6.841/18.405J: Advanced Complexity	Monday, March 10th, 2003
Lecture 9	
Instructor: Madhu Sudan	*Scribe: Fu Liu*

- Counting Problems
- Complexity of Unique Satisfiability
- #P

1 Counting Computations

Considering counting the number of accepting paths in a nondeterministic TM motivates the definition of the class #P.

Definition: A function $f : \{0,1\}^* \to Z^{\geq 0}$ is in #P, if there exists a polynomial time nondeterministic TM M such that

$$f(x) = |\{y | M(x,y) \text{accepts}\}|.$$

2 Complexity of Unique Satisfiability

2.1 Motivation

Motivated by NP completeness problems, Diffie and Hellman '78 introduced "Public Key Crypto. System'. Consider the following examples:

1. Given a formula ϕ and one of its satisfying assignments a, delete a. That is:

$$(\phi, a) \to \phi.$$

2. Given two primes p and q, multiply them to get $p \cdot q$. That is:

$$(p, q) \to p \cdot q.$$

Note that the above two maps are both easy to be calculated. However, if we consider the inverse of these two maps, the first one is to find a satisfying assignment of ϕ, the second is to factor a number into two primes. Both of the two inverse are hard to compute. Then how does the hardness of factoring relate to that of finding a satisfying assignment? Several similar questions were being asked at the time:

1. How did "changing the problem" (SAT $\to$ Factoring) affect its hardness?
2. What if (ϕ, a) were chosen at random?
3. Note that the first mapping is an information-losing map, which loses information of a, and the second one is information-preserving since given $p \cdot q$, there's only one way to factor it if given $p \leq q$. Then what if ϕ has only one satisfying assignment a.? Actually in this case, the first mapping becomes information-preserving.

2.2 Formalizing the Problem

By the discussion of last section, one may think of this problem: "Given a formula ϕ that has only one satisfying assignment a, compute the assignment a." This leads us to a new language USAT. Before give out the formal definition of USAT, we introduce Promise Problems first.

Definition: A promise problem Π consists of two sets, Π_{YES} and Π_{NO}, both of which are subsets of $\{0,1\}^*$ and whose intersection is the empty set, i.e.

- $\Pi = (\Pi_{YES}, \Pi_{N)})$
- $\Pi_{YES}, \Pi_{NO} \subseteq \{0,1\}^*$
- $\Pi_{YES} \cap \Pi_{NO} = \emptyset$.

Given a promise problem, our goal is to find a algorithm A such that given x, $x \in \Pi_{YES} \Rightarrow A(x) = 1$ and $x \in \Pi_{NO} \Rightarrow A(x) = 0$.

Now we define USAT as a promise problem:

- USAT= $(\Pi_{YES}, \Pi_{N)})$
- $\Pi_{YES} = \{\phi | \phi$ has exactly one satisfying assignment.$\}$
- $\Pi_{NO} = \{\phi | \phi$ has no satisfying assignments.$\}$

2.3 Hardness of USAT

Let's look back the inverse of the mapping $(\phi, a) \rightarrow \phi$. If ϕ is given from $USAT_{YES}$, since a is unique, the map is information-preserving. So the inverse exists. But we don't know if it is easy to find the answer. Valiant and Vaziram showed that if we can find an algorithm that compute the inverse mapping efficiently, thne NP = RP:

Theorem: If USAT $\in$ P, then NP = RP.

This theorem is proved by the following lemma:

Lemma: USAT is NP-hard under random reduction from SAT.

We need reduction: $\phi \in SAT? \longrightarrow \psi \in USAT?$ such that:

- $\phi \in SAT \longrightarrow \psi$ has one satisfying assignment with probability $\frac{1}{poly(n)}$.
- $\phi \notin SAT \longrightarrow \psi \notin USAT$ with probability 1.

