"Every NP statement has an exponentially long proof that can be locally tested by looking at just a constant number of bits."

This is off by an exponential in the proof size—we want a polynomial-size proof that can be checked with a logarithmic amount of randomness—but is still nontrivial.

**THM 11.19** \( \text{NP} \leq \text{PCP}_{1/2}^2(\text{poly}(n), 1) \)

To prove this theorem, need to find redundant encoding of proofs/assignments that can be checked with very few queries.

Suppose we have string \( u \in \{0,1\}^n = \text{GF}(2)^n \). We can view \( u \) as a linear function

\[
\begin{align*}
  u & : \{0,1\}^n 
  & \rightarrow \{0,1\}^n \\
  & x \mapsto u(x)
\end{align*}
\]

Encode \( u \) by writing down its function table

\[
\begin{array}{c|c}
  u(x) & x \in \{0,1\}^n
\end{array}
\]

string of length \( 2^n \).
Identify $u = (u_1, \ldots, u_n) \in \{0,1\}^n$ with function

$$x \mapsto u \cdot x = \sum_{i=1}^n u_i \cdot x_i \pmod{2}$$

$WH : \{0,1\}^n \to \{0,1\}^{2^n}$ defined by

$$WH(u) = \{ u \cdot x \mid x \in \{0,1\}^n \}$$

How many different strings in $\{0,1\}^k \equiv 2^k$? Hence there are $2^{2^n}$ strings $f \in \{0,1\}^{2^n}$.

Every such $f$ can be viewed as function table $f : \{0,1\}^n \to \{0,1\}$

Out of these $2^{2^n}$ functions/strings, only $2^n$ (a logarithmic number) are linear functions

So strings corresponding to $WH(u)$ for some $u \in \{0,1\}^n$ are very rare.

Refer to $WH(u)$ as codewords of the Walsh-Hadamard code.

An absolutely crucial fact is that any two codewords $WH(u)$, $WH(v)$, $u \neq v$, are very far from each other.
For \( f, g \in \{0, 1\}^k \) let
\[
\delta(f, g) = \Pr_{i \in [k]} [f_i \neq g_i] = \frac{1}{2^k} \left| \{ i \in [k] \mid f_i \neq g_i \} \right|
\]

**Lemma** For any \( u, v \in \{0, 1\}^n \), \( u \neq v \),
\[
\delta(\text{WH}(u), \text{WH}(v)) = \frac{1}{2}
\]
that is, any two codewords differ in exactly half of the coordinates.


Can also be phrased as follows:

**Random Subsum Principle**
If for \( u, v \in \{0, 1\}^n \) it holds that \( u \neq v \),
then for exactly half of all strings \( x \in \{0, 1\}^n \) it holds that
\[
u \cdot x \neq v \cdot x
\]

Aside about notation:
- Arora-Barak write \( u \circ v \)
- I will try to consistently write \( u \cdot v \)
- Also fairly common to write \( \langle u, v \rangle \)
  Slightly misleading since this is not an inner product (why?) but has many properties of an inner product.
Suppose we are given a function $f : \{0,1\}^n \to \{0,1\}^n$ (i.e., strong $f \in \{0,1\}^{2^n}$).

Want to inspect $f$ in constant # positions and decide whether $f$ is a Walsh-Hadamard codeword, i.e., whether $f$ is linear.

But we already know this can be done using the BKR linearity test:

Pick $x, y \in \{0,1\}^n$ uniformly and independently at random. Accept if

$$f(x+y) = f(x) + f(y)$$

and reject otherwise.

**Theorem 11.21**

If $f : \{0,1\}^n \to \{0,1\}^n$ is such that

$$\Pr_{x,y \in \{0,1\}^n} \left[ f(x+y) = f(x) + f(y) \right] \geq 1 - \delta$$

for some $\delta < \frac{1}{2}$, then there exists some linear function $L : \{0,1\}^n \to \{0,1\}^n$ such that $\delta(f, L) \leq \delta$. 
That is, \( f \) is \( \delta \)-close to some linear function.

