CAS CS 535: Complexity Theory

Lecturer: Mark Bun — Fall 2023

Lecture Notes 21:

Arthur-Merlin Proofs

Reading.

- Arora-Barak § 8.2

Last time: Toda's Theorem, Interactive Proofs

Definition 1. **IP** is the class of languages L admitting interactive proofs where the verifier V runs in randomized poly-time, the total communication is $\mathrm{poly}(|x|)$, and

$$\begin{aligned} x \in L &\implies \exists P \quad \Pr[\langle P, V\rangle(x) = 1] \geq 2/3, && \text{(completeness)} \\ x \notin L &\implies \forall P^* \quad \Pr[\langle P^*, V\rangle(x) = 1] \leq 1/3 && \text{(soundness).} \end{aligned}$$

1 Graph Non-Isomorphism

The Impossible Whopper example we did last time actually inspires an interesting interactive proof for the graph non-isomorphism problem.

Let $G = ([n], E)$ be an undirected graph. For a permutation $\pi : [n] \to [n]$, define $\pi(G) = ([n], \{(\pi(i), \pi(j)) \mid (i, j) \in E\})$ to be the same graph, but with all vertex labels permuted under π.

Write $G_0 \cong G_1$ (read: G_0 and G_1 are isomorphic) if there exists $\pi \in S_n$ such that $G_0 = \pi(G_1)$.

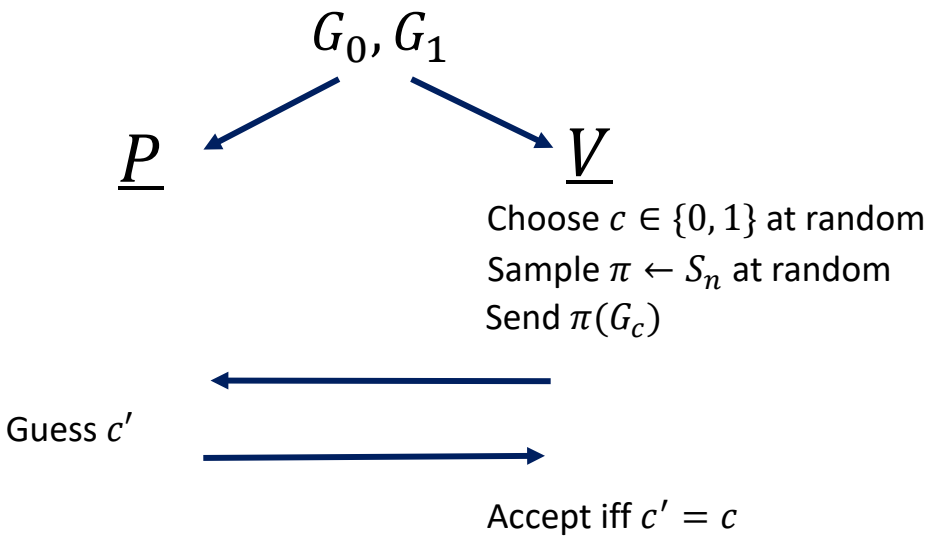
Define

$$\mathsf{GI} = \{\langle G_0, G_1\rangle \mid G_0 \cong G_1\}.$$

Then $\mathsf{GI} \in \mathbf{NP}$, taking π as the certificate.

- GI is currently not known to be in $\mathbf{P}$. (A relatively recent breakthrough of Babai'16 showed that it can be solved in quasi-polynomial time.)
- The complementary problem $\mathsf{GNI} = \overline{\mathsf{GI}}$ is not known to be in $\mathbf{NP}$, but it does have an efficient interactive proof, described below.

Claim 2. $\mathsf{GNI} \in \mathbf{IP}$.



Proof. The computation and check done by V run in poly-time. For correctness, first, if $G_0 \ncong G_1$, then exactly one of these graphs is isomorphic to $\pi(G_c)$. So P can determine $c' = c$ and hence

$$\Pr[\langle P, V\rangle(G_0, G_1) = 1] = 1.$$

On the other hand, if $G_0 \cong G_1$, then both graphs are isomorphic to $\pi(G_c)$. So no matter what strategy P^* is used to guess c, we have

$$\Pr[\langle P^*, V\rangle(G_0, G_1) = 1] \leq 1/2.$$

□

2 Bounded Rounds and Public Coins

We defined the class **IP** to consist of languages with (private coin) interactive proofs using an arbitrary polynomial number of messages. We can refine the definition by taking $\mathbf{IP}[k]$ to be the class of languages with proofs using k messages, where k might be a constant or it might be a growing function of n. In this notation, $\mathbf{IP} = \mathbf{IP}[\mathrm{poly}(n)]$.

