Last time: started unit on counting problems
- motivation, basics
- \#P, oracles, completeness

Today:
- more \#P-completeness, PERMANENT (PER)
- PER and average-case hardness random self-reducibility

(next time: approx counting)

A little more \#P-completeness:

**Def:** A Bool. formula is monotone if all its gates are \( \land \lor \) (no negation).

\[
\begin{array}{c}
\land, \lor \quad \text{"monotone"} \\
\text{increasing}
\end{array}
\]

\[
\begin{array}{c}
1 \land 0 = 0 \\
1 \land 1 = 1 \\
1 \lor 0 = 1 \\
1 \lor 1 = 1
\end{array}
\]

\[
\begin{array}{c}
\land, \lor \quad \text{"monotone"} \\
\text{increasing}
\end{array}
\]

\[
\begin{array}{c}
\land, \lor \quad \text{"monotone"} \\
\text{increasing}
\end{array}
\]

**MON:** input: a mon Bool. form. \( \varphi(x_1, \ldots, x_n) \)

Q: How many \( x \in \{0,1\}^n \) sat. \( \varphi(x) = 1 \)?

**Decision:** trivial \( \varphi(1') = 1 \)
Fact: \( \#\text{MON} \) is \( \#P \)-complete.

Idea: reduce from \( \#\text{CNF} \) to show:

- given \( \phi \) a 3CNF formula \( \#\text{3CNF} \in \#\text{P} \)
cooking up 2 different formulas, call \( \#\text{MON} \) on each,
subtract results: gives sol to \( \#\text{3CNF} \) on \( \phi \).

Next up: PERMANENT (PER)

Let \( M \) be an \( n \times n \) matrix.

\[
\text{Def } \text{PER}(M) = \sum_{\pi \in S_n} \prod_{i=1}^n x_{i, \pi(i)}
\]

i.e. \( \pi \) a permutation of \([n]\).

\[
M = \begin{pmatrix}
X_{11} & X_{12} \\
X_{21} & X_{22}
\end{pmatrix}
\]

\[
\text{PER} = X_{11}X_{22} + X_{12}X_{21}
\]

\( M \) is 3x3: 6 summands:

\[
X_{11}X_{22}X_{33} + \ldots + X_{13}X_{21}X_{32}
\]

\( \text{PER}(M) \) is a poly of deg \( n \), with \( n! \) monom.

\[
\left( 3x^2 + 4z \right. \text{ not mult.} \quad 4xyz + 2yzw \text{ mult. lin} \right)
\]

\( \text{PER}(M) \): monom. corr. to all ways to take exactly
I var from each row/col.

\[
\pi(3) = 1 \\
\pi(1) = 2
\]

**PER**: studied for 100+ years.

Q: How to compute \( \text{PER}(M) \) eff.?

\[
\rightarrow \text{poly}(\gamma) \cdot n! \quad \text{(by def.)}
\]

\[
\rightarrow \text{poly}(\gamma) \cdot 2^n \quad \text{"Ryser's formula" (incl/excl.)}
\]

**Contrast w/ DETERMINANT:** \( \text{(DET)} \)

\[
M_{n \times n} \ x_{ij}
\]

\[
\text{DET}(M) := \sum_{\pi \in S_n} \text{sign}(\pi) \cdot x_{1, \pi(1)} \cdot x_{2, \pi(2)} \cdotsx_{n, \pi(n)}
\]

i.e. \( \pi \) a perm. of \([n]\)

\[
\text{sign}(\pi) = \begin{cases} 
  +1 & \text{if } \pi \text{ is an even perm.} \\
  -1 & \text{if } \pi \text{ is odd}
\end{cases}
\]

A perm. \( \pi \) can either be written as composes of an even # of transpos. or an odd # of transpos.
\[\text{transpos:} \quad i \rightarrow j \quad j \rightarrow i \quad k \rightarrow k \quad \text{and} \quad k \notin \{i, j\}\]

\[
n = 5 \quad \pi(1) = 1 \quad \pi(3) = 3 \quad \pi(5) = 5 \]
\[
\pi(2) = 4 \quad \pi(4) = 2
\]

**Fact:** \(\text{DET} \in \text{FP}^p; \) poly-time computable

(row/column operation - Gauss. elim. - to convert to upper triangular_mtx:)

\[M \xrightarrow{GE} \begin{bmatrix} 4 & 17 \\ -2 & \end{bmatrix} = M'\]

\(\text{perm (diag.)}\)

**Thm (Valiant):** \(\text{PER} \) is \(\#P\)-complete.

(even \(0/1\) \(\text{PER} \) is \(\#P\)-complete).

\(\text{PER on } M \text{ where } x_{ij} \in \{0, 1\}\)

Let's see why \(0/1\) \(\text{PER} \) is in \(\#P\)?

\(0/1\) \(\text{PER} \) is \(\equiv\) to \(\#P_M\)

Perfect Matchings (in bip. graphs)
A undir. \( G = (V, E) \) is \underline{bipartite} if
\[ V = X \cup Y \quad \text{and} \quad \forall e \in E \ \text{is} \ e = (x, y) \]
\( X \) and \( Y \) are disjoint union.