2.4 Reduction Idea

We assume that ϕ has M satisfying assignments, consider reduction:

$$\phi(x) \longrightarrow \psi(x) = \phi(x) \wedge p(x).$$

Pick a random function $h : \{0,1\}^n \rightarrow \{1, 2, \ldots, 4M\}$, where n is the number of variables of ϕ. Let $p(x) =$ "$h(x) = 1$, then we claim that

$$Pr[(\exists! x : \phi(x) = 1 \wedge h(x) = 1) \wedge (\forall y : \phi(y) \neq 1 \vee h(y) \neq 1)] \geq 1/8.$$

We will give the proof of this claim later. Now let's look at some problems with the implementing ideas:

1. h, p are not succinct.
2. M is not known.

We have amendments to these problems:

1. Pick h from a pairwise independent family of hash functions, $\mathcal{H}$, randomly.

   Definition: $\mathcal{H} = \{h : \{0,1\}^n \to \{1,2,\ldots,\alpha M\}\}$ is a pairwise independent family of hash functions if for $\forall x \neq y \in \{0,1\}^n, \forall a,b \in \{0,1\}^m, Pr_{h\in\mathcal{H}}[h(x) = a \wedge h(y) = b] = \frac{1}{(\alpha M)^2}$.

   Let $\mathcal{H} = \{h_{c,d} | c,d \in \{1,2,\ldots,\alpha M\}\}$, where $h_{c,d}(x) = (c \cdot x + d) \pmod{\alpha M}$. It's clear that $\mathcal{H}$ is a pairwise independent family of hash functions.

   Now, let's look back our reduction. Pick h from the above defined $\mathcal{H}$.

   Let $S \subseteq \{0,1\}^n$ be the set of satisfying assignments of ϕ, $|S| = M$. Then,

$$Pr_{h\in\mathcal{H}}[\exists x,y \in S : x \neq y, h(x) = h(y) = 1] \leq \sum_{x,y} Pr_{h\in\mathcal{H}}[h(x) = h(y) = 1] \leq |S|^2 \cdot \frac{1}{(\alpha M)^2} = \frac{1}{\alpha^2}.$$

$$Pr_{h\in\mathcal{H}}[\exists x \in S : h(x) = 1] \geq \sum_{x} Pr_{h\in\mathcal{H}}[h(x) = 1] - \sum_{x,y} Pr_{h\in\mathcal{H}}[h(x) = h(y) = 1] = \frac{1}{\alpha} - \frac{1}{\alpha^2}.$$

   Therefore,

$$\begin{aligned} & Pr_{h\in\mathcal{H}}[\exists! x \in S : h(x) = 1] \\ \geq\ & Pr_{h\in\mathcal{H}}[\exists x \in S : h(x) = 1] - Pr_{h\in\mathcal{H}}[\exists x,y \in S : x \neq y, h(x) = h(y) = 1] \\ \geq\ & (\frac{1}{\alpha} - \frac{1}{\alpha^2}) - \frac{1}{\alpha^2} \\ =\ & \frac{1}{\alpha} - \frac{2}{\alpha^2}. \end{aligned}$$

   Our goal is that $Pr_{h\in\mathcal{H}}[\exists! x \in S : h(x) = 1] \geq$ some constant. Then it satisfies iff $\alpha > 2$. In particular, when pick $\alpha = 4$, we proved our claim that

$$Pr[(\exists! x : \phi(x) = 1 \wedge h(x) = 1) \wedge (\forall y : \phi(y) \neq 1 \vee h(y) \neq 1)] = Pr_{h\in\mathcal{H}}[\exists! x \in S : h(x) = 1] \geq 1/8.$$

2. It turns out that if we estimate M within a factor of 2, the probability calculation still works. Therefore, we only need compute M' such that $M \leq M' \leq 2M$. Thus, we randomly pick M' from $\{1,2,4,\ldots,2^n\}$.

2.5 Summarizing the Reduction

- Given ϕ
- Pick $M' \in_R \{1,2,4,\ldots,2^n\}$
- Let $\mathcal{H}$ be a pairwise independent family of hash functions mapping $\{0,1\}^n \to \{1,2,\ldots,4M'-1\}$
- Pick $h \in \mathcal{H}$ at random, let $\psi(x) = \phi(x) \wedge (h(x) = 1)$.

3 Toda's Theorem

Theorem: PH $\subseteq P^{\#P}$.