CSC 2421: Approximation Algorithms **Winter 2003**

Lecture 11: March 25th

Lecturer: Charles Rackoff *Scribes: Travis Gagie and Dan Zhao*

Today's lecture will cover:

1. improvements on the PCP Theorem;
2. an improved lower bound for MAX3SAT;
3. constant lower bounds on the approximation ratio for CLIQUE, INDEPENDENTSET, and VERTEXCOVER;
4. better constant lower bounds for CLIQUE; and
5. the beginnings of a nearly linear lower bound for CLIQUE.

Further material can be found in the text.

1 Improvements on the PCP Theorem

In the previous lecture, we stated the PCP Theorem:

$$\mathrm{NP} = \mathrm{PCP}_{1,\frac{1}{2}}\left(O(\log n), O(1)\right).$$

Using this, we showed that, for all $L \in \mathrm{NP}$, there exists a polytime-computable gap-introducing reduction from L to MAX3SAT that introduces a "gap of constant size". We now combine these results to prove another version of the PCP Theorem.

Theorem 1. *For some constant $s < 1$,*

$$\mathrm{NP} = \mathrm{PCP}_{1,s}\left(O(\log n), 3\right).$$

Proof. We first show that, for some $s < 1$, $\text{NP} \subseteq \text{PCP}_{1,s}(O(\log n), 3)$.

Let L be a language in NP. Given an instance y of L, we can construct in polytime a multiset of 3-clauses $\mathcal{F}$ such that for some $\epsilon > 0$:

- if $y \in L$, then all of the 3-clauses in $\mathcal{F}$ are simultaneously satisfiable; and,
- if $y \notin L$, then fewer than a $1-\epsilon$ fraction of the 3-clauses in $\mathcal{F}$ are simultaneously satisfiable.

Let a proof π for y be a truth assignment to the variables in $\mathcal{F}$.

Let V be a probabilistic polytime verifier that, given y and a proof π:

1. computes $\mathcal{F}$ as described above; and
2. accepts if and only if π satisfies a 3-clause chosen from $\mathcal{F}$ uniformly at random.

The second step, choosing a 3-clause, takes $O(\log n)$ random bits, where n is the length of y. Notice that V only needs to examine the (at most) 3 bits of π that assign truth values to the variables in the chosen 3-clause.

Suppose that $y \in L$. Then all of the 3-clauses in $\mathcal{F}$ are simultaneously satisfiable. Thus, there exists a proof π such that π satisfies all of the 3-clauses in $\mathcal{F}$. Therefore, $V(y, \pi)$ accepts with probability 1.

Now suppose that $y \notin L$. Then, for all proofs π, π satisfies fewer than a $1-\epsilon$ fraction of the 3-clauses in $\mathcal{F}$. Therefore, $V(y, \pi)$ accepts with probability less than $1-\epsilon$.

Thus, via V, $L \in \text{PCP}_{1,1-\epsilon}(O(\log n), 3)$.

Since $\text{NP} \supseteq \text{PCP}_{c,s}\left(O(\log n), n^{O(1)}\right)$, we have $\text{NP} = \text{PCP}_{1,1-\epsilon}(O(\log n), 3)$. □

Theorem 1 has the following corollary. The proof has the same form as that of Corollary 5 in the last set of notes, and depends on repeated simulation.

Corollary 2. *For some constant b,*

$$\text{NP} = \text{PCP}_{1,\frac{1}{2}}(O(\log n), b).$$

In 1998, Guruswami, Lewin, Sudan, and Trevisan [GLST98, Gur99] proved a theorem that (nearly) combines the advantages of Theorems 1 and 2.

Theorem 3 (Guruswami, Lewin, Sudan, and Trevisan, 1998). *For all $\epsilon > 0$,*

$$\mathrm{NP} = \mathrm{PCP}_{1,\frac{1}{2}+\epsilon}(O(\log n), 3)\,.$$

Here, it is important that the verifier can choose 3 bit positions in the proof *adaptively*. That is, the position of the second bit examined can depend on the truth value of the first, and the position of the third bit examined can depend on the truth values of the first and second.

