18.405J/6.841J Advanced Complexity Theory	May 3, 2001
Lecture 21	
Lecturer: Dan Spielman	*Scribe: Hiro Iwashima*

In this lecture, we will use NP $\subseteq$ PCP(($logn$),O(1)) to prove the following theorem:

Theorem 1 *it is NP-hard to approximate Max-Clique within n^ϵ for some $\epsilon > 0$*

We will start with something easier. We will show hardness within a factor of 2

Approximating Max-Clique

The basic idea

- $3SAT \in PCP(log, 1) \Rightarrow \exists$ Ptime verifier V, input Π,ϕ, and $V^\Pi(\phi)$ that uses $logn$ random bits, reads $O(1)$ bits of proof s.t.

$$\phi \in 3SAT \Rightarrow \exists \text{ proof } \Pi \; Pr[V^\Pi(\phi) \text{ accepts}] = 1$$
$$\phi \notin 3SAT \Rightarrow \forall \Pi \; Pr[V^\Pi(\phi) \text{ accepts}] < \frac{1}{2}$$

- want to transform $\phi \rightarrow$ graph if ϕ satisfiable, then the graph has a large clique

V: random bits choose the bits to read $\Rightarrow$ non-adaptive. Let us assume that V is non-adaptive, V uses R random bits, and it makes Q queries. We will construct the reduction in the following way:

Graph nodes

Graph nodes indexed by $\{0,1\}^R$ x $\{0,1\}^Q \Rightarrow 2^{R*Q}$, where r corresponds to the random string on string r, V reads bits $\Pi_{b_{r,1}}, ..., \Pi_{b_{r,Q}}$, where $b_{r,i}$ are indices. now given Q bits $\Rightarrow 2^Q$ ways of setting these bits. node(r,q) corresponds to setting $\Pi_{b_{r,i}}$ to q_i for $i \in 1..Q$

Graph edges

put an edge from (r,q) to (r',q') if it corresponds to a consistent settings of bits in proof. i.e. there do not exist i,j s.t. $b_{r,i} = b_{r',j'}$ and $q_i \neq q_j$
$\Rightarrow$ nodes in a row are inconsistent
$\Rightarrow$ bits read on r and r' are disjoint

$\Rightarrow$ only kep settings that cause verifiers to accept

Example:

Π_1, Π_2, Π_3, R=1, Q=2
V:0 reads Π_1 and Π_2, accept if they are equal
V:1 reads Π_2 and Π_3, accept if they are NOT equal

Fact1:

if $\phi \in 3SAT, \exists\Pi$ s.t. verifier accepts: Pr[V acc]= 1
from Π_1, get clique of size 2^R
(get one node in every row)

Fact2:

a clique in a graph corresponds to a partial assignment of $\Pi_1, ..., \Pi_N$, moreover, if clique-size is S, then this assignment can be extended to a Π s.t.

$$Pr[V^n(\phi)] \geq \frac{S}{2^R}$$

extend the assignment arbitrariliy $\rightarrow$ at most one node in each row, each node represents one setting of random bits for which V will accept Π

but need $P < \frac{1}{2}$ if $\phi \notin 3SAT$
$\phi \notin 3SAT \Rightarrow MaxClique \leq \frac{2^R}{2}$

In this reduction, input ϕ, output graph G, important that V runs in Ptime, number nodes in $G \subseteq poly(n) = 2^{R+Q} = 2^{O(logn)+O(1)} = n^{O(1)}$

$$\phi \in 3SAT \Rightarrow Max\_Clique(G) = 2^R$$
$$\phi \notin 3SAT \Rightarrow Max\_Clique(G) \leq \frac{2^R}{2} \Rightarrow \text{ NP-hard}$$

how do we raise ratio higher? ....
to get a factor better than 2, alter PCP system modify verifier to repeat k times, accept only if accept on each run.

$$\begin{aligned} \phi \in SAT \Rightarrow \exists \Pi \text{ Pr[acc] } &= 1 \\ \phi \in SAT \Rightarrow \forall \Pi \text{ Pr[acc] } &< 2^{-k} \\ &\rightarrow (2^R, \frac{2^R}{2^k}) \end{aligned}$$

as long as k = O(1), okay
if naively repeat it $O(logn)$ times, now use $O(log^2 n)$ random bits, $O(logn)$ queries
$2^{O(log^2 n)}$ ... no longer polynomial... but can reuse random bits by walk-on-expander! $O(logn + k(n))$ bits to produce $O(k(n))$ pseudo-random strings and error probability $2^{-k(n)}$. Setting

$$k(n) = O(logn)...say k(n) = log(n)$$

we get:

$$\begin{aligned} R = O(logn) &\text{ random bits} \\ Q = O(logn) &\text{ queries} \\ \phi \in SAT \Rightarrow \text{ Pr[acc]} &= 1 \\ \phi \in SAT \Rightarrow \text{ Pr[acc]} &< \frac{1}{n} \end{aligned}$$

If N = # nodes in graph

$$\leq 2^{R+Q} = 2^{O(logn)} = n^c \text{ for some c}$$

So NP-hard to distinguish between Max_Clique of 2^R or $\frac{2^R}{n}$ where $n = N^{\frac{1}{\epsilon}}$, $\epsilon = \frac{1}{c}$

Reducing satisfiable clauses in 3CNF

Recall from before, showed MAX 3SAT hard to approximate reduction:

$\phi \rightarrow \phi', \exists \epsilon :$

$$\phi \in 3SAT \Rightarrow \phi' \in 3SAT$$

$\phi \notin 3SAT \Rightarrow \forall$ settings of vars, of most $(1-\epsilon)$ fraction of clauses of ϕ' were satisfiable.

Moreover, each var in ϕ' appears in at most 3 clauses.

first show this 3SAT is NP hard

Lemma: can take any 3CNF ϕ and transform it to ϕ' such that each var appears at most 3 times in ϕ',

$$\phi \in 3SAT <=> \phi' \in 3SAT$$

consider x, say appears k times (k clauses), create $X_{1,1}, ..., X_{1,k}$ put in clauses. Add clauses $X_{1,1} \rightarrow X_{1,2} \rightarrow ... \rightarrow X_{1,k} \rightarrow X_{1,1}(\not X_{1,1} \vee X_{1,2})$

Problem:
if at most $1-\epsilon$ fraction of clauses satisfiable in ϕ, at most $1-\frac{\epsilon}{k}$ clauses satisfiable in ϕ'.
idea:
put constant degree d expander on $X_{1,1}, ..., X_{1,k}$
for each edge, do implications both ways
property:
if $s \leq \frac{k}{2}$, then every set of s nodes has at least s edges connecting it to rest of graph

example: hypercube

Say ϕ had M clauses, could not satisfy more than $(1-\epsilon) * m$. ϕ' has $m + 3m + 2d$ clauses, again conclude it's impossible to simultaneously satisfy all but $\epsilon * m$ of them: $6 * d + 1$ clauses $\Rightarrow (1 - \frac{\epsilon}{6d+1}) = frac$