CSE 201 Advanced Complexity
Lectures 14 and 15
May 19, 21, and 26 2003

Private Coins = Public Coins
$\mathbf{IP^k = AM^k}$

Instructor: Russell Impagliazzo
Notes scribed by: Sashka Davis

There are two notions of interactive proof systems. In private coin model (IP) introduced by Goldwasser, Micali, and Rackoff, the Verifier uses a secret random source to compose the queries to the Prover. The public coin model (AM) the Verifier uses public coin tosses, meaning that the Prover sees the coin tosses but not all at the same time. AM was introduced by Babai "*to place the matrix group non-membership and matrix group order problems in a complexity class* **just above NP** [?]." Goldwasser and Sipser showed that the two interactive proof systems IP and AM are equivalent in power. The proof requires two steps (techniques): *m-game parallel repetition* also known as the amplification lemma, and the *approximate set size lower bound technique.* The approximate set lower bound is an application of the Carter-Wegman universal hash function, was first used by L. Stockmeyer to show that approximate counting is in Σ_3^P. We have used and analyzed the double hashing in proving that if we can solve the search problem for polynomial time computable relation $R(x, y)$ δ-almost on the uniform distribution, then we can solve it for any samplable distribution.

An interactive proof system consists of two parties a Prover and a Verifier. The Prover is capable of computing any function, while the Verifier is a probabilistic polynomial time algorithm. Both Prover and Verifier know the instance x. A language L is in $\mathbf{IP^k}$ if there is a k-round interactive proof protocol:

$$
\begin{array}{ccc}
\text{Prover} & & \text{Verifier} \\
 & \xleftarrow{\;z_1=f(1,x,r)\;} & \\
 & \xrightarrow{\;y_1\;} & \\
 & \xleftarrow{\;z_2=f(2,x,r,y_1)\;} & \\
 & \xrightarrow{\;y_2\;} & \\
 & \vdots & \\
 & \xleftarrow{\;z_k=f(k,x,y_1,\ldots,y_{k-1})\;} & \\
 & \xrightarrow{\;y_k\;} &
\end{array}
$$

Verifier outputs $V(x, r, y_1, y_2, \cdots, y_k)$.

$$x \in L \implies \text{there exists a Prover such that} \Pr_r\left[V(x, r, y_1, \cdots, y_k) = accept\right] \geq \frac{2}{3}$$

$$x \notin L \implies \max_{\text{Prover}} \Pr_r[\, V(x, r, y_1, \cdots, y_k) = accept\,] \leq \frac{1}{3}$$

1 m-game parallel repetition

Consider a private coin interactive proof system with k rounds ($\mathbf{IP^k}$). The goal is to define a new protocol (private coin) based on the protocol defined above, and to exponentially strengthen the consistency and soundness of the interactive proof without increasing the number of rounds. This new protocol is called m-game parallel repetition. Consistency is established easily, Section **??** proves exponential soundness.

Next we define the m-game parallel repetition. Verifier uses m random (secret) strings $r_1, \cdots, r_m$. The m-game parallel repetition is a widened version of the original protocol, where Verifier sends to the Prover m requests, each based on different (independent) random strings. Each request sent to the Prover is an m-tuple and so is Prover's response. At the end the Verifier computes the majority of the outcomes of the m independent games played.

$$
\begin{array}{ccc}
\text{Prover} & & \text{Verifier} \\
& \xleftarrow{z_1^1,\cdots,z_1^m,z_1^i=f(1,x,r_i)} & \\
& \xrightarrow{y_1^1,\cdots,y_1^m} & \\
& \xleftarrow{z_2^1,\cdots,z_2^m,z_2^i=f(2,x,r_i)} & \\
& \xrightarrow{y_2^1,\cdots,y_2^m} & \\
& \vdots & \\
& \xleftarrow{z_k^1,\cdots,z_k^m,z_k^i=f(k,x,y_1^i,\ldots,y_{k-1}^i)} & \\
& \xrightarrow{y_k^1,\cdots,y_k^m} &
\end{array}
$$

Verifier outputs

$$V = \begin{cases} 1 & \text{if } \sum_{i=1}^m V^i(x, r_i, y_1^i, y_2^i, \cdots, y_k^k) \geq \frac{m}{2} \\ 0 & \text{otherwise} \end{cases}$$

We must prove soundness and consistency.