Earlier, Håstad [Hås97, Hås01] proved a similar but incomparable result, which we will use in Section 2 to obtain a stronger hardness result for MAX3SAT.

Theorem 4 (Håstad, 1997). *For all $\epsilon > 0$,*

$$\mathrm{NP} = \mathrm{PCP}_{1-\epsilon,\frac{1}{2}+\epsilon}(O(\log n), 3)\,.$$

Furthermore, it is sufficient to use a verifier that chooses 3 bit positions in the proof *a priori*. That is, the verifier:

1. examines the instance and the random bits;
2. chooses 3 bits positions in the proof and a truth value t;
3. examines the 3 chosen bits of the proof; and
4. accepts if and only if the exclusive-or of the 3 bits is t.

2 An Improved Lower Bound for MAX3SAT

In the previous lecture, we used the PCP Theorem to obtain a hardness result for MAX3SAT approximation. We will use Theorem 4 in a similar way to obtain a stronger hardness result.

Theorem 5. *If* $\mathrm{P} \neq \mathrm{NP}$*, then no polytime algorithm approximates* MAX3SAT *to within a factor strictly less than* $\frac{8}{7}$.

Proof. Let L be an NP-complete language, and let $\epsilon > 0$. By Theorem 4, for some constant d and some verifier V, we have $L \in \mathrm{PCP}_{1-\epsilon,\frac{1}{2}+\epsilon}(d \log n, 3)$ via V. Moreover, V chooses the 3 bits it examines a priori, as described at the end of Section 1.

Suppose that we want to decide membership in L of an instance y, with $|y| = n$. Furthermore, suppose that proofs for V on y are of length at most u, where u is polynomial in n.

Let $r_1, \ldots, r_{n^d}$ be the n^d binary strings of length $d \log n$; these correspond to the possible choices that V might make for its random bits. First, we construct n^d 3-CNF formulas. These formulas, together, will contain variables $x_1, \ldots, x_u$ that correspond to the bits of a proof for V on y. Then, we consider the multiset of 3-clauses occurring in these formulas.

Let π be a proof for y. For $1 \leq i \leq n^d$, suppose that $V(y, \pi)$ chooses random string r_i. Let j, k, and l be the bit positions V chooses to examine and let t be the truth value V chooses.

If $t = 0$, then let F_i be the 3-CNF formula

$$(\neg x_j \vee \neg x_k \vee \neg x_l) \wedge (\neg x_j \vee x_k \vee x_l) \wedge (x_j \vee \neg x_k \vee x_l) \wedge (x_j \vee x_k \vee \neg x_l).$$

Notice that F_i is logically equivalent to $\neg(x_j \oplus x_k \oplus x_l)$, where $\oplus$ denotes the exclusive-or operator. Therefore, $V(y, \pi)$ with random bits r_i accepts if and only if π satisfies F_i.

If $t = 1$, then let F_i be the 3-CNF formula

$$(x_j \vee x_k \vee x_l) \wedge (x_j \vee \neg x_k \vee \neg x_l) \wedge (\neg x_j \vee x_k \vee \neg x_l) \wedge (\neg x_j \vee \neg x_k \vee x_l).$$

Notice that F_i is logically equivalent to $x_j \oplus x_k \oplus x_l$. Therefore, $V(y, \pi)$ with random bits r_i accepts if and only if π satisfies F_i.

Let $\mathcal{F}$ be the multiset of 3-clauses that occur in $F_1, \ldots, F_{n^d}$. Notice that $|\mathcal{F}| = 4n^d$.

Suppose that $y \in L$. Then there exists a proof π such that, for at least a $1 - \epsilon$ fraction of the choices of random bits, $V(y, \pi)$ accepts. This means that π is a truth assignment to $x_1, \ldots, x_u$ that satisfies at least a $1 - \epsilon$ fraction of $F_1, \ldots, F_{n^d}$. Since all of the F_i's contain the same number of 3-clauses, it follows that π satisfies at least a $1 - \epsilon$ fraction of the 3-clauses in $\mathcal{F}$.

