CS221: Computational Complexity Prof. Salil Vadhan

Lecture 29: $P^{\#P} \subseteq IP$

12/4 *Scribe: Dave Fetterman*

Contents

1 Announcements

- Class Friday in MD-125.
- Relevant Readings for this lecture: PCMI lecture notes (Handout), Papadimitriou 12.2.

2 The unsound protocol

Last time, we started proving the following theorem.

Theorem 1 $\mathbf{P}^{\#\mathbf{P}} \subseteq \mathbf{IP}$

Recall we were giving an interactive proof system for the following problem: E#SAT = $\{(\phi, k) :$ ϕ has exactly k satisfying assignments$\}$.

Last time we saw a protocol exhibiting completeness but not soundness. Here is the old protocol, which we will subsequently fix.

Observation: ϕ has exactly k satisfying assignments iff there exist k_0, k_1 such that:

- $k_0 + k_1 = k$
- $\phi_0(x_1, x_2, \cdots x_n) \equiv \phi(0, x_2, \ldots, x_n)$ has exactly k_0 satisfying assignments.
- $\phi_1(x_1, x_2, \cdots x_n) \equiv \phi(1, x_2, \ldots, x_n)$ has exactly k_1 satisfying assignments.

First attempt at an IP for E#SAT:

- ϕ, k given to both *Prover* and *Verifier*

Prover:

- Send k_0 and k_1 to *Verifier*

Verifier:

- Choose $b \in \{0, 1\}$ at random.
- Send b to *Prover*.

As in the last lecture, the prover's strategy is to recursively show that $(\phi_b, k_b) \in$ E#SAT, and at the end of n iterations, the verifier can be sure that the trivial formulae will be *true* or *false*.

This clearly works correctly if $(\phi, k) \in$ E#SAT, but we see that the soundness condition can be violated with probability exponentially near to 1, causing the verifier to accept a false input.

How do we avoid this problem?

- We have the latter two conditions from above, and on false input, we can only guarantee that one is violated. Verifier has no way of deciding this efficiently.
- *Solution:* Allow the verifier choices is a larger set (larger than $b \in \{0, 1\}$).
- *Idea:* Allow variables to take values in a large set F and guarantee that most values reflect inconsistency, rather than just half of them.

3 Arithmetization

We will extend $\phi : \{0, 1\}^n \to \{0, 1\}$ to $\tilde{\phi} : \mathbb{F}^n \to \mathbb{F}$, where $\mathbb{F}$ is a large finite field, so $\tilde{\phi}$ is a low-degree polynomial. The benefits of this will stem from the fact that distinct, low-degree polynmials agree in very few places. Note that we recently saw a very general way of extending boolean functions to low-degree polys, but that only ensures that $\tilde{\phi}$ can be computed in $\mathbf{PSPACE}^{\phi}$, whereas we need the verifier to be able to verify $\tilde{\phi}$ itself. Thus we must do this in a very particular way, exploiting the structure of SAT. What we will achieve is:

- $\tilde{\phi}|_{\{0,1\}^n} = \phi$.
- $d \equiv deg(\tilde{\phi}) \leq |\phi|$
- $\tilde{\phi}(x)$ can be evaluated in polynomial time given ϕ and $x \in \mathbb{F}^n$.

Now on to the recursive definition, $\phi \mapsto \tilde{\phi}$:

- $x_i \mapsto x_i$
- $\neg\phi \mapsto 1 - \tilde{\phi}$
- $\phi \wedge \psi \mapsto \tilde{\phi} \cdot \tilde{\psi}$

- $\phi \vee \psi \mapsto 1 - (1 - \tilde{\phi})(1 - \tilde{\psi})$ (by DeMorgan's).

We also choose $\mathbb{F} = Z_q$, where q is a prime such that $q > 2^d \geq 2^n$. Thus, P wants to prove to V that $\Sigma_{x_1 \in \{0,1\}} \Sigma_{x_2 \in \{0,1\}} \cdots \Sigma_{x_n \in \{0,1\}} \tilde{\phi}(x_1, x_2, \cdots, x_n) = k$.

The idea now is similar to the earlier case, but the domain of α in the following is $\mathbb{F}$ instead of $\{0,1\}$. Let $k_\alpha = \Sigma_{x_2 \in \{0,1\}} \Sigma_{x_3 \in \{0,1\}} \cdots \Sigma_{x_n \in \{0,1\}} \tilde{\phi}(\alpha, x_2, \cdots, x_n)$ be denoted by formula (*).