We can also consider interactive proofs where we restrict the verifier to reveal its coin tosses. Without loss of generality, we can assume that random coin tosses are all its sends to the prover, since any additional computation could be done by the prover itself. This gives rise to public coin or "Arthur-Merlin" proofs.

Definition 3. For every k, the class $\mathbf{AM}[k]$ consists of languages with k-round interactive proofs where the verifier's messages are random bits. (And it is not allowed to access randomness beyond the bits it sends.)

The whimsical name comes from the following analogy (Babai-Moran, "Arthur-Merlin games: A randomized proof system, and a hierarchy of complexity classes").

> King Arthur recognizes the supernatural intellectual abilities of Merlin but does not trust him. How should Merlin convince the intelligent but impatient King that a string x belongs to a

given language L? ... An Arthur-Merlin protocol defines a combinatorial game, to be played by Arthur, whose moves are random, and Merlin, who is capable of making optimal moves.

# Messages	Private coins	Public coins
$\mathrm{poly}(n)$	$\mathbf{IP} = \mathbf{IP}[\mathrm{poly}]$	$\mathbf{AM}[\mathrm{poly}]\ (= \mathbf{IP})$
k	$\mathbf{IP}[k]$	$\mathbf{AM}[k] = \overbrace{\mathbf{AMA}\ldots}^{k}$
2	$\mathbf{IP}[2]$	$\mathbf{AM}$
2 (Prover speaks first)	$\mathbf{MA}$	$\mathbf{MA}$

Low levels of the "$\mathbf{AM}$ hierarchy" admit very clean descriptions that are worth unpacking.

Definition 4. A language $L \in \mathbf{MA}$ if there exists a deterministic poly-time V such that

$$\begin{aligned} x \in L &\implies \exists \pi \quad \Pr_r[V(x, \pi, r) = 1] \geq 2/3 \\ x \notin L &\implies \forall \pi^* \quad \Pr_r[V(x, \pi^*, r) = 1] \leq 1/3. \end{aligned}$$

Definition 5. A language $L \in \mathbf{AM}$ if there exists a deterministic poly-time V such that

$$\begin{aligned} x \in L &\implies \Pr_r[\exists \pi V(x, \pi, r) = 1] \geq 2/3 \\ x \notin L &\implies \Pr_r[\exists \pi^* V(x, \pi^*, r) = 1] \leq 1/3 \\ &\equiv \Pr_r[\forall \pi^* V(x, \pi^*, r) = 0] \geq 2/3. \end{aligned}$$

Proposition 6. $\mathbf{MA} \subseteq \mathbf{AM}$

Proof. Let $L \in \mathbf{MA}$, with associated verifier V. Repeat the verifier $O(p(n))$ times (using the same proof, but fresh randomness) to get

$$\begin{aligned} x \in L &\implies \exists \pi \in \{0,1\}^{p(n)}\ \Pr[V'(x, \pi, r) = 1] \geq 1 - 2^{-2p(n)} \\ x \notin L &\implies \forall \pi^* \in \{0,1\}^{p(n)}\ \Pr[V'(x, \pi^*, r) = 1] \leq 2^{-2p(n)}. \end{aligned}$$

Now consider the following $\mathbf{AM}$ protocol for L: On challenge r, the prover responds with the "good" choice of π from the $\mathbf{MA}$ protocol. Then we have

$$\begin{aligned} x \in L &\implies \Pr[V'(x, \pi, r) = 1] \geq 1 - 2^{-2p(n)} \geq 2/3 \\ x \notin L &\implies \Pr[\exists \pi^* V'(x, \pi^*, r) = 1] \\ &\leq \sum_{\pi^* \in \{0,1\}^{p(n)}} \Pr[V'(x, \pi^*, r) = 1] \\ &\leq 2^{p(n)} \cdot 2^{-2p(n)} = 2^{-p(n)} \leq 1/3. \end{aligned}$$

□

Proposition 7. $\mathbf{MA} \subseteq \mathbf{\Sigma}_2^p \cap \mathbf{\Pi}_2^p$ *and* $\mathbf{AM} \subseteq \mathbf{\Pi}_2^p$.

