Advanced Proof Systems - Course Material

Yosef Israel Goren

November 5, 2022

Lecture 1

1 Recap

1.1 P - Polynomial (Class)

$L \in \{0,1\}^*$ is also in P if there exists an efficient algorithm which decides it.

1.2 NP - Nondeterministic Polynomial

$L \in \{0,1\}^*$ is also in NP if there exists an efficient algorithm V and a polynomial p which follow:

1. Completeness: $\forall x \in L, \exists y : V(x,y) = 1 \wedge |y| < p(|x|)$
2. Soundness: $\forall x \notin L \forall y : V(x,y) \neq 1 \vee |y| \geq p(|x|)$

1.3 PPT - Probobalistic Polynomial Time

This is a class of algorithms which must run in time polynomial to the size of their input, but also - must be capable of randomization, or 'flipping coins'.

1.4 IP - Interactive Proof

A key difference between an Interactive Proof and a proof for an NP proof is that the latter necessarily requires the prover to provide the verifier with something he can use to prove the truth of the calim to others.

We denote $(P,V)(x)$ to be the output of V (verifier) after the interaction between P and V on the input x. Both P and V can be thought of as PPT algorithms or programs which are capable of communicating with one another.

These interactions are often described with an interaction diagram:

- P sends to V something

- V sends to P something else
- ...
- V accepts iff ...

Formal Definition: We say that $L \in IP$ if there exists a polynomial algorithm V, an unbounded algorithm P and some constant $c \in (0.5, 1]$ s.t.

1. Completeness: $\forall x \in L, Pr[(P, V)(x) = 1] > c$
2. Soundness: if $x \notin L, \forall P^* \in \mathbf{M}, Pr[(P^*, V)(x) = 1] < 1 - c$

Note: $\mathbf{M}$ denotes the set of turing machines.

2 Equivalence of IP separation constants

2.1 Iterative Runs

Given an IP protocol (P, V), let (P^k, V^k) be the protocol obtained by running (P, V) k times sequentially. V^k accepts iff in all iterations V accepted.

2.2 Lemma

if (P, V) is IP with perfect completeness then for every polynomial k, (P^k, V^k) is IP with perfect completeness and soundness error $2^{-\Omega(k)}$

Proof:

1. V^k is efficient (composition of polynomials).
2. Perfect Completeness - due to Perfect Completeness of the original protocol - each iteration is guaranteed to succeed thus the protocol always does.
3. Soundness: Let $x \in L, P^*$. We will show that $Pr[(P^*, V^k)(x) = 1] \leq 2^{-k}$. Denote by E_i the event that V^k accepts in the i'th iteration. Thus:

$$Pr[E_1 \wedge E_2 \wedge \dots E_k] = \prod_{i=1}^{k} Pr[E_i | E_1 \wedge E_2 \wedge \dots E_k]$$

Claim: $Pr[E_i | E_1 \wedge E_2 \wedge \dots E_k] \leq 0.5$.
Proof: Assume toward a contradiction that $Pr[E_i | E_1 \wedge E_2 \wedge \dots E_k] > 0.5$
We design a prover P^{**} that convinces V with up to > 0.5:
P^{**} emulates (P^*, V) for iterations $1...i-1$ until the event $E_1 \wedge E_2 \wedge \dots E_k$ happens and then runs (P^*, V) as the i'th iteration. Since the run of (P^*, V) for the i'th iteration only happens under the condition $Pr[E_i | E_1 \wedge E_2 \wedge \dots E_k]$ - the probability for (P^*, V) to happen on the i'th iteration is exactly $Pr[E_i | E_1 \wedge E_2 \wedge \dots E_k] > 0.5$ but this is also the

probability for $Pr[(P^{**}, V)(x) = 1]$. Contradiction.

Thus:

$$Pr[E_1 \wedge E_2 \wedge \dots E_k] = \prod_{i=1}^{k} Pr[E_i | E_1 \wedge E_2 \wedge \dots E_k] \leq \prod_{i=1}^{k} 0.5 = 2^{-k}$$

3 Graph Isomorphism & IP Example

3.1 Graph Isomorphism - Definition

The graphs $G_1 = (V, E_1), G_2 = (V, E_2)$ are isomorphic or $(G_1, G_2) \in GI$ if $\exists \pi : V \longrightarrow V$ s.t. $(u, v) \in E_1 \iff (\pi(u), \pi(v)) \in E_2$.
More simply - two graphs are isomorphic if they are identical up to a renaming of their vertices.

GNI is the set of pairs of graphs which are isomorphic.

- Claim (no proof): $NP \subseteq IP$.

- Claim (proof sketch): $GI \in IP$.
  By finding the permutation π it is easy to check the GI condition over a given (G_1, G_2) thus we have an NP relation, meaning $GI \in NP$.

- Claim (no proof): $GNI \in IP$.

Tutorial 1

4 IP and NP

Claim: $NP \subseteq IP$.
Proof: let $L \in NP$. There exists some NP relation R for L, with an efficient algorithm M_R which decides it.
Now define an IP protocol:

- Both P and V get x.
- If $x \in L$ P find y s.t. $(x, y) \in R$, and send it to V. Otherwise send ϵ.
- V checks if $(x, y) \in R$ by running M_R (known to be efficient) and accepts iff $M(x, y)$ accepts.

