**CS 6810: Theory of Computing** **Spring 2026**

# Lecture 22: April 16, 2026

*Lecturer: Eshan Chattopadhyay* *Scribe: Saket Murali*

## 1 Probabilistically Checkable Proofs (PCPs)

Motivation: Throughout this class so far, we have studied how decision problems can be NP-complete or even undecidable. Since the 1980s, people have wanted to know whether we can approximate such hard problems. PCPs give us *hardness of approximation* results.

**Definition 1.1.** *$L \in \mathsf{PCP}(r, q)$ if there exists a randomized, polynomial-time query access algorithm $V$ that uses $\leq r$ bits of randomness and makes $\leq q$ queries to a proof string $\pi$, satisfying:*

- ***Completeness:*** $x \in L \Rightarrow \exists\, \pi$ *s.t.* $\Pr[V^\pi(x) = 1] = 1$.
- ***Soundness:*** $x \notin L \Rightarrow \forall\, \pi,\ \Pr[V^\pi(x) = 1] < \frac{1}{2}$.

*Here $V^\pi(x)$ denotes query access to $\pi$, and $V$ is a probabilistic Turing machine.*

**Observation 1.2.** $\mathsf{PCP}(\log n, O(1)) \subseteq \mathsf{NP}$.

*Proof.* Since $V$ uses $r = O(\log n)$ bits of randomness, there are $2^r$ different random strings, so $V$ can access at most $q \cdot 2^r$ locations in $\pi$. Thus $|\pi| \leq q \cdot 2^r = O(1) \cdot n^{O(1)} = \text{poly}(n)$, so $\pi$ is of polynomial length.

Therefore, an NP verifier $V_{\mathsf{NP}}$ can take $\pi$, simulate $V_{\mathsf{PCP}}$ on all possible random strings, and reject if any execution of $V_{\mathsf{PCP}}$ rejects. □

Before we think about proving what PCP class NP is contained in, we must study some techniques, including **linearity testing**. We will use these techniques to prove that $\mathsf{NP} \subseteq \mathsf{PCP}(O(n), O(1))$ in the next class. Note that the O(n) parameter is easier to prove than O(log n); intuitively, it makes sense that as the proof length increases, the same number of queries suffices to obtain enough information. This is because a succinct proof requires the verifier to pay attention to every detail, whereas a longer proof has enough redundancy that a constant number of random spot-checks can still catch a cheating prover.

## 2 Linearity Testing

We consider query access to a function $f : \{0,1\}^n \to \{0,1\}$. Note that $\{0,1\}^n \equiv \mathbb{F}_2^n$.

**Question:** Is $f$ linear? That is, does $f(x + y) = f(x) + f(y)$ for all $x, y \in \mathbb{F}_2^n$ (where addition is over $\mathbb{F}_2$)?

**Definition 2.1.** *For $f, g : \{0,1\}^n \to \{0,1\}$, define*

$$\text{dist}(f, g) = \Pr_{x \in \{0,1\}^n}[f(x) \neq g(x)].$$

*Let $\mathsf{LIN} = \{f : \{0,1\}^n \to \{0,1\} \mid f$ is linear$\}$, Define*

$$\text{dist}(f, \mathsf{LIN}) = \min_{g \in \mathsf{LIN}} \text{dist}(f, g).$$

**Definition 2.2.** *A randomized query access algorithm $A$ is a $(q, \lambda)$-*linearity tester *if:*

- ***Completeness:*** $f \in \mathsf{LIN} \Rightarrow \Pr[A^f = 1] = 1$.
- ***Soundness:*** $f \notin \mathsf{LIN} \Rightarrow \Pr[A^f \text{ rejects}] \geq \lambda \cdot \text{dist}(f, \mathsf{LIN})$.

*and $A$ makes at most $q$ queries.*

Note: The goal is to have $q$ and $\lambda$ be constants. This is ambitious, since $f$ is over $2^n$ inputs and linearity is a very global property.

## 2.1 The BLR Test

**BLR Test (Blum, Luby, Rubinfeld):** Randomly pick $x, y \in \mathbb{F}_2^n$.

- **Accept** if $f(x + y) = f(x) + f(y)$.
- **Reject** if $f(x + y) \neq f(x) + f(y)$.

**Theorem 2.3.** *The BLR test is a $\left(3, \frac{2}{9}\right)$-linearity tester.*

*Proof.* **Completeness** is clear: if $f$ is linear, then $f(x + y) = f(x) + f(y)$ always holds.

**Soundness:** Suppose $f \notin \mathsf{LIN}$. We want to show:

$$\Pr[\mathsf{BLR}^f \text{ rejects}] \geq \tfrac{2}{9} \cdot \text{dist}(f, \mathsf{LIN}).$$

Define a helper function $g : \{0,1\}^n \to \{0,1\}$ by

$$g(x) = \begin{cases} 1 & \text{if } \Pr_{y \in \{0,1\}^n}[f(x+y) + f(y) = 1] \geq \frac{1}{2} \\ 0 & \text{else.} \end{cases}$$

Note: if $f$ were linear, then $f(x + y) + f(y) = f(x)$, so the definition would reduce to $g(x) = f(x)$.

We use the following two claims to prove soundness.

