Today

- Hardness of Uniquely satisfiable instances of SAT.
- Counting problems: $P \neq \overline{P}$.

Unique satisfiability

Motivation: Hard functions in cryptography.
Diffie-Hellman motivation for cryptography:
The map $(\phi, a) \mapsto \phi$, where $a$ satisfies $\phi$ is easy to compute but hard to invert.
So maybe similarly the map $(p, q) \mapsto p \cdot q$ is also easy to compute but hard to invert.
Can now start building cryptographic primitives based on this assumption.

Issues

Many leaps of faith:
- Specific problem has changed.
- The inputs have to be generated randomly.
- They have to have known “satisfiability”.
- etc. etc.

Initial big worry: The map $(\phi, a) \mapsto \phi$ loses information, while $(p, q) \mapsto p \cdot q$ does not. And NP-hardness requires “loss of information”.
Worry goes away, if we know $\phi$ has only one satisfying assignment. But then is problem as hard?

Formalizing the problem

Promise Problems: Generalize languages $L$.
$\Pi = (\Pi_{YES}, \Pi_{NO})$, $\Pi_{YES}, \Pi_{NO} \subseteq \{0, 1\}^*$, $\Pi_{YES} \cap \Pi_{NO} = \emptyset$.
Algorithm $A$ solves problem $\Pi$, if:
(Completeness): $x \in \Pi_{YES} \Rightarrow A(x)$ accepts.
(Soundness): $x \in \Pi_{NO} \Rightarrow A(x)$ rejects.
(Can extend to probabilistic algorithms naturally.)

Unique SAT: $USAT = (USAT_{YES}, USAT_{NO})$:
$\Pi_{YES} = \{\phi|\phi$ has exactly one sat. assgmt.\}$.
$\Pi_{NO} = \{\phi|\phi$ has no sat. assgmnts.\}$.
Formal question: Is $USAT \in P$? (Does there
exist a polytime algorithm \( A \) solving \( \text{USAT} \)?)

\[ \text{Valiant-Vazirani theorem} \]

Theorem: \( \text{USAT} \in P \) implies \( \text{NP} = \text{RP} \).

Proved via the following lemma.

Lemma: There exists a randomized reduction from \( \text{SAT} \) to \( \text{USAT} \).

\[ \phi \mapsto \psi \text{ such that } \phi \not\in \text{SAT} \text{ implies } \psi \in \text{USAT}_{\text{NO}}. \phi \in \text{SAT} \text{ implies } \psi \in \text{USAT}_{\text{YES}} \text{ with probability } 1/\text{poly}(n). \]

Again: Question stated without randomness, but answer mentions it! (Can also mention answer without randomness: \( \text{NP} \subseteq \text{P}/\text{poly} \) or \( \text{PH} \) collapses etc.)

\[ \text{Proof Idea} \]

\[ \psi \] will have as its clauses, all clauses of \( \phi \) and some more. \( \psi(x) = \phi(x) \land \rho(x) \).

So hopefully, will reduce \# sat. assignts to one.

Furthermore, can put any polynomial time decidable constraint \( \rho(x) \) (Since every computation can be transformed into \( \text{SAT} \). Exercise coming up.)

So what is \( \rho(x) \) going to be?

\[ \text{Proof Idea} \]

Suppose we know there exist \( M \) sat. assignts to \( \phi \).

Will pick a random function \( h: \{0, 1\}^n \rightarrow \{0, \ldots, M - 1\} \).

Hopefully this distinguished satisfying assignments, and we can let \( \rho(x) \) be the condition \( h(x) = 0 \).

Calculations imply this works out with constant probability.
**Caveats in the solution**

- How to do this reduction in polytime? Not enough time to represent $h$!
- Don’t know $M$!

Amendments:

- Will pick pairwise independent hash function.
- Will guess $M$ approximately (to within a factor of 2).

Things will work out!

**Pairwise independent hash families**

Defn: $H \subseteq \{f : \{0,1\}^n \rightarrow \{0,1\}^m\}$ is pairwise independent family if for all $a \neq b \in \{0,1\}^n$ and $c, d \in \{0,1\}^m$

$$\Pr_{h \in H}[h(a) = c \text{ AND } h(b) = d] = (1/2^m)^2.$$  

$H$ is nice if $h \in H$ can be efficiently sampled and efficiently computed.

Example: Pick $A \in \{0,1\}^{m \times n}$ and $b \in \{0,1\}^m$ at random. Let $h_{A,b}(x) = Ax + b$. Then $H = \{h_{A,b}\}_{A,b}$ is a nice, pairwise independent family.

Proof: Exercise.

---

**Randomized reduction from SAT to USAT**

Given $\phi$:

- Pick $m \in \{2, \ldots, n+1\}$ at random (and hope that # satisfying assignments is between $2^{m-2}$ and $2^{m-1}$.)
- Pick $h$ at random from nice p.w.i. family $H$.
- Let $\psi(x) = \phi(x) \land (h(x) = 0)$.
- Output $\psi$.

**Analysis**

Let $S = \{x | \phi(x)\}$.

Hope: $2^{m-2} \leq |S| \leq 2^{m-1}$.

Claim: $\Pr_m[\text{Hope is realized}] \geq 1/n$.