Suppose that $y \notin L$. Then, for all proofs π, $V(y, \pi)$ accepts for fewer than a $\frac{1}{2} + \epsilon$ fraction of the choices of random bits. This means that π is a truth assignment to $x_1, \ldots, x_u$ that satisfies fewer than a $\frac{1}{2} + \epsilon$ fraction of $F_1, \ldots, F_{n^d}$. Since all of the F_i's contain exactly 4 clauses, it follows that π satisfies fewer than a

$$\frac{4\left(\frac{1}{2}+\epsilon\right)+3\left(\frac{1}{2}-\epsilon\right)}{4}=\frac{7+2\epsilon}{8}$$

fraction of the 3-clauses in $\mathcal{F}$.

Therefore, for all $\epsilon > 0$, if there exists a polytime algorithm that approximates Max3Sat to within a factor of $\frac{8(1-\epsilon)}{7+2\epsilon}$, then we can decide membership for an NP-complete language in polytime, implying P = NP. Since

$$\lim_{\epsilon\to 0}\frac{8(1-\epsilon)}{7+2\epsilon}=\frac{8}{7},$$

this is equivalent to the claim. □

3 Constant Lower Bounds for Clique, IndependentSet, and VertexCover

In the first lecture, we gave a simple 2-approximation algorithm for finding a minimum unweighted vertex cover. This algorithm finds a maximal matching M and includes in its vertex cover all vertices incident to edges in M. We listed slightly better results which also hold in the more general weighted case. For example, algorithms by Bar-Yehuda and Even [BYE85] and Monien and Speckenmeyer [MS85] achieve an approximation ratio of $2-\frac{2\ln\ln n}{\ln n}(1-o(1))$, where n is the number of vertices in the graph. It is conjectured that no polytime algorithm approximates VertexCover to within a factor of $2-\Omega(1)$.

Håstad [Hås97, Hås01] proved that no polytime algorithm approximates VertexCover to within a factor strictly less that $\frac{7}{6}$. Dinur [Din99] and Dinur and Safra [DS02] improved this lower bound to $10\sqrt{5}-21$.

Recall that the complement of a minimum vertex cover is a maximum independent set, and vice versa. However, this does *not* imply a trivial relation between approximation results for the two problems.

On the other hand, since a clique in a graph is an independent set of the same size in the complement of that graph, an approximation result for one of Clique or IndependentSet implies the same result for the other problem.

For example, suppose that we know that, in a given graph on 100 vertices, the size of the minimum vertex cover is either 98 or 99. Then we have approximated the size of the minimum vertex cover to within a factor of $\frac{99}{98}$. However, this is equivalent to only having approximated the size of the maximum independent set to within a factor of 2.

In fact, we *will* obtain results for CLIQUE, INDEPENDENTSET, and VERTEXCOVER simultaneously, but this is due to the nature of the proof.

Theorem 6. *If* P $\neq$ NP*, then no polytime algorithm approximates:*

- CLIQUE *to within a factor strictly less than* $\frac{8}{7}$*;*
- INDEPENDENTSET *to within a factor strictly less than* $\frac{8}{7}$*;*
- VERTEXCOVER *to within a factor strictly less than* $\frac{17}{16}$*.*

Proof. Fix $\epsilon > 0$ and let y be an input instance for an NP-complete language L.

We can use the construction given in the proof of Theorem 5 to construct in polytime a multiset of 3-clauses $\mathcal{F}$ such that:

- if $y \in L$, then there exists a truth assignment that satisfies at least a $1 - \epsilon$ fraction of the 3-clauses in $\mathcal{F}$; and,
- if $y \notin L$, then every truth assignment satisfies fewer than a $\frac{7}{8} + \epsilon$ fraction of the 3-clauses in $\mathcal{F}$.

Let $m = |\mathcal{F}|$.

We use the standard Karp-reduction[CLRS01] from 3SAT to INDEPENDENTSET to obtain a graph G on $3m$ vertices such that G contains an independent set of size k if and only if k 3-clauses of $\mathcal{F}$ are simultaneously satisfiable. Notice that:

- if $y \in L$, then G contains an independent set of size $(1 - \epsilon)m$; and,
- if $y \notin L$, then G does not contain an independent set of size $\left(\frac{7}{8} + \epsilon\right) m$.