Matching: set of edges s.t. no \( u \in X \) touches 2 edges.

If \( |X| = |Y| \),
a perfect matching is a matching of \( |X| \) edges.

Claim: \(
\#PM = 0 \text{ or } 1 \text{ PER. (So PERE#P)}
\)

Proof: \( G = (X \cup Y, E) \): set \( M \) to have \( x_{ij} = 1 \)
iff \( (i, j) \) is in \( E \)
0 otherwise \( x_{ij} = 0 \)
(adjacency matrix of \( G \))

Perm \( \pi \) has \( x_{i, \pi(i)} \ldots x_{n, \pi(n)} = 1 \) iff each \( x_{i, \pi(i)} = 1 \) iff
\( \pi \) corr. to a PM in \( G \).
So \( \text{PER} = \sum_{\pi \in S_n} x_1, \pi(1) \cdot x_2, \pi(2) \cdots x_n, \pi(n) \)
equals (\text{# of possible PM that are actually present in } G.)

**Random Self-Reducibility of PER**

RSR

\( \Rightarrow \text{conn. between worst-case and avg-case hardness.} \)

"hard": hard in worst-case sense: no eff alg. solves ALL instances correctly.

Q: could it be true that for some hard problem, there's a poly-time alg. that's right on 99.99% of all instances? Yes!

Ex:

\( \text{CUBE-ROOT-CLIQUE} = \{ G : G \text{ is an } n \text{-node graph containing a clique of size } n^{\frac{1}{3}} \}. \) \text{ Solw NPC.}

\(2^{\Omega(n^2)}\) instances: \( n \) nodes, \( \Omega(n^2)\) poss. edges, \( \Omega(n^2)\) graphs on \( n \) nodes.

A random input to this: a uniform random \( G \) from \( \Omega\), each edge \((i,j)\) present w.p. \( \frac{1}{3} \).
\[ \Pr \left[ \text{such a } G \text{ has an } n^{13} \text{-size clique} \right] \leq \frac{1}{2^{\Theta(n^{2/3})}}. \]

Fix \( S \subseteq [\ell] \), \( 15l = n^{13} \). There are \( \binom{n}{2} \) possible edges in \( S \), \( \Pr[\text{all those edges are present in } G] = \frac{1}{2^{\Theta(n^{13})}} \).

There are only \( \binom{n^{13}}{\ell^{13}} < n^{\ell^{13}} = 2^{\Theta(n^{4/3} \cdot \log n)} \) many poss. for \( S \).

So (u.b.i.) \( \Pr[G \text{ any } n^{13}-\text{clique exists in } G] \leq \frac{1}{2^{\Theta(n^{4/3})}} \cdot 2^{\Theta(n^{4/3} \cdot \log n)} = \frac{1}{2^{\Theta(n^{1/3})}} \).

---

Led to ask: (in worst-case sense)

are there "hard" problems for which we can be confident they're average-case hard (i.e., problem, assuming worst-case hard, is provably avg-case hard)? YES; PER.
PER is RSR

"F is RSR": if you can eff. compute F correctly on random inputs (w.h.p), then you can eff. compute F corr. on any input (w.h.p).

(idea: combine outcomes of F on several rand. points (each indiv. random, but collectively correlated) ? lets you compute F(x) for the chosen x)

Toy ex: Sps \( g: \{1, 2, \ldots, N\} \rightarrow \mathbb{R} \)

Sps \( g \) have oracle for \( g \).

Sps \( g \) is a "noisy linear function":

\[ f(x) = ax + b \rightarrow \text{real lin. fn.} \]

\[ g(x) = f(x) \text{ for } 0.99 \text{ many } x \in \{1, \ldots, N\} \]

Don't know \( a, b \), which \( x \) have \( g(x) = f(x) \).

Contrap: if worst-case hard, also worst-case hard

Goal: find \( f(z) \) \( z \) a partic. value in \( \{N\} \)

given oracle for \( g \). (your enemy)

Just calling \( g \) on \( z \): bad idea (could have \( a \in \) s.t. \( g(z) \neq f(z) \).)
Here's how:

1. Pick two independent \( i, j \in [N] \). (unit)
2. Call \( g(i), g(j) \)
3. Assume \( g(i) = a_i + b \) \( \quad \{ \text{true w.p.} \geq 98\% \} \)
   \[ g(j) = a_j + b \]
4. Solve for \( a, b \) (get right values w.p. \( \geq 98\% \))
5. Output \( a x + b \).

Works.

Worked b/c \( a x + b \) has "algebraic structure". PER has alg. structure too!

\( L \) (linear)
\( \Rightarrow (\text{low deg. poly}) \)
\( \Rightarrow \text{We'll show RSR.} \)

It'll be clearer to work \( \mod p \), \( p \) prime:

\( \mathbb{Z}_p = \{0, 1, \ldots, p-1\} \) do \text{with} \( \mod p \)

\( +, \times, -, \div \mod \)

\( \Rightarrow \text{PER mod } p \)

**MODPERM**: 

Input: \((M, p)\) \( M \) is \( n \times n \) mtx \( \text{with} \)
entries in \( \mathbb{Z}_p \), \( p \) prime

Output: value of \( \text{PER}(M) \mod p \).