Theorem 1. **[Consistency]** *If the Prover has a strategy in the original Protocol to convince the Verifier with probability at least $\frac{2}{3}$ then in the m-game protocol the Verifier accepts with probability at least $1 - 2^{\Omega(m)}$.*

Proof. All the m-games played in parallel are independent, so we will use one of the Chernoff bounds ([**?**]) to prove the result, namely: If X is a sum of Boolean independent random variables, μ be the expectation of the sum, and $\delta \in (0, 1]$ then $\Pr[X < (1-\delta)\mu] < e^{-\frac{\mu \cdot \delta^2}{2}}$. We apply the bound with $X = \sum_{i=1}^m V^i(x, r_i, y_1^i, y_2^i, \cdots, y_k^k)$, $\mu = \frac{2}{3}m$, and $\delta = \frac{1}{4}$. $\Pr[\,\sum_{i=1}^m (V^i) < \frac{m}{2}\,] \leq e^{-\frac{m}{12}}$, thus $\Pr[$ Verifier accepts $] \geq 1 - 2^{\Omega(m)}$. □

Our next goal is to establish soundness of the new protocol:

Theorem 2. **[Soundness]** *If no Prover strategy in the original Protocol causes the Verifier to accept with probability bigger than $\frac{1}{3}$ then no Prover strategy in the m-game parallel repetition causes the Verifier to accept with probability bigger than $2^{-\Omega(m)}$.*

The idea is that Prover's best strategy for the m-game parallel repetition is m independent uses of his best strategy for the original protocol (a single-fold game). In what follows we will first look at the original protocol as a game tree, and establish definitions used later to analyze the m-game tree.

1.1 Game tree representation of the original protocol

We visualize the original protocol as a game tree. The leaves of the tree are labeled with the

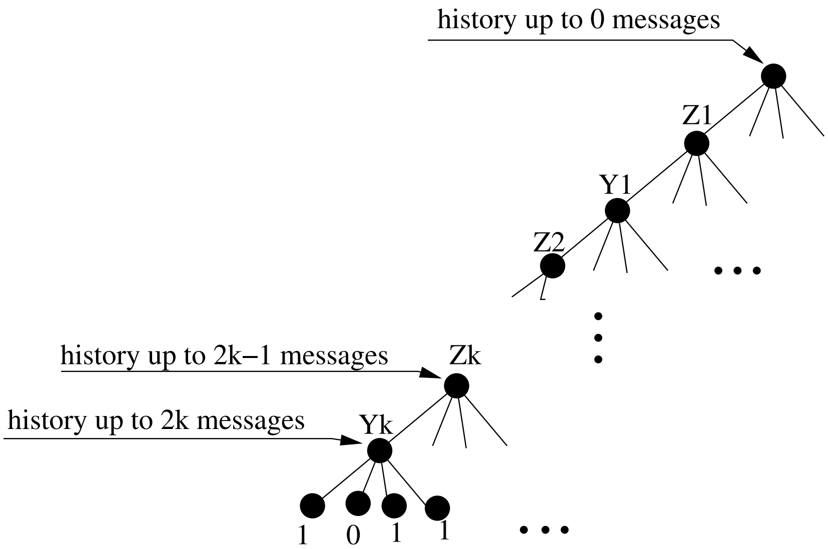

Figure 1: Game tree for the original protocol (single fold game).