Since ϵ was chosen to be an arbitrary positive constant and

$$\lim_{\epsilon \to 0} \frac{1-\epsilon}{\frac{7}{8}+\epsilon} = \frac{8}{7},$$

this proves the first and second parts of the theorem.

The complement of an independent set is a vertex cover, and vice versa. Therefore:

- if $y \in L$, then G' has a vertex cover of size less than or equal to

  $$3m - (1-\epsilon)m = (2+\epsilon)m;$$

  and,

- if $y \notin L$, then G' does not have a vertex cover of size less than or equal to

  $$3m - \left(\frac{7}{8}+\epsilon\right)m = \left(\frac{17}{8}-\epsilon\right)m.$$

Since ϵ was chosen to be an arbitrary positive constant and

$$\lim_{\epsilon \to 0} \frac{\frac{17}{8}-\epsilon}{2+\epsilon} = \frac{17}{16},$$

this proves the third part of the theorem. □

4 Better Constant Lower Bounds for Clique

In this section, we improve our $\left(\frac{8}{7}-\epsilon\right)$-hardness result for Clique to show that Clique cannot be approximated in polytime to within any constant, unless $\mathrm{P} = \mathrm{NP}$.

Theorem 7. *If* $\mathrm{P} \neq \mathrm{NP}$*, then, for any constant* $\alpha \geq 1$*, no polytime algorithm approximates* Clique *to within a factor of* α*.*

Proof. Suppose that we want to estimate the size of the largest clique in a given graph G. Let G_1 be the composition of G with itself. That is, G_1 is the graph obtained by replacing every vertex in G by a copy of G. An example is illustrated in Figure 1, with heavy lines

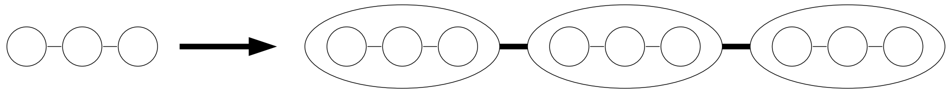

Figure 1: G_1 is the composition of G with itself.

denoting graph joins. A *graph join* of two graphs makes every vertex in the first graph adjacent to every vertex in the second, and vice versa.

Notice that the largest clique in G_1 is the composition of the largest clique in G with itself. Therefore, the size of the largest clique in G_1 is the square of the size of the largest clique in G. This gives us a lower bound of $\left(\frac{8}{7} - \epsilon\right)^2$ for the approximation ratio of any polytime algorithm. We can repeat this construction any constant number of times, to get as large a constant as we like for the lower bound. $\square$

Notice that, if we repeat the construction described in the proof of Theorem 7 more than a constant number of times, then the resulting graph will be more than polynomial in size and the construction will take more than polytime.

5 The Beginnings of a Nearly Linear Lower Bound for Clique

We have shown that, if P $\neq$ NP, then Clique cannot be approximated in polytime to within any constant factor. At first, this sounds like a reasonable lower bound. However, the best upper bound known [BH90, BH92] is much larger, $O\left(\frac{n}{(\log n)^2}\right)$, where n is the number of vertices in the graph. This is only slightly better than the trivial n-approximation algorithm that always returns 1 (for non-empty graphs).

Feige, Goldwasser, Lovász, Safra, and Szegedy [FGL$^+$91, FGL$^+$96] was the first paper to show a connection between PCP results (although they weren't called that at the time) and inapproximability results. Define quasi-polynomial time to be time $n^{(\log n)^{O(1)}}$, and define QP to be the class of languages that can be accepted in quasi-polynomial time. FGLSS showed that if NP $\not\subseteq$ QP, then for all $\epsilon > 0$, no quasi-polytime algorithm approximates Clique to within a factor of $2^{(\lg n)^{1-\epsilon}}$ (where n is the number of vertices).

Håstad [Hås96, Hås99] showed that, for all $\epsilon > 0$, no polytime algorithm approximates Clique to within a factor of

- $n^{\frac{1}{2}-\epsilon}$, if P $\neq$ NP, or
- $n^{1-\epsilon}$, if ZPP $\neq$ NP.

(Recall that ZPP is the class of languages decidable by 0-sided-error randomized algorithms that run in expected polytime.)

