguments presented will assume they are chosen appropriately.

Invoke Theorem 2 with the given parameter $\ell = \Theta(\log \log n)$ to get a LABEL COVER instance on graph $G = (V; E)$ with sides V_L, V_R, label set Σ, and constraints $\pi_e, e \in E$.

Also invoke Lemma 2 to find a set system $(U; C_1, \overline{C}_1, \ldots, C_{|\Sigma|}, \overline{C}_{|\Sigma|})$ such that any collection of k sets among $C_1, \overline{C}_1, \ldots, C_{|\Sigma|}, \overline{C}_{|\Sigma|}$ that cover U must include a complementary pair. Note that $|U| = \text{poly}(|\sigma|, 2^\ell)$ which, for the given parameters is bounded by $n^{O(\log \log n)}$. We assume, from now on, that this construction was successful (this is discussed more at the end of the analysis).

For each edge e of the LABEL COVER instance G, we let $(U^e; C_1^e, \overline{C}_1^e, \ldots, C_{|\Sigma|}^e, \overline{C}_{|\Sigma|}^e)$ be a new copy of $(U; C_1, \overline{C}_1, \ldots, C_{|\Sigma|}, \overline{C}_{|\Sigma|})$. In particular, the sets U^e are disjoint for various e.

Finally, for each $u \in V_L$ and each $a \in \Sigma$ we construct a cover-set $S_{u,a} = \cup_{e \in \delta(u)} C^e_{\pi_e(a)}$ and for each $v \in V_R$ and each $b \in \Sigma$ we construct a cover-set $S_{v,b} = \cup_{e \in \delta(v)} \overline{C}^e_b$. The idea here is that a complementary pair in the set system for edge $e = (u, v)$ naturally corresponds to labels a, b that satisfy constraint π_e: i.e. if $\pi_e(a) = b$ then $S_{u,a}$ includes $C^e_{\pi(a)} = C^e_b$ and $S_{v,b}$ includes $\overline{C}^e_b$ so they collectively cover U^e.

The final Set Cover instance $(X, \mathcal{S})$ has $X = \cup_{e \in E} U^e$ and

$$\mathcal{S} = \{S_{u,a} : u \in V_L, a \in \Sigma\} \cup \{S_{v,b} : v \in V_R, b \in \Sigma\}.$$

Note that $|X| = |U| \cdot |E| = n^{O(\log \log n)}$ and that $|\mathcal{S}| = |V| \cdot |\Sigma| = n^{O(\log \log n)}$ and that the entire reduction takes $n^{O(\log \log n)}$ time.

31.2.3 Completeness

Claim 1 *If $x \in L$ then there is* SET COVER *solution using $|V|$ sets.*

Proof. Let $\sigma : V \to \Sigma$ be a labelling of G that satisfies all π_e constraints. Our SET COVER solution $\mathcal{C}$ is simply $\{S_{w,\sigma(w)} : w \in V\}$. Note $|\mathcal{C}| = |V|$.

We show that $\mathcal{C}$ indeed covers U. Consider any edge $e = (u, v) \in E$, we show $\mathcal{C}$ covers U^e. Note that $\mathcal{C}$ includes sets $S_{u,\sigma(u)}$ and $S_{v,\sigma(v)}$ and that $\pi_e(\sigma(u)) = \sigma(v)$. Therefore, $\mathcal{C}$ covers the complementary pair $C^e_{\pi_e(\sigma(u))} = C^e_{\sigma(v)} \subseteq S_{u,\sigma(u)}$ and $\overline{C}^e_{\sigma(v)} \subseteq S_{v,\sigma(v)}$ so it covers U^e. ■

31.2.4 Soundness

Claim 2 *If $x \notin L$ then any* SET COVER *solution requires at least $k \cdot |V|/8$ sets.*

Proof. Let $\mathcal{C}$ be any subset of $\mathcal{S}$ that covers $X = \cup_{e \in E} U^e$.

Definition 2 *Say a vertex $w \in V$ is* good $|\{s \in \Sigma : S_{w,s} \in \mathcal{C}\}| \leq k/2$*, otherwise say w is* bad*. Say an edge $e = (u, v) \in E$ is* good *if both u and v are good, otherwise say e is* bad*.*

A bad vertex contributes many sets to $\mathcal{C}$. Our strategy is to show that there are many bad edges, thus there are many bad vertices.

- **There are many bad edges**

  We randomly construct a labelling $\sigma : V \to \Sigma$ such that a good edge will have its corresponding constraint satisfied with reasonably high probability. However, the soundness in the construction of G ensures that few constraints can be satisfied. Thus, there are few good edges.

  More precisely, construct σ randomly by setting $\sigma(w)$ to be a label chosen uniformly at random from $\{s : S_{w,s} \in \mathcal{C}\}$. If this set is empty, then set $\sigma(w)$ to be an arbitrary label. Do this independently for each $w \in V$.

  Let $e = (u, v)$ be a good edge. Because $\mathcal{C}$ covers U^e and because at most k sets of the form $S_{u,a}$ or $S_{v,b}$ are in $\mathcal{C}$ then they must contain some complementary pair. That is, there are labels $a, b \in \Sigma$ such $S_{u,a}, S_{v,b} \in \mathcal{C}$ and for the sets $C^e_{\pi(a)} \subseteq S_{u,a}$ and $\overline{C}^e_b \subseteq S_{v,b}$ we have $\pi(a) = b$. We then see

  $$\Pr[\pi_e \text{ is satisfied by } \sigma] \geq \Pr[\sigma(u) = a \text{ and } \sigma(v) = b] \geq 4/k^2.$$

  On the other hand, by the soundness in Theorem 2

  $$(\#\text{ of good edges}) \cdot \frac{4}{k^2} \leq \mathrm{E}[\#\text{ of constraints satisfied by } \sigma] \leq 2^{-\ell} \cdot |E|.$$

By our choice of k and ℓ we have $k^2/4 \cdot 2^{-\ell} \leq 1/2$ so at most half of the edges are good. Thus, there are at least $|E|/2$ bad edges.