Proof idea: The proofs are similar to the proof that $\mathbf{BPP} \subseteq \mathbf{\Sigma}_2^p$ – Basically, that proof allows one to replace every Arthur message by a round of existential and a round of universal guessing, in either order.

Proposition 8. *For any constant k, we have* $\mathbf{AM}[k] = \mathbf{AM}(= \mathbf{AM}[2])$.

Proof idea: Generalizing Proposition 6 lets us repeatedly switch and collapse Merlin and Arthur moves.

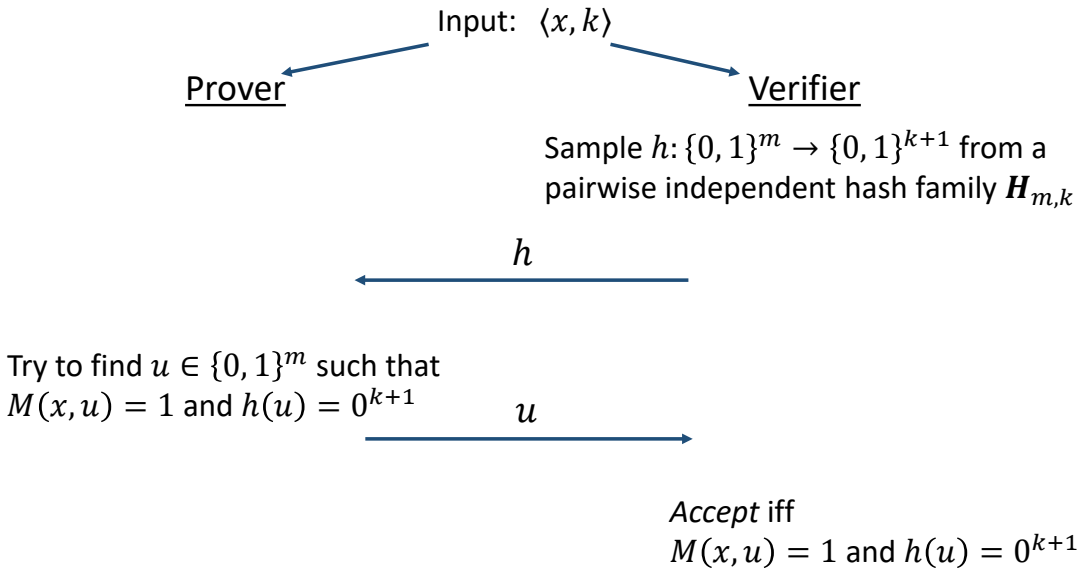
3 Approximate Counting in AM

For a function $f : \{0,1\}^* \to \mathbb{N}$, define the promise problem $\mathsf{Gap}_2 f$ by

$$\mathsf{Gap}_2 f_Y = \{\langle x, k\rangle \mid f(x) \geq 2^k\}$$
$$\mathsf{Gap}_2 f_N = \left\{\langle x, k\rangle \mid f(x) \leq \frac{1}{2} \cdot 2^k\right\}.$$

This captures the computational problem of estimating the value of f up to a constant factor. It's also called the "set size lower bound" problem, since the goal is to establish a lower bound on $f(x)$, interpreted as the size of a set of certificates for an $\mathbf{NP}$ relation. On your homework, you (essentially) showed that if $f \in \#\mathbf{P}$, then $\mathsf{Gap}_2 f \in \mathbf{PromiseBPP}^{\mathbf{NP}}$. That proof actually shows $\mathsf{Gap}_2 f \in \mathbf{PromiseAM}$.