output of the Verifier, 0 if the Verifier rejects and 1, otherwise. The internal nodes represent the messages exchanged by both players (Verifier and Prover), so each node is the history of the game so far. The root of the tree is node h_0, which is history up to 0 messages, and node h_{2k} represents the complete history up to $2k$ messages. On even history the Verifier moves, and on odd history the Prover moves. For a given history $h_{2i+1} = (z_1, y_1, z_2, y_2, \cdots, z_i)$ denote the consistency of the history to be the set of those random strings r, that make the history valid, i.e., $Cons(h_{2i+1}) = \{r \mid \forall i \; 1 \leq i \leq i \;\; f(i, r, x, (y_1, \cdots, y_{i-1})) = z_i)\}$. The fist query sent by the Verifier z_1 depends only on r and the instance x, but further requests do depend on r, x, and the previous responses from the Prover.

Claim 3. $\forall h_{2i+1}, \forall y_{i+1} \quad Cons(h_{2i+1}) = \cup_{z_{i+2}} Cons(h_{2i+1} \circ (y_{i+1}, z_{i+2}))$.

Define the weight of a consistency to be $w(h_{2i+1}) = |Cons(h_{2i+1})|$.

Lemma 4. $\forall h_{2i+1}, \forall y_{i+1} : w(h_{2i+1}) = \sum_{z_{i+2}} w(h_{2i+1} \circ (y_{i+1}, z_{i+1}))$.

Label any history h at any level i, with the probability of the history $P(h_i)$. Then odd history, which corresponds to a move by the Prover will be

$$P(h_{2i+1}) = \max_{y_{i+1}} P(h_{2i+1} \circ z_{i+1})$$

For the moves of the Verifier we have

$$P(h_{2i}) = \sum_{z_{i+1}} \frac{w(h_{2i} \circ z_{i+1})}{w(h_{2i})} P(h_{2i} \circ z_{i+1}) \tag{1}$$

Claim 5. *For every Prover strategy and any history h_j:*

$$\Pr_{r \in_u Cons(h_j)} [\text{ Verifier accepts} \mid \text{starting at } h_j] \leq P(h_j)$$

Proof. By induction on the height on the tree. Beginning with a tree of height 1,2, etc. □

Corollary: There exists a Prover strategy such that for any history h_j

$$\Pr_{r \in_u Cons(h_j)} [\text{ Verifier accepts } \mid \text{starting at } h_j] = P(h_j)$$

1.2 Analysis of the m-game parallel repetition

In m-game parallel repetition the Verifier uses m random strings $r_1, \cdots, r_m$ and sends a query which is an m-tuple and the Prover responds with an m-tuple. Basically, m independent invocations of the original protocol are played in parallel (as the name suggests). We look at a Prover who wants to win as many sub-games in the m-game as possible. The Prover's goal is to maximize

$$\sum_{i=1}^{m} V(r_i, x, (y_1^i, \cdots, y_k^i))$$

To analyze the m-game we build a game tree as above, except the nodes at level j are tuples $(h_j^1, \cdots, h_j^m)$. Label the leaves as the number of i's so that $V^i(x, r_i, h_{2k}^i) = 1$. In the new protocol the Verifier accepts by majority, that is if $\sum_{i=1}^m V^i(x, r_i, h_{2k}^i) \geq \frac{m}{2}$ then output 1, otherwise 0. Our goal is to establish Theorem **??**. Define random variables R_j to be the expected number of wins starting at level j, that is $R_j = \sum_{i=1}^m P(h_j^i)$. If Prover plays his optimal strategy going from level j to $j+1$ the expected number of wins should not change. His odds could change (for worse), however, if he plays a sub-optimal strategy. We will not consider sub-optimal strategies, since that makes the bound stronger. Our goal will be to show that when the Verifier moves, with high probability $R_j \leq R_{j-1} + m^{\frac{2}{3}}$. (If we demonstrate that we are almost done.)