- **So there are many bad vertices**

  We know that G is a regular graph (c.f. Theorem 2), so say $|\delta(w)| = D$ for each $w \in V$ (which also means $D \cdot |V| = 2 \cdot |E|$).

  Let b_E be the number of bad edges and b_V be the number of bad vertices. Finally, for each edge $e \in E$ let $b(e)$ be the number of bad endpoints of E. We have:

  $$|E|/2 \leq b_E \leq \sum_{e \in E} b(e) = D \cdot b_V$$

  so $b_V \geq |E|/(2D) = |V|/4$.

- **Therefore $\mathcal{C}$ is big**

  For each bad vertex w, we count at least $k/2$ sets in $\mathcal{C}$ of the form $S_{w,s}$. Therefore,

  $$|\mathcal{C}| \geq k/2 \cdot b_v \geq k \cdot |V|/8.$$

■

31.2.5 Wrapping Up

We saw in the completeness case that there is a solution using only $|V|$ sets and in the soundness case that no solution uses fewer than $k \cdot |V|/8$ sets. Therefore, we cannot approximate the problem better than a factor of $k/8$ (under the complexity theory assumption described below). We want to state this in terms of the new Set Cover instance size. We have $|X| = n^{O(\log \log n)}$ and $k/8 = \Theta(\log n \cdot \log \log n)$, so $k/8 = O(\log |X|)$. This is what we wanted to show.

The running time of the reduction is $n^{O(\log \log n)}$ which is not polynomial. Furthermore, the reduction is only guaranteed to have the above soundness properties with probability at least $1/2$ (the completeness always holds if the random construction from Lemma 2 does not have the desired properties).

So, if we can approximate Set Cover within a ratio better than $k/8$, then we can randomly decide every $L \in \mathbf{NP}$ with an algorithm that runs in $n^{O(\log \log n)}$ time, always accepts a *yes* instance, and rejects every *no* instance with probability at least $1/2$. This is done simply by applying this randomized reduction from L to Set Cover and then using the Set Cover approximation to decide between *yes* and *no* instances by seeing if it found fewer than $k \cdot |V|/8$ sets or not.

This shows that there is some constant $c > 0$ such that there is no $(c \cdot \ln |X|)$-approximation for Set Cover unless $\mathbf{NP} \subseteq \mathbf{co\text{-}RTIME}(n^{O(\log \log n)})$. A nice exercise in complexity theory is to show that $\mathbf{NP} \subseteq \mathbf{co\text{-}RP}$ implies $\mathbf{NP} \subseteq \mathbf{ZPP}$ and the same arguments work for these $n^{O(\log \log n)}$-time analogs. So, in fact we have just shown that no $(c \cdot \ln |X|)$-approximation exists unless $\mathbf{NP} \subseteq \mathbf{ZPTIME}(n^{O(\log \log n)})$.

Finally, if we use the deterministic construction stated in Lemma 1 then the running time of the reduction from L to Set Cover is deterministic so it would establish that there is no $(c \cdot \ln |X|)$-approximation unless $\mathbf{NP} \subseteq \mathbf{DTIME}(n^{O(\log \log n)})$.

31.3 Discussion

Lund and Yannakakis [LY94] provide the first logarithmic hardness for Set Cover, proving there is no $(\log_2 n)/4$-approximation unless $\mathbf{NP} \subseteq \mathbf{DTIME}(n^{O(\text{polylog}n)})$ where polylog$n$ means $\log^d n$ for some constant d. They also show a slightly tighter bound of $(\log_2 n)/2$ assuming $\mathbf{NP} \not\subseteq \mathbf{ZPTIME}(n^{O(\text{polylog}n)})$.

Feige sharpened this result, proving that for any constant $c < 1$ that there is no $(c \cdot \ln n)$-approximation for Set Cover unless $\mathbf{NP} \subseteq \mathbf{DTIME}(n^{O(\log \log n)})$ [F98]. This is tight even up to the leading constant because the greedy algorithm is a $\ln n + O(1)$ approximation. Feige's work, like the reduction in this lecture, critically relies on the hardness of Label Cover that was ultimately proven by Raz [R98]. Dinur and Steurer further build on this work by providing a better bound on the hardness of Label Cover in a critical case which, ultimately, leads to the same $(c \cdot \ln n)$-hardness for any constant $c < 1$ under the standard assumption $\mathbf{P} \neq \mathbf{NP}$ [DS14].

References

DS14 I. Dinur and D. Steurer, An analytic approach to parallel repetition, In Proceedings of STOC, 2014.

F98 U. Feige, A Threhold of $\ln n$ for approximating set cover, Journal of the ACM, 45(4):634–652, 1998.

LY94 C. Lund and M. Yannakakis, On the hardness of approximating minimization problems, Journal of the ACM, 41(5):960–981, 1994.

R98 R. Raz, A parallel repetition theorem, SIAM Journal on Computing, 27(3):763–803, 1998.