**Claim 2.4.** $f \notin \mathsf{LIN} \Rightarrow \Pr[\mathsf{BLR}^f \text{ rejects}] \geq \frac{1}{2} \cdot \text{dist}(f, g)$.

**Claim 2.5.** $\Pr[\mathsf{BLR}^f \text{ rejects}] < \frac{2}{9} \Rightarrow g$ *is linear.*

**Proving soundness from Claims 2.4 and 2.5:**

- If $\Pr[\mathsf{BLR}^f \text{ rejects}] \geq \frac{2}{9}$, then

$$\Pr[\mathsf{BLR}^f \text{ rejects}] \geq \frac{2}{9} \geq \frac{2}{9} \cdot \text{dist}(f, \mathsf{LIN})$$

  (since $\text{dist}(f, \mathsf{LIN}) \leq 1$).

- If $\Pr[\mathsf{BLR}^f \text{ rejects}] < \frac{2}{9}$, then by Theorem 2.5, $g$ is linear, and by Theorem 2.4,

$$\Pr[\mathsf{BLR}^f \text{ rejects}] \geq \tfrac{1}{2} \cdot \text{dist}(f, g) \geq \tfrac{1}{2} \cdot \text{dist}(f, \mathsf{LIN}) \geq \tfrac{2}{9} \cdot \text{dist}(f, \mathsf{LIN}).$$

.

**Proof of Claim 2.4:**

Define $a(x) = \Pr_y[f(x) \neq f(x+y) + f(y)] : \{0,1\}^n \to \mathbb{R}$.

Observe: $a(x) > \frac{1}{2} \Leftrightarrow g(x) \neq f(x)$ (by definition of $g$).

Now,

$$\begin{aligned}\Pr[\mathsf{BLR}^f \text{ rejects}] &= \Pr_{x,y}[f(x) \neq f(y) + f(x+y)] \\ &= \mathbb{E}_x[a(x)] \\ &\geq \tfrac{1}{2} \cdot \Pr_x[g(x) \neq f(x)] \\ &= \tfrac{1}{2} \cdot \operatorname{dist}(f,g).\end{aligned}$$

**Proof of Claim 2.5:**

Define $b(x) = \Pr_y[g(x) \neq f(x+y) + f(y)]$.

*Key Claim:* $\Pr[\mathsf{BLR}^f \text{ rejects}] < \frac{2}{9} \Rightarrow \forall\, x,\ b(x) < \frac{1}{3}$.

*Proof of Key Claim.* Since $f(x+y) + f(y)$ agrees with $g(x)$ for a $(1 - b(x))$-fraction of $y$'s and disagrees for a $b(x)$-fraction:

$$\Pr_{y,y'}\big[f(x+y) + f(y) = f(x+y') + f(y')\big] = (1-b(x))^2 + b(x)^2 = 2b(x)^2 - 2b(x) + 1.$$

Also, over $\mathbb{F}_2$, $f(x+y)+f(y) = f(x+y')+f(y')$ is equivalent to $f(x+y)+f(x+y') = f(y)+f(y')$, and by a union bound:

$$\begin{aligned}\Pr_{y,y'}\big[f(x+y) + f(x+y') = f(y) + f(y')\big] \geq 1 &- \Pr_{y,y'}\big[f(y) + f(y') \neq f(y+y')\big] \\ &- \Pr_{y,y'}\big[f(x+y) + f(x+y') \neq f(y+y')\big].\end{aligned}$$

Both subtracted terms equal the BLR rejection probability (in each case the two inputs and their sum are uniform in $\mathbb{F}_2^n$), so both are strictly less than $\frac{2}{9}$. Hence:

$$2b(x)^2 - 2b(x) + 1 > 1 - \tfrac{2}{9} - \tfrac{2}{9} = \tfrac{5}{9},$$

which rearranges to $(3b(x) - 1)(3b(x) - 2) > 0$. Thus $b(x) < \frac{1}{3}$ or $b(x) > \frac{2}{3}$; since $b(x) \leq \frac{1}{2}$, we conclude $b(x) < \frac{1}{3}$.

Now we use the Key Claim to show $g$ is linear. We show $g(u+v) = g(u)+g(v)$ for all $u, v \in \{0,1\}^n$. Since $b(x) < \frac{1}{3}$ for all $x$, each of the following holds for more than $\frac{2}{3}$ of $y$'s:

$$\begin{aligned}g(u) &= f(u+y) + f(y), \\ g(v) &= f(v+y) + f(y), \\ g(u+v) &= f(u+y) + f(v+y).\end{aligned}$$

(The third equation follows from the definition of $b(u+v)$ under the substitution $y \mapsto v+y$.) Since each event fails for fewer than $\frac{1}{3}$ of $y$'s and $b(u)+b(v)+b(u+v) < 1$, by a union bound there exists $y^*$ for which all three hold simultaneously. Then:

$$\begin{aligned}g(u) + g(v) + g(u+v) &= \big[f(u+y^*) + f(y^*)\big] + \big[f(v+y^*) + f(y^*)\big] + \big[f(u+y^*) + f(v+y^*)\big] \\ &= 2f(u+y^*) + 2f(v+y^*) + 2f(y^*) = 0\end{aligned}$$

over $\mathbb{F}_2$, so $g(u+v) = g(u) + g(v)$ and $g$ is linear. $\square$