To see this, let $f(x) = \#\{u \in \{0,1\}^m \mid M(x,u) = 1\}$ for some poly-time TM M, and let $\mathcal{H}_{m,k} = \{h : \{0,1\}^m \to \{0,1\}^{k+1}\}$ be an efficiently sampleable/computable family of pairwise independent hash functions. The consider the following two-message interactive proof for the $\mathsf{Gap}_2 f$ problem:



Completeness: If $f(x) \geq 2^k$, then

$$\Pr_h[\exists u \text{ s.t. } M(x,u) = 1 \wedge h(u) = 0^{k+1}] \geq \frac{1}{3}$$

using a Valiant-Vazirani style concentration argument. Meanwhile,

Soundness: If $f(x) \leq \frac{1}{2}2^k$, then

$$\Pr_h[\exists u \text{ s.t. } M(x,u) = 1 \wedge h(u) = 0^{k+1}] \leq \frac{1}{4}.$$

To get an $\mathbf{AM}$ protocol, we have to "shift" the acceptance probabilities a bit. This can be done by repeating the protocol a few times in parallel, and accepting iff the fraction of accepting iterations is, say, at least $7/24$.

4 Consequences of Approximate Counting Protocol

4.1 Graph Non-Isomorphism

Recall that $\mathsf{GNI} = \{\langle G_1, G_2\rangle \mid G_1 \ncong G_2\}$. The first interactive proof for GNI we saw seemed to make essential use of the verifier's ability to hide its randomness from the prover. It turns out that one can use a reduction to approximate counting to give a public coin protocol for this problem.

Proposition 9. $\mathsf{GNI} \in \mathbf{AM}$.

Proof sketch. Given two graphs G_1, G_2 on n vertices, define

$$f(G_1, G_2) = \#\{H \mid H \cong G_1 \text{ or } H \cong G_2\}.$$

Morally speaking, this is a $\#\mathbf{P}$ counting problem, where the certificate is a permutation π such that either $\pi(H) = G_1$ or $\pi(H) = G_2$.[1]

If $G_1 \cong G_2$, then (as long as the graph has no nontrivial automorphisms), we have $f(G_1, G_2) = n!$. On the other hand, if $G_1 \ncong G_2$, then (again, assuming no nontrivial automorphisms for either graph), we have $f(G_1, G_2) = 2n!$. Running our approximate counting protocol with $k = \log(2n!)$ thus gives an $\mathbf{AM}$ protocol for this problem.

To deal with the issue of automorphisms, i.e., permutations π that take a graph to itself, we modify our definition of f to

$$f(G_1, G_2) = \#\{(H, \pi) \mid (H \cong G_1 \text{ or } H \cong G_2) \text{ and } \pi(H) = H\}.$$

One can verify, using the fact that the set of automorphisms of a graph is a subgroup of S_n, that for any pair of graphs, we have $G_1 \cong G_2 \implies f(G_1, G_2) = n!$ and $G_1 \ncong G_2 \implies f(G_1, G_2) = 2n!$. □

This result has the following interesting consequence:

Theorem 10. *If* GI *is* $\mathbf{NP}$*-complete, then* $\mathbf{PH}$ *collapses.*

Proof. If GI is $\mathbf{NP}$-complete, then GNI is $\mathbf{coNP}$-complete. This implies $\mathbf{coNP} \subseteq \mathbf{AM}$. Now we have

$$\begin{aligned}
\mathbf{\Sigma}_2^p &= \exists\forall\mathbf{P} \\
&= \exists\mathbf{coNP} \\
&\subseteq \exists\mathbf{AM} \\
&= \mathbf{MAM} \\
&= \mathbf{AM} \\
&\subseteq \mathbf{\Pi}_2^p,
\end{aligned}$$

so the polynomial hierarchy collapses to the second level. □

[1] The reason why this isn't quite correct is that we should be counting certificates themselves. More accurately, we have $f \in \# \cdot \mathbf{NP}$, defined as the set of functions g for which $g(x) = \#\{u \mid \exists v \text{ s.t. } M(x, u, v) = 1\}$ where M runs in poly time. (See, e.g., Hemaspaandra-Vollmer, "The Satanic Notations: Counting Classes beyond $\#\mathbf{P}$ and Other Definitional Adventures.") The approximate counting protocol we saw works for this larger class as well.

4.2 Simulating Private Coins with Public Coins

One can (in a sense) generalize the idea behind our proof that $\mathsf{GNI} \in \mathbf{AM}$ to show

Theorem 11 (Goldwasser-Sipser)**.** *For every k computable in $\mathrm{poly}(n)$ time, we have*

$$\mathbf{IP}[k] \subseteq \mathbf{AM}[k+2].$$

The basic idea is to use the approximate counting protocol to establish a lower bound on the number of random strings that would have caused the private coin verifier to accept in the original protocol.