**Aside about terminology**

Please note that our "\( \delta \)-close" is what Arora-Barak refer to as "\((1-\delta)\)-close".

Suppose that we want not only to test \( f \) for linearity but evaluate the linear function that \( f \) encodes. What if function table of \( f \) is slightly distorted so that some \( \delta \)-fraction of values have been corrupted?

Can we still evaluate \( f \) with constant number of queries to find value \( f(x) \), even if position \( x \) is corrupted?

Formally, suppose for \( \delta < \frac{1}{4} \) that \( f \) is \( \delta \)-close to some linear function \( f' \). Since linear functions have distance exactly \( \frac{1}{2} \) between each other, \( \tilde{f} \) is unique. Can we evaluate \( \tilde{f} \) with constant \# queries to \( f \)?
Use linearity!

1. Choose $x' \in \{0,1\}^n$
2. Set $x'' = x + x'$
3. Read $y' = f(x')$
   $y'' = f(x'')$
4. Output answer $\tilde{f}(x) = y' + y''$

$x'$ is uniformly distributed, so
$\Pr[ f(x') \neq \tilde{f}(x') ] \leq \delta$ by assumption.

$x''$ also uniformly distributed, so
$\Pr[ f(x'') \neq \tilde{f}(x'') ] \leq \delta$

$x'$ and $x''$ are NOT independent, but
for any events $A$ and $B$ the union bound says that

$$\Pr[ A \cup B ] \leq \Pr[ A ] + \Pr[ B ]$$

So except with probability $1 - 2\delta$
we have $y' = \tilde{f}(x')$ and $y'' = \tilde{f}(x'')$

$= \tilde{f}(x + x')$
in which case

$y' + y'' = \tilde{f}(x') + \tilde{f}(x + x')$

$= \tilde{f}(x)$

by linearity.
This is called LOCAL DECODING of the Walsh-Hadamard code, or SELF-CORRECTION.

Now we want to prove Thm 11.19

\[ NP \leq PCP_{1,\frac{1}{2}}(\text{poly}(n),1) \]

Sufficient to prove for one NP-complete language. We will use:

**QuadEq**

Given quadratic equations \( \{ E_1, \ldots, E_m \} \) over \( n \) variables, where \( E_i \) is of the form

\[
\sum_{i=1}^{n} \sum_{j=1}^{n} a_{i,j} u_i u_j = b_{i,j} \quad (*)
\]

(\( a_{i,j}, b_{i,j} \in \{0,1\} \)), is there a \( 0,1 \)-assignment to the \( u_i \)'s satisfying all equations?

**Proposition**

QuadEq is NP-complete (even if every equation \( E_i \) has just a constant number of nonzero coefficients)
**Proof Exercise.**

(Since \( u'_i = (u_i)^2 \) in \( \text{GF}(2) \), we can assume that the format is as in (*) above without any loss of generality.)

Hence, any \textsc{quordea} instance can be described by

\[
\begin{align*}
&\text{an} \quad m \times n^2 \text{-dim matrix } A \quad \text{(over \text{GF}(2))} \\
&\text{an} \quad m \text{-dim vector } b \quad \text{(over \text{GF}(2))}
\end{align*}
\]

Given assignment to \( u = (u_1, \ldots, u_n) \), assignment to degree-2 monomials given by

\[
(u_1, u_1, u_2, \ldots, u_1 u_n, u_2 u_1, \ldots, u_n u_1) =
(u_2, u_2, u_3, \ldots, u_n u_1) =
= u \otimes u \quad \text{TENSOR PRODUCT}
\]

For such \( u \), \( u \otimes u \) instance is satisfied if \( \, A \cdot u = b \).

