Computational Complexity. Lecture 17

Randomized Reductions and Valiant-Vazirani.

Alexandra Kolla
Today

- Polynomial identity testing
- Randomized Reductions
- Valiant-Vazirani
Polynomial identity testing

- Given a polynomial with integer coefficients in implicit form, decide if it is identically zero.
- No known poly-time algorithm.
- We describe a poly-time probabilistic algorithm.
- Polynomial given in form of algebraic circuit.
Polynomial identity testing

- Like Boolean circuits but AND, OR and NOT replaced by +, -, x.
- Formally, a n-variable algebraic circuit is a DAG with the sources labeled by the variables $x_1, ..., x_n$ and each non-source node having in-degree 2, labeled by an operator from the set {+, -, x}.
- Single sink in the graph which is the output.
- This algebraic circuit describes polynomial from $\mathbb{Z}^n \rightarrow \mathbb{Z}$. 
Polynomial identity testing

- Define the class ZEROP = set of algebraic circuits that compute the identically zero polynomial.

- Polynomial identity testing = deciding membership in ZEROP, since we can reduce the problem of deciding whether two circuits C, C' compute the same polynomial to ZEROP by constructing circuit D(x_1, ..., x_n) = C(x_1, ..., x_n) - C(x_1, ..., x_n)'
Polynomial identity testing

- ZEROP problem non trivial cause compact circuits can represent polynomials with large number of terms.
- E.g. circuit of size $2n$ can compute $\Pi_i (1 + x_i)$ which has $2^n$ terms.
- There is a simple randomized poly time algorithm for testing membership in ZEROP.
Zchwartz-Zippel lemma

• **Lemma.** If $p(x_1, \ldots, x_n)$ is an n-variate non-zero polynomial of degree $d$ over a finite field $F$, then $p$ has at most $d F^{n-1}$ roots. Equivalently, $\Pr[p(a_1, \ldots, a_n) = 0] \leq \frac{d}{F}$. 
Polynomial identity testing

- coRP algorithm for ZEROP:
  - Choose a field $F$ of size at least $3d$.
  - Choose random $a_1, \ldots , a_n \in F^n$.
  - Accept if $p_1(a_1, \ldots , a_n) = p_2(a_1, \ldots , a_n)$
  - Always accept if polynomials are equivalent.
  - (Ex). If the two polynomials not equivalent, reject with probability at least $2/3$. 
Randomized reductions

- Useful to define randomized reductions between complexity classes.

- **Definition.** Language $B$ reduces to language $C$ under a randomized polynomial time reduction, denoted $B \leq_r C$, if there is a probabilistic polynomial time algorithm $A$, such that for every $x \in \{0,1\}^*$, $\Pr[C(A(x)) = B(x)] \geq \frac{2}{3}$.
Randomized reductions

- Not transitive definition.

- Useful in the sense that if $\mathcal{C} \in \text{BPP}$ and $\mathcal{B} \leq_r \mathcal{C}$, then $\mathcal{B} \in \text{BPP}$.

- We could have defined \text{NP} with randomized reductions, we would get different class.
Next, we show the hardness of Unique-SAT.

Suppose there is an algorithm for the satisfiability problem that always finds a satisfying assignment for formulae that have exactly one satisfying assignment and behaves arbitrarily on other instances.

Then we can get an RP algorithm for 3SAT, thus NP=RP.
Valiant-Vazirani

- Proof by presenting randomized reduction.
- Given in input CNF formula $\phi$ produces output a polynomial number of CNF formulae $\psi_1, \ldots, \psi_n$. If $\phi$ is satisfiable then w.h.p. at least one of the $\psi_i$ are satisfiable. Otherwise, w.p 1 all of them are unsatisfiable.
- Describe main idea.
Pairwise independent hash functions

- **Definition.** Let $H$ be a family of functions of the form $h : \{0,1\}^n \rightarrow \{0,1\}^m$. We say that $H$ is a family of pair-wise independent hash functions if for every two different inputs $x, y \in \{0,1\}^n$ and for every two possible outputs $a, b \in \{0,1\}^m$ we have

$$\Pr_{h \in H} [h(x) = a \text{ and } h(y) = b] = \frac{1}{2^{2m}}$$
Pairwise independent hash functions

- Means that for every disjoint $x, y$, when we pick $h$ at random from $H$ then the random variables $h(x)$ and $h(y)$ are independent and uniformly distributed.
- In particular, for every $x \neq y$ and for every $a, b$, we have

$$\Pr_{\begin{array}{c} h \in H \\ h \notin \{a, b\} \end{array}} [h(x) = a \mid h(y) = b] = \Pr_{\begin{array}{c} h \in H \\ h \notin \{a, b\} \end{array}} [h(x) = a]$$
Construction of family of pairwise independent hash functions

- For m vectors $a_1, \ldots, a_m \in \{0,1\}^n$ and m bits $b_1, \ldots, b_m$ define
  $h_{a_1,\ldots,a_m,b_1,\ldots,b_m} : \{0,1\}^n \rightarrow \{0,1\}^m$
  as $h_{a,b} = (a_1 \cdot x + b_1, \ldots, a_m \cdot x + b_m)$

And let $H_{\text{AFF}}$ be the family of functions defined this way. Then $H_{\text{AFF}}$ is a family of pairwise independent hash functions (ex).
The proof

- **Lemma.** Let $\mathcal{T} \subseteq \{0,1\}^n$ be a set such that $2^k \leq |\mathcal{T}| \leq 2^{k+1}$ and let $\mathcal{H}$ be a family of pairwise independent hash functions of the form $h: \{0,1\}^n \rightarrow \{0,1\}^{k+2}$. Then, if we pick $h$ at random from $\mathcal{H}$, there is a constant probability that there is a unique element $x \in \mathcal{T}$ such that $h(x) = 0$. Precisely,

$$\Pr_{h \in \mathcal{H}} \left[ |\{x \in \mathcal{T}: h(x) = 0\}| = 1 \right] \geq \frac{1}{8}$$
The proof

- **Lemma.** There is a probabilistic polynomial time algorithm that, on input a CNF formula $\phi$ and an integer $k$ outputs a formula $\psi$ such that
  - If $\phi$ is unsatisfiable then is $\psi$ unsatisfiable
  - If $\phi$ has at least $2^k$ and less than $2^{k+1}$ satisfying assignments then there is a probability at least $1/8$ that the formula $\psi$ has exactly one satisfying assignment.
Valiant-Vazirani

- **Theorem.** Suppose there is a polynomial time algorithm that on input a CNF formula having exactly one satisfying assignment, finds this assignment. Then NP=RP.