Proof: Claim is true for some $m \in \{2, \ldots, n+1\}$. Prob. we pick that $m$ is $1/n$. 
Analysis (contd.)

Claim: \( \Pr_h[ \text{Exactly one } x \in S \text{ maps to 0 — Hope}] \geq 1/8. \)

Define \( G_x \): Event that \( x \) maps to 0 and no other \( y \in S \) maps to 0.

Prob. we wish to lower bound is (conditioned on Hope):

\[
\Pr_h[\bigcup_{x \in S} G_x] = \sum_x \Pr_h[G_x]
\]

(since \( G_x \)'s are mutually exclusive).

\( \Pr_h[h(x) = 0] = 1/2^m. \)

\( \Pr_h[h(x) = 0 \text{ and } h(y) = 0] = 1/4^m. \)

\( \Pr_h[h(x) = 0 \text{ and } \exists y \in S - \{x\}, s.t.h(y) = 0] \leq |S|/4^m. \)

Concluding the analysis

With probability \( 1/8n \) reduction produces \( \psi \) with exactly one satisfying assignment. If you can decide satisfiability in such cases then can decide satisfiability probabilistically in all cases.

New topic: Counting classes

Given NP machine, how many accepting paths does it have?

\( \#P \) is class of functions \( f : \{0,1\}^* \rightarrow \mathbb{Z}^{\geq 0} \) such that there exists a machine \( M(\cdot,\cdot) \) running in polytime in first input such that for every \( x, f(x) = \{y|M(x,y)\}. \)

\( \text{P}'\#P \) is class of languages decidable with oracle access to \( \#P \) functions.

Very important class: Has usual complete functions \( \#\text{SAT}, \# \text{ Hamiltonian cycles}, \) and also \( \# \) cycles in digraph.

Most novel: \( \# \) matchings in bipartite graph; also permanent of non-negative matrix.
How powerful is $\text{P}^\#\text{P}$?

- $\text{P}^\#\text{P} \subseteq \text{PSPACE}$.
- $\text{BPP} \subseteq \text{P}^\#\text{P}$.
- $\text{NP} \subseteq \text{P}^\#\text{P}$.
- $\text{co-NP} \subseteq \text{P}^\#\text{P}$.

What about $\Sigma^\#_2$? Open till Toda’s theorem.

Thm [Toda]: $\text{PH} \subseteq \text{P}^\#\text{P}$.

No known reasons to believe $\text{P}^\#\text{P} \neq \text{PSPACE}$. (Can you prove anything?)

Operators on complexity classes

An “operator” maps a complexity class into a related one.

A short list: $\exists, \forall, \text{BP}, \bigoplus$.

$\mathcal{C} \mapsto \mathcal{O} \cdot \mathcal{C}$.

$\cdot \mathcal{C}$ is simple: complements of languages in $\mathcal{C}$.

In all other cases, think of machines in $\mathcal{C}$ as two input machines and operator shows how to quantify over second input.

- $\exists$, does there exist second input?
- $\forall$, for every second input.
- $\bigoplus$: for odd # of second inputs,

- $\text{BP}$, for at least $c(n)$ fraction of second input if $x \in L$ versus at most $s(n)$ if $x \notin L$, where $c(n) - s(n) \geq 1/\text{poly}(n)$.

(Sample) definition:

$L \in \bigoplus \mathcal{C}$ if there exists a machine $M(\cdot, \cdot) \in \mathcal{C}$ (whose second input should be polynomial-length in the first input) such that $w \in L \iff |\{x|M(w, x)\}|$ is odd.

Example operations:

- $\exists \cdot \text{P} = \text{NP}$.
- $\forall \cdot \text{P} = \text{co-NP}$.
- $\exists \cdot \Sigma^P_3 = \Sigma^P_3$.
- $\forall \cdot \Sigma^P_3 = \Pi^P_4$.
- $\text{BP} \cdot \text{P} = \text{BPP}$.

Proof of Toda’s Theorem

Main ingredients:

- Operators on complexity classes.
- Closure properties.
- Randomness
- Algebra
- Blah Blah Blah
Toda's theorem steps

1. $\Sigma_i^P \subseteq \text{BP} \cdot \bigoplus P_{k-1}^P$.
   $\Pi_i^P \subseteq \text{BP} \cdot \bigoplus \Pi_{k-1}^P$.
   (Extends Valiant-Vazirani.)

2. $\text{BP} \cdot \bigoplus P$ amplifies error.
   (Subtle.)

3. $\bigoplus \cdot \text{BP} \cdot \bigoplus P \subseteq \text{BP} \cdot \bigoplus \cdot \bigoplus P \subseteq \text{BP} \cdot \bigoplus P$.
   (Surprising, but straightforward.)

4. $\text{BP} \cdot \text{BP} \cdot \bigoplus P \subseteq \text{BP} \cdot \bigoplus P$.
   (Not surprising. Straightforward.)

After all the above:

Theorem: $\text{PH} \subseteq \text{BP} \cdot \bigoplus P$.

Completely separate theorem:

Theorem: $\text{BP} \cdot \bigoplus P \subseteq \text{P}^\#P$.

Details tomorrow.