Instead of the construction given in the proof of Theorem 7, we will use a construction that increases the size of the gap without increasing the size of the graph by so much. First, we give an alternate proof of Theorem 7. Then we will modify this proof to give a better inapproximability result. This will be completed in the next lecture. The construction we use now is from the paper by Feige, Goldwasser, Lovász, Safra, and Szegedy.

Alternate Proof of Theorem 7. *If* P $\neq$ NP*, then, for any constant* α*, no polytime algorithm approximates* Clique *to within a factor of* α*.*

Proof. Let L be an NP-complete language. Assume that, for some s and b that we will specify in the next lecture and some constant d, $L \in \mathrm{PCP}_{1,s}(d \lg n, b)$ via verifier A. For now, we can think of s and b as being constants. (Notation: $\lg n$ means $\log_2 n$.)

Suppose that we want to decide membership in L of an instance y, with $|y| = n$. Let $r_1, \ldots, r_{n^d}$ be the n^d binary strings of length $d \log n$; these correspond to the possible choices that A might make for its random bits.

Define an *accepting transcript* for V on y with random bits r_i to be a sequence of b pairs $(q_1, a_1), \ldots, (q_b, a_b)$ such that, for every proof π that has values $a_1, \ldots, a_b$ in positions $q_1, \ldots, q_b$, $A(y, \pi)$ with random bits r_i examines those bits of π, in that order, and accepts.

We construct a graph G whose vertices are pairs (i, σ) consisting of

1. an integer i between 1 and n^d, and
2. an accepting transcript σ for A on y with random bits r_i.

Notice that, given y, r_i, and $a_1, \ldots, a_b$, it is possible to compute $q_1, \ldots, q_b$. It follows that there are at most 2^b accepting transcripts for V on y with random bits r_i. Therefore, since there are n^d choices of random bits, G has at most $2^b n^d$ vertices.

We put an edge between two vertices (i, σ) and (i', σ) of G if σ and σ' do not assign different truth values to any bit of the proof. Note that if there is an edge between two different vertices (i, σ) and (i', σ'), then $i \neq i'$.

Suppose that $y \in L$. Then there exists a proof π such that $A(y, \pi)$ accepts for all choices of random bits. Therefore, for each choice of random bits r_i, there is an accepting transcript of $A(y, \pi)$ with random bits r_i, so there is a clique of size n^d in G.

Now suppose that $y \notin L$. Then, for every proof π, $A(y, \pi)$ accepts for fewer than an s fraction of the choices for random bits. We claim that this implies that every clique in G is of size less than sn^d.

Suppose not. Then there exists a clique $(i_1, \sigma_1), \ldots, (i_t, \sigma_t)$ in G, for some $t \geq sn^d$. Notice that, since $\sigma_1, \ldots, \sigma_t$ do not assign different truth values to the same bit of the proof, they can be combined and extended to form a proof π such that $A(y, \pi)$ accepts with any of the random strings $r_{i_1}, \ldots, r_{i_t}$. This means that $A(y, \pi)$ accepts with probability at least s, which is a contradiction.

This gives us that, if P $\neq$ NP, then no polytime algorithm approximates Clique to within a factor of $\frac{1}{s}$.

Using repeated simulation, we can make s any constant greater than 0 at the cost of increasing the number of random bits used and the number of bits of the proof examined by a constant factor. Thus, choosing $s = 1/\alpha$ proves the claim. $\square$

This construction presents the same problem as the construction in our original proof of Theorem 7. That is, the graph is too large compared to the gap. In the rest of this lecture and in the next lecture, we will describe how to decrease the soundness parameter s without the size of the graph growing too quickly.

Construction:

As above, let $L \in \mathrm{PCP}_{1,s}(d \lg n, b)$ via verifier A. Let us assume (without loss of generality) that the proof length is n^d; this will be important later.

For some k that we will specify in the next lecture, let V be the probabilistic polytime verifier that:

1. runs $k \lg n$ simulations of A using new random bits for each and
2. accepts if and only if all of the simulations accept.

Then $L \in \mathrm{PCP}_{1,s^{k \lg n}}\,(dk(\lg n)^2, bk \lg n)$ via V, where the proof length is still n^d.