P :

- send $\{k_\alpha\}_{\alpha \in \mathbb{F}}$

V :

- Check $k_0 + k_1 = k$
- choose $\alpha \in \mathbb{F}$
- send α

The protocol continues with the recursive proof on $\tilde{\phi}(\alpha, x_2, \cdots, x_n)$ and k_α.

Some questions then arise: how does P send a huge list $\{k_\alpha\}$ to V? What have we gained in soundness by this modification? Observe first, (*) has degree $\leq d$ in α. This allows us to envision this protocol: P:

- Compute degree d polynomial $p_1(x) = \Sigma_{x_2 \in \{0,1\}} \Sigma_{x_3 \in \{0,1\}} \cdots \Sigma_{x_n \in \{0,1\}} \tilde{\phi}(x, x_2, \cdots, x_n)$
- send polynomial p_1*

V:

- Check $p_1(0) + p_1(1) = k$
- Choose $\alpha_1 \in \mathbb{F}$ randomly and send to P

P:

- Compute polynomial $p_2(x) = \Sigma_{x_3 \in \{0,1\}} \Sigma_{x_4 \in \{0,1\}} \cdots \Sigma_{x_n \in \{0,1\}} \tilde{\phi}(\alpha_1, x, x_3, \cdots, x_n)$
- send polynomial p_2

V:

- Check $p_2(0) + p_2(1) = p_1(\alpha_1)$
- Choose $\alpha_2 \in \mathbb{F}$ randomly and send to P

continuing for n steps... V:

- Check $p_n(0) + p_n(1) = p_{n-1}(\alpha_{n-1})$

- Choose $\alpha_n \in \mathbb{F}$ randomly
- Check $p_n(\alpha_n) = \tilde{\phi}(\alpha_1, \ldots, \alpha_n)$
- Accept if all of these checks pass, failing otherwise.

Claim 2 *This is an interactive proof for* E#SAT.

Proof:

- Efficiency: V is poly-time by inspection and our choice of arithmetization.
- Completeness: By construction.
- Soundness: Suppose ϕ does not have exactly k satisfying assignments. We'll show that $\forall P^*, \Pr[V accepts : (P^*, V)(\phi, k)] \leq \frac{nd}{|\mathbb{F}|} \leq \frac{nd}{2^d} \to 0$. Let $p_1^*, \ldots, p_n^*$ be polynomials sent by P^*. Let $p_1, \ldots, p_n$ be polynomials computed according to specified P-strategy. Assume $p_1(0) + p_1(1) = \#$ satisfying assignments to $\phi \neq k$. Then either $p_1^*(0) + p_1^*(1) \neq k$ or $p_1^* \neq p_1$. In the first case, V rejects immediately. Otherwise, w.p. $\geq 1 - \frac{d}{|\mathbb{F}|}, p_1^*(\alpha_1) \neq p_1(\alpha_1)$ by the Basic Theorem of Algebra (two distinct polynomials of degree d agree in at most d places).

  Suppose $p_1^*(\alpha_1) \neq p_1(\alpha_1) = p_2(0) + p_2(1)$. Then either

  - $p_2^*(0) + p_2^*(1) \neq p_2^*(\alpha_1) \Rightarrow$ V rejects immediately.
  - $p_2^* \neq p_2 \Rightarrow w.p. 1 - \frac{d}{|\mathbb{F}|}$ over $\alpha_2, p_2^*(\alpha_2) \neq p_2(\alpha_2)$
  - etc...

  So by union bound overall, $\Pr[V$ rejects or $p_n^*(\alpha_n) \neq p_n(\alpha_n)] \geq 1 - n\frac{d}{|\mathbb{F}|} \geq \frac{2}{3}$.

■

Now that we see interactive proofs are so much more powerful than classical proofs, containing all of $\mathbf{P}^{\#\mathbf{P}}$ (and hence the entire **PH**, by Toda), we'd like to fully characterize their power. This was done shortly thereafter:

Theorem 3 (Shamir) *:* **IP** = **PSPACE**

You will show on the problem set **IP** $\subseteq$ **PSPACE**. The other direction is based on the arithmetization ideas above, applied to QBF. See the PCMI notes for details. But how are we to interpret this? Think about **PSPACE**-complete problems like GO or CHESS. If we have Kasparov (an "optimal chess oracle"), he can efficiently convince you (a weak, poly-time verifier) that a move is optimal. Additionally, there emerge many nice philosophical interpretations. For example, it validates our everyday intuition that one can be taught more through interaction (as in a classroom) than just by reading.