Consider a single game i. If j is even then it is the Verifier's move, thus with probability $\frac{w(h_j^i \circ z_{\frac{j}{2}+1})}{w(h_j^i)}$ the probability of the history after $(j+1)$-st messages is $P(h_{j+1}^i) = P(h_j^i)$. Thus $\mathbf{Exp}[P(h_{j+1}^i)] = P(h_j^i)$ (follows from (1)).

Lemma 6. *Let $A_1, \cdots, A_m$ be Boolean independent random variables ($\forall A_i \in [0,1]$), each with expectation $\mathbf{Exp}[A_i] = a_i$. Then $\mathbf{Pr}[|\sum A_i - \sum a_i| \geq t] \leq 2 \cdot e^{-\frac{t^2}{8m}}$.*

Proof. Pick $A_1, \cdots, A_m$ according to the distribution. Then flip a coin with bias A_i. Let

$$X_i = \begin{cases} 1, & \text{if heads} \\ 0, & \text{if tails} \end{cases}$$

Overall $\mathbf{Exp}[X_i] = a_i$, but $\mathbf{Exp}[X_i|A_i] = A_i$. Now apply Chernoff bound

$$\Pr[|\sum X_i - \sum a_i| \geq \frac{t}{2}] \leq e^{-\frac{t^2}{8m}}$$

$$\Pr[|\sum X_i - \sum A_i| \geq \frac{t}{2}] \leq e^{-\frac{t^2}{8m}}$$

$\square$

R_{2k+1} is the expected outcome of the m-game. (A child of history h_{2k} in the m-game tree is a payoff vector, which is an m-tuple of 0's and 1's).

$$\Pr[\text{ Verifier rejects in m-game }] = \Pr[R_{2k+1} < \frac{m}{2}]$$

The expected number of wins of the Prover at the top of the m-game tree is

$$R_0 = \sum_{i=1}^{m} P(h_0^i) = \sum_{i=1}^{m} \frac{1}{3} = \frac{m}{3}$$

because the probability of winning a single game is at most 1/3.

Next we show that with high probability $R_j \leq R_{j-1} + m^{\frac{2}{3}}$ from which the proof of Theorem **??** follows.

Apply Lemma **??** with $A_i = P(h_{j+1}^i|h_j^i)$, $a_i = P(h_j^i)$, and $t = m^{\frac{2}{3}}$. We have that for all j:

$$\Pr[|R_{j+1} - R_j| > m^{\frac{2}{3}}] < 2 \cdot e^{-\frac{m^{1/3}}{8}}$$

By the union bound $\Pr[\ \exists j \text{ such that } |R_{j+1} - R_j| > m^{\frac{2}{3}}] < 2k \cdot e^{-\frac{m^{1/3}}{8}}$

$$\Pr[\ \forall j\ \ |R_{j+1} - R_j| < m^{\frac{2}{3}}] > 1 - 2k \cdot e^{-\frac{m^{1/3}}{8}}$$

The last inequality reads with probability at least $(1 - 2k \cdot e^{-\frac{m^{1/3}}{8}}) \qquad R_{2k+1} - R_{2k} < m^{2/3}$.

$$\begin{array}{ll} |R_{2k+1} - R_{2k}| < m^{\frac{2}{3}} & \text{Verifier's move} \\ |R_{2k} - R_{2k-1}| = 0 & \text{Prover's move} \\ |R_{2k-1} - R_{2k-2}| < m^{\frac{2}{3}} & \text{Verifier's move} \\ |R_{2k-2} - R_{2k-3}| = 0 & \text{Prover's move} \\ \vdots & \end{array}$$

When the Verifier moves, with very high probability the expected number of wins do not increase by more than $m^{\frac{2}{3}}$. On the other hand when the Prover moves, if he plays his optimal strategy, the expected number of wins should not change[1].