1. Completeness: If $x \in L$, such y must exist (NP definition) thus P will find it, and V will have $(x, y) \in R$ so M_R and V will accept.
2. Soundness: If $x \notin L$, there is no y which such that $(x, y) \in R$ so no matter what any P^* sends - $M_R(x, y)$ rejects and so V rejects too.

5 Similar Proof Systems

5.1 Arthur-Merlin - AM

- Both parties get some input x.
- Arthur sends Merlin some randomized α.
- Merlin sends back some β.
- Arthur accepets according to some PPT algorithm which is a function of x, α and β (usually denoted $A(x, \alpha, \beta)$).

5.2 Merlin-Arthur - MA

- Both parties get input x.
- Merlin sends β to Arthur.
- Arthur generates some random value α.
- Arthur accepts according to some PPT algorithm which is a function of x, α, β.

Theorem: $MA \subseteq AM$.
Proof: Let $L \in MA$. WLOG (without loss of generality), L has a MA protocol with perfect completeness (we will come back to this assumption later in the course).
Denote by $p(n)$ the length β ($|\beta| \leq p(n)$).

Using repetition we can get to any protocol with perfect completeness and a soundness error of $2^{-p(n)-1}$ (as seen in lecture).
Sketching the repeated protocol would look like:

- M' sends β to A'
- A' sends back a list of α values (as many as there are repetitions).
- A' decides wether to accept.

This is because there is not reason for the prover's proof (β) to change due to different sampling of α, so it is always the same and can be sent once. So the length of Merlin's message does not change in the repeating protocol.

Now consider the same M', A' protocol but where A' sends the aggregated α before M' sends β.

Claim: This new protocol is AM. Proof:

1. Completeness: $\forall x \in L, M'$ sends the same β without looking at α:

$$Pr[(M', A)(x) = 1] = Pr[A(x, \alpha, \beta) = 1] = 1$$

2. Soundness: Let $x \notin L$, fix M^*. Consider:

$$Pr[(M^*, A')(x) = 1] = Pr[\exists \beta \in \{0,1\}^p : A'(x, \alpha, \beta) = 1]$$

$$= \cup_{\beta \in \{0,1\}^p} Pr[A'(x, \alpha, \beta) = 1] \leq_{UB} 2^p 2^{-(p+1)} = \frac{1}{2}$$

Note: UB denotes Union Bound.

Lecture 2

6 Recap

6.1 P - Polynomial (Class)

$L \in \{0,1\}^*$ is also in P if there exists an efficient algorithm which decides it.

6.2 NP - Nondeterministic Polynomial

$L \in \{0,1\}^*$ is also in NP if there exists an efficient algorithm V and a polynomial p which follow:

1. Completeness: $\forall x \in L, \exists y : V(x,y) = 1 \wedge |y| < p(|x|)$
2. Soundness: $\forall x \notin L \forall y : V(x,y) \neq 1 \vee |y| \geq p(|x|)$

6.3 PPT - Probobalistic Polynomial Time

This is a class of algorithms which must run in time polynomial to the size of their input, but also - must be capable of randomization, or 'flipping coins'.

6.4 IP - Interactive Proof

A key difference between an Interactive Proof and a proof for an NP proof is that the latter necessarily requires the prover to provide the verifier with something he can use to prove the truth of the calim to others.

We denote $(P, V)(x)$ to be the output of V (verifier) after the interaction between P and V on the input x. Both P and V can be thought of as PPT

algorithms or programs which are capable of communicating with one another.

These interactions are often described with an interaction diagram:

- P sends to V something
- V sends to P something else
- ...
- V accepts iff ...

Formal Definition: We say that $L \in IP$ if there exists a polynomial algorithm V, an unbounded algorithm P and some constant $c \in (0.5, 1]$ s.t.

1. Completeness: $\forall x \in L, Pr[(P, V)(x) = 1] > c$
2. Soundness: if $x \notin L, \forall P^* \in \mathbf{M}, Pr[(P^*, V)(x) = 1] < 1 - c$

Note: **M** denotes the set of turing machines.

7 Equivalence of IP separation constants

7.1 Iterative Runs

Given an IP protocol (P, V), let (P^k, V^k) be the protocol obtained by running (P, V) k times sequentially. V^k accepts iff in all iterations V accepted.

