In this lecture, we will use \( \text{NP} \subseteq \text{PCP}((\log n), O(1)) \) to prove the following theorem:

**Theorem 1** it is \( \text{NP-hard to approximate Max-Clique within } n^\epsilon \) 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

- \( 3\text{SAT} \in \text{PCP}(\log, 1) \Rightarrow \exists \) Prime verifier \( V \), input \( \Pi, \phi \), and \( V^\Pi(\phi) \) that uses \( \log n \) random bits, reads \( O(1) \) bits of proof s.t.

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

- want to transform \( \phi \rightarrow \) graph if \( \phi \) 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 \( \{0, 1\}^R \times \{0, 1\}^Q \Rightarrow 2^{R+Q} \), where \( r \) corresponds to the random string on string \( r \), \( V \) reads bits \( \Pi_{b_{r, 1}}, \ldots, \Pi_{b_{r, Q}} \), 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 let settings that cause verifiers to accept

#### Example:

\( \Pi_1, \Pi_2, \Pi_3, R=1, Q=2 \)

- \( V \) reads \( \Pi_1 \) and \( \Pi_2 \), accept if they are equal
- \( V \) reads \( \Pi_2 \) and \( \Pi_3 \), accept if they are NOT equal

#### Fact 1:

if \( \phi \in 3\text{SAT}, \exists \Pi \) s.t. verifier accepts: \( \Pr[V \text{ acc}] = 1 \)

from \( \Pi_1 \), get clique of size \( 2^R \)

(get one node in every row)
Fact2:

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

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

extend the assignment arbitrarily \( \rightarrow \) at most one node in each row, each node represents one setting of random bits for which \( V \) will accept \( \Pi \)

but need \( P < \frac{1}{2} \) if \( \phi \notin 3\text{SAT} \)

\( \phi \notin 3\text{SAT} \Rightarrow \text{MaxClique} \leq \frac{2^R}{2^k} \)

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

\[
\phi \in 3\text{SAT} \Rightarrow \text{MaxClique}(G) = 2^R
\]

\( \phi \notin 3\text{SAT} \Rightarrow \text{MaxClique}(G) \leq \frac{2^R}{2^{k^2}} \Rightarrow \text{NP-hard} \)

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.

\[
\phi \in \text{SAT} \Rightarrow \exists \Pi \ Pr[\text{acc}] = 1
\]

\( \phi \in \text{SAT} \Rightarrow \forall \Pi \ Pr[\text{acc}] < 2^{-k} \)

\( \rightarrow (2^R, \frac{2^R}{2^k}) \)

as long as \( k = O(1) \), okay

if naively repeat it \( O(\log n) \) times, now use \( O(\log^2 n) \) random bits, \( O(\log n) \) queries

\( 2^{O(\log^2 n)} \) ... no longer polynomial... but can reuse random bits by walk-on-expander! \( O(\log n + k(n)) \) bits to produce \( O(k(n)) \) pseudo-random strings and error probability \( 2^{-k(n)} \). Setting

\[
k(n) = O(\log n) \ldots \text{say} k(n) = \log(n)
\]

we get:

\[
R = O(\log n) \text{ random bits}
\]

\[
Q = O(\log n) \text{ queries}
\]

\( \phi \in \text{SAT} \Rightarrow \Pr[\text{acc}] = 1 \)

\( \phi \in \text{SAT} \Rightarrow \Pr[\text{acc}] < \frac{1}{n} \)

If \( N = \# \) nodes in graph

\[
\leq 2^{R+Q} = 2^{O(\log n)} = n^c \text{ for some } c
\]

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

22-2
Reducing satisfiable clauses in 3CNF

Recall from before, showed MAX 3SAT hard to approximate reduction:
\[ \phi \rightarrow \phi', 3\epsilon : \]

\[ \phi \in 3SAT \Rightarrow \phi' \in 3SAT \]
\[ \phi \notin 3SAT \Rightarrow \forall \text{ settings of vars, of most } (1 - \epsilon) \text{ fraction of clauses of } \phi' \text{ were satisfiable.} \]

Moreover, each var in \( \phi' \) appears in at most 3 clauses.

first show this 3SAT is NP hard

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

\[ \phi \in 3SAT \leq \Rightarrow \phi' \in 3SAT \]

consider \( x \), say appears \( k \) times (k clauses), create \( X_{1,1}, ..., X_{1,k} \) put in clauses. Add clauses \( X_{1,1} \rightarrow X_{1,2} \rightarrow ... \rightarrow X_{1,k} \rightarrow X_{1,1}(\neg X_{1,1} \vee X_{1,2}) \)

Problem:
if at most \( 1 - \epsilon \) fraction of clauses satisfiable in \( \phi \), at most \( 1 - \frac{\epsilon}{k} \) clauses satisfiable in \( \phi' \).

idea:
put constant degree \( d \) expander on \( X_{1,1}, ..., X_{1,k} \)
for each edge, do implications both ways

property:
if \( s \leq \frac{\epsilon}{k} \), then every set of \( s \) nodes has at least \( s \) edges connecting it to rest of graph

example: hypercube

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