Hence, \textsc{quordea} can be restated as the following problem:

**Given** \( A, b \) as above, find \( n^2 \)-dimensional vector \( u \) such that

1. \( A \cdot u = b \)
2. Exists \( u \) such that \( u = u \otimes u \)
**PCP, ATTEMPT 1**

**Proof**: Assignment $u \in \{0,1\}^n$

**Test**: Pick equation $E_{i,j}$ at random, read required $u_i, u_j$ (for any $i,j \neq 0$) and check if $E_{i,j}$ satisfied

**Completeness**: $OK, \gamma = \frac{1}{16}$

**Soundness error**: $\rightarrow \frac{5}{16}$

**Randomness**: logarithmic

**Queries**: $O(1)$ [assuming $k$-query]

---

**PCP, ATTEMPT 2**

**Proof**: $u \in \{0,1\}^n$

**Test**: For every equation $E_{i,j}$ flip a coin. Sum up all equations for which coin = 1

Check $\sum_{E \in S} \sum_{i,j} a_{E,i,j} u_i u_j = \sum_{E \in S} b_E$ (**+**)

**Completeness**: $OK, \gamma = 1$

**Soundness**: $OK, \gamma = \frac{1}{2}$

Given assignment $u$, at least one equation $E_{*,i,j}$ violated
Sum \( (\text{T}) \) for all other equations in chosen set \( S \). Then flip coin for \( e^* \). Two cases:

1. Sum \( (\text{T}) \) without \( e^* \) is satisfied. Then with prob \( = \frac{1}{2} \) \( e^* \) is included, which violates \( (\text{T}) \).

2. Sum \( (\text{T}) \) without \( e^* \) is already violated. Then with prob \( = \frac{1}{2} \) we don't pick \( e^* \) and \( (\text{T}) \) is also violated.

Randomness: Polynomial

Query complexity: Expected linear!

PCP, Attempt 3

Proof: Ask for \( L = u \oplus u \) encoded as \( WH(u) \)

Test: Pick subset \( S \) of equations and check \( (\text{T}) \) as before. Note that we know \( \sum e_i \geq 6e \) without querying anything. And

\[ \sum e_i \sum_{i,j} a_{e_i e_j} u_i u_j \]

is a linear function! Hence we
read it from $WH(U)$ in an error-correcting way, after first having checked that $\Pi = WH(U)$ is indeed an encoding of a linear function. # queried constant.

Completeness $OK, = 1$

Soundness seems OK... Except, how do we know $U = u \otimes u$? We totally don't... if $A$'s invertible, Xavier's error is 1/2

Randomness: polynomial

Query complexity: constant.

**PCP: ATTEMPT 4 (FINAL)**

$\Pi \in \{0, 1\}^{2^n + 2^{n^2}}$ encoding $WH(u)$ and $WH(U)$ (assuming that $U = u \otimes u$)

Let us denote $f = WH(u)$

$g = WH(U)$

Test (2) Test $f$ and $g$ repeatedly (but constant # times) until we are sure except with some probability $\varepsilon$ that $f$ is $0.001$-close to linear $f$ $g$ is $0.001$-close to linear $g$
For the remaining steps, assume $f$ and $g$ are close to these functions $\tilde{f}, \tilde{g}$ (otherwise we will fail, but this is only with probability $\varepsilon$).

2. $\tilde{f} = WH(n)$ for some $n$.
   $\tilde{g} = WH(U)$ for some $U$.
   
   Can read from $f$ and $g$ in error-correcting way.

   Check 10 times:
   
   Pick $r, r' \in \{0, 1\}^n$.
   
   Reject unless $\tilde{f}(r) \tilde{f}(r') = \tilde{g}(r \oplus r')$.

3. Pick random subset $S$ of equations as in (4) and check that (4) is satisfied by reading the corresponding position in $\tilde{g} = WH(U)$ from $g$ in an error-correcting way.

   Accept if the read bit is $= \sum_{i \in S} b_i$.

