The lecture will follow the paper by Irit Dinur [1].

1. Preliminary Definitions

Definition 1 (NP-completeness). A language $L$ is NP-complete under many-one reduction if for any language in NP there is a polynomial time many-one reduction to $L$. Examples include SAT, 3-SAT, 3-colorability.

The 3-colorability problem is defined by the following input, output structure
Input: A graph $G$
Output: “Yes” iff the vertices of $G$ can be assigned one of three colors such that no two vertices of any edge get the same color.

2. PCP Theorem

We present the prosaic form of the PCP version first.

Definition 2. Let $\phi$ be an instance of SAT. For $\tau$ a truth assignment, we define $\text{UNSAT}_\tau(\phi) = \text{fraction of } \phi \text{'s clauses false under } \tau$. We define $\text{UNSAT}(\phi) = \min \text{UNSAT}_\tau(\phi)$.

Theorem 1 (PCP Version 1: Prosaic Form). There is a constant $\delta < 1$ and $k \geq 3$ (in fact $k = 3$ works too) such that if $L$ satisfies the following conditions then $L$ is NP-complete.

1. If $\phi$ is an instance of $k$-SAT and if $\phi$ is satisfiable, then $\phi \in L$.

2. If $\phi$ is an instance of $k$-SAT and $\phi$ is unsatisfiable, then $\text{UNSAT}(\phi) \geq \delta \implies \phi \in L$.

Loosely speaking, this implies that approximating the number of $\phi$’s clauses that can be simultaneously satisfied is NP hard.

Definition 3. A language $L$ is in PCP$(r(n), q(n))$, iff
1. There is a polynomial time verifier \( V = V(x, \pi) \) which chooses \( \leq r(n) \) bits at random, examines only \( q(n) \) bits of \( \pi \) and either accepts or rejects.

2. “Completeness”: For \( x \in L \), \( \exists \pi \) s.t. \( \Pr[V(x, \pi) \text{ accepts}] = 1 \).
   “Soundness”: For \( x \notin L \), \( \forall \pi, \Pr[V(x, \pi) \text{ accepts}] \leq 1/2 \)

**Theorem 2** (PCP Version 2). \( SAT \subseteq PCP(O(\log n), O(1)) \)

This version of the PCP theorem, justifies the name Probabilistically Checkable Proofs.

The proof \( \pi \) is supplied by the Prover. The verifier \( V \) choses some \( O(\log n) \) random bits and accepts/rejects by seeing \( O(1) \) bits of \( \pi \).

Without loss of generality, it can be assumed that \( \pi \) has at most \( q(n)2^r(n) \) many bits. In particular, for PCP, \( |\pi| = n^{O(1)} \). In this lecture, we will prove the equivalence of the two versions of the PCP theorem. The actual proof of the PCP theorem will be presented in the subsequent lectures.

### 3. Equivalence of versions of PCP

**Theorem 3.** The two versions of the PCP theorem are equivalent

**Proof.** Case 1: Version 1 implies Version 2.

Suppose \( SAT \) is many-one polynomial-time reducible to any language \( L \) of version 1. Then we have a poly time function \( f(x) \) such that \( \forall x \in \{0,1\}^* \), if \( x \in SAT \), then \( f(x) \in SAT \), if \( x \notin SAT \), then \( UNSAT(f(x)) \geq \delta \). The PCP protocol for SAT is as follows.

**Input:** \( x \)

**Output:** Decide if \( x \in SAT \)

**Algorithm:**

**Step 1:** Compute \( \phi = f(x) \)

**Step 2:** Repeat \( N \) times:

- Choose a clause of \( \phi \) at random.
- For each literal \( x_j \) in \( \phi \)
  - look up its value as \( j^{th} \) bit of \( \pi \).
  - If clause is not satisfied, Reject

**Step 3:** Accept

**Algorithm 1:** PCP protocol for SAT

\( N \) is chosen large enough so that \((1 - \delta)^N < 1/2\).

This shows that \( SAT \subseteq PCP(O(\log n), O(1)) \) which implies that \( NP \subseteq PCP(O(\log n), O(1)) \).

Assume $SAT$ has a $PCP(c_1 \log n, c_2)$ protocol $V$ where $c_1, c_2$ are constants.

We want to form a many-one reduction $f(x)$.

The reduction $f(x)$ works as follows.

1. For each choice $r$ of the $c_1 \log n$ random bits (there are $2^{c_1 \log n} = n^{O(1)}$ many $r$’s):
   Run $V(x, \pi, r)$ which examines $c_2$ of the bits of $\pi$. Form an instance of $SAT, \phi_r$ where
   (a) Variables of $\phi_r$ are bits of $\pi$ examined.
   (b) Values of $\phi_r$ is true iff $V(x, \pi, r)$ accepts.

   (Size of $\phi_r$ is $O(1)$).

2. Let $f(x) = \bigwedge \phi_r$ ($\phi_r$’s are CNF formulas).
   Since this was a PCP reduction, either there is a $\pi$ that satisfies all $\phi_r$’s or every $\pi$ has at least fraction 1/2 of the $\phi_r$’s false. If each $\phi_r$ has $\leq k$ ($k = k(c_2)$), then $UNSAT(f(x)) \geq \frac{1}{k^2}$. We can then take $\delta = 1/2k$.

This completes the proof.

References