7.2 Lemma

if (P, V) is IP with perfect completeness then for every polynomial k, (P^k, V^k) is IP with perfect completeness and soundness error $2^{-\Omega(k)}$

Proof:

1. V^k is efficient (composition of polynomials).
2. Perfect Completeness - due to Perfect Completeness of the original protocol - each iteration is guaranteed to succeed thus the protocol always does.
3. Soundness: Let $x \in L, P^*$. We will show that $Pr[(P^*, V^k)(x) = 1] \leq 2^{-k}$. Denote by E_i the event that V^k accepts in the i'th iteration. Thus:

$$Pr[E_1 \wedge E_2 \wedge \ldots E_k] = \prod_{i=1}^{k} Pr[E_i | E_1 \wedge E_2 \wedge \ldots E_k]$$

   Claim: $Pr[E_i | E_1 \wedge E_2 \wedge \ldots E_k] \leq 0.5$.
   Proof: Assume toward a contradiction that $Pr[E_i | E_1 \wedge E_2 \wedge \ldots E_k] > 0.5$
   We design a prover P^{**} that convinces V with up to > 0.5:

P^{**} emulates (P^*, V) for iterations $1...i-1$ until the event $E_1 \wedge E_2 \wedge \ldots E_k$ happens and then runs (P^*, V) as the i'th iteration. Since the run of (P^*, V) for the i'th iteration only happens under the condition $Pr[E_i|E_1 \wedge E_2 \wedge \ldots E_k]$ - the probability for (P^*, V) to happen on the i'th iteration is exactly $Pr[E_i|E_1 \wedge E_2 \wedge \ldots E_k] > 0.5$ but this is also the probability for $Pr[(P^{**}, V)(x) = 1]$. Contradiction.

Thus:

$$Pr[E_1 \wedge E_2 \wedge \ldots E_k] = \prod_{i=1}^{k} Pr[E_i|E_1 \wedge E_2 \wedge \ldots E_k] \leq \prod_{i=1}^{k} 0.5 = 2^{-k}$$

8 Graph Isomorphism & IP Example

8.1 Graph Isomorphism - Definition

The graphs $G_1 = (V, E_1), G_2 = (V, E_2)$ are isomorphic or $(G_1, G_2) \in GI$ if $\exists \pi : V \longrightarrow V$ s.t. $(u, v) \in E_1 \iff (\pi(u), \pi(v)) \in E_2$.
More simply - two graphs are isomorphic if they are identical up to a renaming of their vertices.

GNI is the set of pairs of graphs which are isomorphic.

- Claim (no proof): $NP \subseteq IP$.

- Claim (proof sketch): $GI \in IP$.
  By finding the permutation π it is easy to check the GI condition over a given (G_1, G_2) thus we have an NP relation, meaning $GI \in NP$.

- Claim (no proof): $GNI \in IP$.

Tutorial 2

9 Perfect Completeness?

We have seen in the lecture how an IP can be reduced to an AM proof, or in other words; a public coin IP. Here, we would like to show how any AM can be reduced to an AM with perfect completeness (which is also an IP with perfect completeness).
This means that $\forall L \in IP$, it must have a public coin, perfectly complete interactive proof.

To construct the reduction, we start with the following z-round public coin protocol ($AM[z]$), which runs on input X:

- A samples $\alpha \leftarrow \{0,1\}^{rc}$, and sends it to M.
- M calculates $\beta = M(X, \alpha)$ and sends it to m.
- A accepts iff $A(X, \alpha, \beta) = 1$.

Where we assume completeness error $\epsilon > 0$:

$$\forall x \in L : Pr[(M, A)(X) = 1] \geq 1 - \epsilon$$

$$\Rightarrow \forall x \in L : Pr[\exists \beta : A(X, \alpha, \beta) = 1] \geq 1 - \epsilon$$

Now, we want to use it to construct an equivalent with perfect completeness; for that end, consider the alternative protocol:

- M' samples $s_1, s_2, ..., s_k \leftarrow \{0,1\}^{rc}$. s.t. (*)

  $$\forall \alpha \{0,1\}^{rc}, \exists i \in [k] : s_i \oplus \alpha \notin REJ$$

  .
- M' sends $s_1, s_2, ..., s_k$ to A'.
- A' samples $\alpha \leftarrow \{0,1\}^{rc}$ and sends it to M'.
- M' calculates $\forall i : \beta_i = M(X, s_i \oplus \alpha)$ and sends it.
- A' accepts iff $(\exists i : A(X, s_i \oplus \alpha, \beta_i)) = 1$

Lemma 1: if $x \in L$ then pre-processing succeeds.
We denote $\bar{s} = (s_1, s_2, ..., s_k)$. We say that $\bar{s}$ is 'good' if it satisfies (*).
To prove the lemma, we can show that:

$$\exists \bar{s} : \bar{s} \texttt{ is good}$$

We will show this by first showing:

$$Pr[\bar{s} \texttt{ is good }] > 0$$

To start off:

$$Pr_{\bar{s}}[\bar{s} \texttt{ is not good }] = Pr_{\bar{s}}[\exists \alpha, \forall i : s_i \oplus \alpha \in REJ]$$

$$= Pr_{\bar{s}}[\bigcup_{\alpha} (\forall i : s_i \oplus \alpha \in REJ)] \leq 2^{rc} \cdot \epsilon^k$$

From here we can see that for k of at-least rc, the probability is less then 1, meaning that the complementory probability is more than 0, meaning :

$$\exists \bar{s} : \bar{s} \texttt{ is good}$$

Completeness is trivial given Lemma 1.
Soundness of the protocol can be found in notes of Tutorial 2 in the website.