See pages 251-252 in Arora-Barak for the detailed analysis. Below follow the highlights.
Randomness $\text{poly}(n)$ - OK
Query complexity $\text{constant}$ - OK

Let us do completeness and soundness together. For the soundness, there is a satisfying assignment $u$ and $\Pi$ is the concatenation of $f = WH(u)$ and $g = WH(u \oplus u)$

Step 1 In the completeness case, the lineararity tests accept with prob 1

In the soundness case, we reject with probability $\geq 0.001$ if functions are not that close to linear. Pick some constant $K$ so that

$$1 - (1 - 0.001)^K > 1 - \epsilon/2$$

Then repeating both tests $K$ times we will reject $f, g$ that are not 0.001-close with probability $1 - \epsilon$

For the rest of the analysis, assume

$$\delta(f, \tilde{f}) = 0.001$$
$$\delta(g, \tilde{g}) = 0.001$$

for unique linear functions $f, \tilde{f}$
Step 2. In a correct proof \( \tilde{f} = f \), \( \tilde{f} = \tilde{g} \) and we have

\[
\tilde{f}(r) \tilde{f}(r') = \left( \sum_{i \in [n]} u_i r_i \right) \left( \sum_{j \in [n]} u_j r'_j \right)
\]

\[= \sum_{i,j} w_{ij} r_i r'_j \]

\[= (u \otimes u) \cdot (r \otimes r') \]

\[= \tilde{g}(r \otimes r') \]

which always accepts.

Suppose \( \tilde{g} = W^T(w) \) for \( w \neq u \otimes u \).

Write \( w \) as \( n^2 \)-matrix \( W = w_{ij} \).

Then

\[\tilde{g}(r \otimes r') = w \cdot (r \otimes r')\]

\[= \sum_{i,j} w_{ij} r_i r'_j \]

\[= r W r' \]

\[\tilde{f}(r) \tilde{f}(r') = (u \otimes r)(u \otimes r') \]

\[= \left( \sum_i u_i r_i \right) \left( \sum_j u_j r'_j \right) \]
\[
= \sum_{i,j,i',j'} u_{ij} r_{ij}^' \\
= r W r^'
\]

We reject if \( r W r' \neq r L L r' \)

By randomsubset principle, at least
half of all \( r \) satisfy

\[ r W = r L \quad (\dagger) \]

(by same argument as for check of \((\dagger)\) above)

Fix \( r \) st. \((\dagger)\) holds

For such \( r \), at least half of \( r' \)
satisfy

\[ (r W) r' \neq (r L L) r' \]

So with probability \( \geq 1/4 \) test rejects

Repeating 10 times, we get rejection

probability \( \geq 1 - (3/4)^{10} > 0.9 \)

(assuming all error-correcting results are
OK, but if this fails we can add this small probability to previous \( \varepsilon \)).
Step 3  If we haven't already rejected we can assume that there is some $u \in \text{EQ}_1$ such that $T \cong WH(u), WH(u \otimes u)$

where we denote $\tilde{f} = WH(u)$
$\tilde{g} = WH(u)$ for $u \otimes u = 0$

If the QUADEx instance is unsatisfiable, $u$ falsifies at least one constraint.

Picking random $S$ and checking $(t)$, we have 50% probability of detecting this.
Repeat whole test 2-3 times as needed to get down below 1/2 soundness error.

Formally, we would have to be a bit more careful with our probabilities and do calculations along the lines below.

Scenario: No-instance $E$ of QUADEx

Proof $T = f, g$

Case 1

$f$ and $g$ are not close to linear

Pr [failure to detect nonlinearity] $\leq \varepsilon_1$
Case 2 \( f(x) \approx f(y) \) does not hold

\[ Pr[\text{failure to detect this}] \leq \sum_{\text{error-correcting reads}} Pr[\text{read fails}] + \sum_{\text{correct reads}} Pr[\text{test fails given correct read}] \leq \sigma_1^2 + \sigma_2^2 \leq \varepsilon_2 \]

Case 3 \( f(x) \approx \text{encodings of } \text{WH}(u) \text{ and } \text{WH}(u \oplus u) \)

\[ Pr[\text{failure to detect a falsifying assignment}] \leq \sum_{\text{reads}} Pr[\text{read fails}] + \sum_{\text{reads}} Pr[\text{Test}(t) \text{ fails}] \leq \delta_1^2 + \delta_2^2 \leq \varepsilon_3 \]

Total failure probability \( \leq \max \{ \varepsilon_1, \varepsilon_2, \varepsilon_3 \} \leq \varepsilon_1 + \varepsilon_2 + \varepsilon_3 \)

(And usually \( \varepsilon \)-values can be made small enough anyway)