Notice that applying the construction described in the proof above to V yields a lower bound of $\frac{1}{s^{k \lg n}} = n^{k \lg \frac{1}{s}}$, at the cost of the graph being of size

$$n^{dk \lg n} 2^{bk \lg n} = n^{dk \lg n} n^{bk}.$$

If we could greatly reduce the number of random bits used by the verifier, so that we could effectively diminish the role of d in the above expression, then we would be close to our first goal: a clique gap equal to the number vertices raised to a constant power.

Instead of a deterministic reduction that uses the verifier V, we will use a *probabilistic* reduction that uses randomness to choose a verifier V'; V' will be related to V, but will use many fewer random bits.

We uniformly at random choose a sequence of m binary strings $R = r_1, \ldots, r_m$, each of length $dk(\lg n)^2$. We will explain later what values m and k should have. Notice that the choice of R is made as part of the reduction and is not made by a verifier. Therefore, the $mdk(\lg n)^2$ random bits required will not be included in the number of random bits used by our new verifier.

Using a probabilistic reduction will require us to strengthen the assumption we use, from $\mathrm{P} \neq \mathrm{NP}$ to something involving a randomized complexity class.

The probabilistically chosen $R = r_1, \ldots, r_m$ determines the verifier V' that, given y and a proof π:

1. uniformly at random chooses a number i from $\{1, 2, \ldots, m\}$; and
2. accepts if and only if $V(y, \pi)$ accepts with random bits r_i.

Notice that V' uses only $\lg m$ random bits.

In the next lecture, we will show how we can choose m such that, if $y \notin L$, then with probability of at least $1/2$ over the choice of R, for all proofs π, $V'(y, \pi)$ accepts with probability only slightly greater than s. It turns out that m will be small enough so that the fact that V' uses only $\lg m$ randomness will give us a clique gap equal to the number vertices raised to a constant power.

References

[BH90] R. Boppana and M. Halldórsson. Approximating maximum independent sets by excluding subgraphs. In *Proceedings of the Scandinavian Workshop on Algorithm Theory*, pages 13–25, 1990.

[BH92] R. Boppana and M. Halldórsson. Approximating maximum independent sets by excluding subgraphs. *BIT*, 32:180–186, 1992.

[BYE85] R. Bar-Yehuda and S. Even. A local-ratio theorem for approximating the weighted vertex cover problem. *Annals of Discrete Mathematics*, 25:27–46, 1985.

[CLRS01] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press/McGraw-Hill, 2nd edition, 2001.

[Din99] I. Dinur. *On the Hardness of Approximating the Minimum Vertex Cover and the Closest Vector in a Lattice*. PhD thesis, Tel-Aviv University, 1999.

[DS02] I. Dinur and S. Safra. The importance of being biased. In *Proceedings of the ACM Symposium on Theory of Computing*, pages 33–42, 2002.

[FGL+91] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Approximating clique is almost NP-complete. In *Proceedings of the IEEE Symposium on Foundations of Computer Science*, pages 2–12, 1991.

[FGL+96] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Interactive proofs and the hardness of approximating cliques. *Journal of the ACM*, 43:268–292, 1996.

[GLST98] V. Guruswami, D. Lewin, M. Sudan, and L. Trevisan. A tight characterization of NP with 3 query PCPs. In *Proceedings of the IEEE Symposium on Foundations of Computer Science*, pages 8–17, 1998.

[Gur99] V. Guruswami. Query-efficient checking of proofs and improved PCP characterizations of NP. Master's thesis, Massachusetts Institute of Technology, 1999.

[Hås96] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. In *Proceedings of the IEEE Symposium on Foundations of Computer Science*, pages 627–636, 1996.

[Hås97] J. Håstad. Some optimal inapproximability results. In *Proceeding of the ACM Symposium on Theory of Computing*, pages 1–10, 1997.

[Hås99] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica*, 182:105–142, 1999.

[Hås01] J. Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48:798–859, 2001.

[MS85] B. Monien and E. Speckenmeyer. Ramsey numbers and an approximation algorithm for the vertex cover problem. *Acta Informatica*, 22:115–123, 1985.