Sum up all equalities and inequalities above, and we have $|R_{2k+1} - R_0| < k.m^{\frac{2}{3}}$, with

[1]Prover's best strategy is deterministic. At best hist odds stay the same.

probability at least $(1 - 2k \cdot e^{-\frac{m^{1/3}}{8}})$, where R_{2k+1} is the expected number of games won. $\Pr_r[\,|R_{2k+1} - R_0| < k.m^{\frac{2}{3}}\,] \geq 1 - 2k \cdot e^{-\frac{m^{1/3}}{8}}$.

Chose $m = \Omega(k^3)$ then $R_{2k+1} < \frac{m}{3} + k.m^{\frac{2}{3}} = \frac{m}{2}$, with probability $(1 - 2k \cdot e^{-\frac{m^{1/3}}{8}})$. We conclude that with probability at least $(1 - 2k \cdot e^{-\frac{m^{1/3}}{8}})$ Verifier rejects.
Q.E.D.

2 IP=AM, step 2

Now we show that languages that have an interactive proof in the private coin model, have a proof in the public coin model. Suppose we have:

$$
\begin{array}{ccc}
\textsf{Prover} & & \textsf{Verifier} \\
 & \xleftarrow{\;y_1=f(1,x,r)\;} & \\
 & \xrightarrow{\;z_1\;} & \\
 & \xleftarrow{\;y_2=f(2,x,r,h_2)\;} & \\
 & \xrightarrow{\;z_2\;} & \\
 & \vdots & \\
 & \xleftarrow{\;y_k=f(k,x,h_{2k-1})\;} & \\
 & \xrightarrow{\;z_k\;} &
\end{array}
$$

Verifier outputs $V(x, r, h_{2k})$.

$$P(h_i) = \max_{\textsf{Prover}\ \text{strategy}} \Pr[\textsf{Verifier accepts, given } h_i]$$

In public coin protocol the Prover knows the random bits. He is not allowed to pick them, otherwise he will win, since picking the random bits corresponds to picking the moves of the Verifier. The idea is to use the set lower bound technique to either force the Prover (Merlin) to be honest or to catch him if he cheats. In the new protocol Merlin sends Arthur, the size of the set of "good" y's. A y is "good" y if it is consistent with the current history, and will cause Arthur to accept at the end if $x \in L$, and reject if $x \notin L$. Let $y \in \{0,1\}^s$, where s is the length of the message. Arthur selects a random hash function, hashes the set of all possible y's, and challenges Merlin to find one such y. So a single round of the IP protocol becomes the following 5 message exchanges between Arthur and Merlin. Let ρ_0 be the length of the random tape in the basic protocol and $\rho = m.\rho_0$. Thus 2^{ρ_i} is the number of random tapes consistent with the state of the protocol at the i-th round of the protocol (i.e. consistent with the history until the i-th round of the protocol).

Merlin		Arthur
	$\xrightarrow{t_1}$	$t_i \in \{0, ..., \rho_{i-1}\}$ $\rho_i = \rho_{i-1} - t_i$
	$\xleftarrow{M,b}$	$M \in_R \mathcal{H}, b \in_R \{0,1\}^{t_i - 2\log\rho}$ $\mathcal{H} = \{h \mid h : \{0,1\}^s \to \{0,1\}^{t_i - 2\log\rho}\}$
	$\xrightarrow{y_i}$	$M.y_i = b, h_{2i-1} = h_{2i-2} \circ y_i$
	$\xleftarrow{N,b'}$	$N \in_R \mathcal{H}', b' \in \{0,1\}^{\rho_i - 2\log\rho}$ $\mathcal{H}' = \{h \mid h : \{0,1\}^\rho \to \{0,1\}^{\rho_i - 2\log\rho}\}$
	$\xleftarrow{r,z_i}$	$r \in Cons(h_{2i-1}), Nr = b'$
$h_{2i} = h_{2i-1} \circ z_i$		

What does the above exchange accomplish?

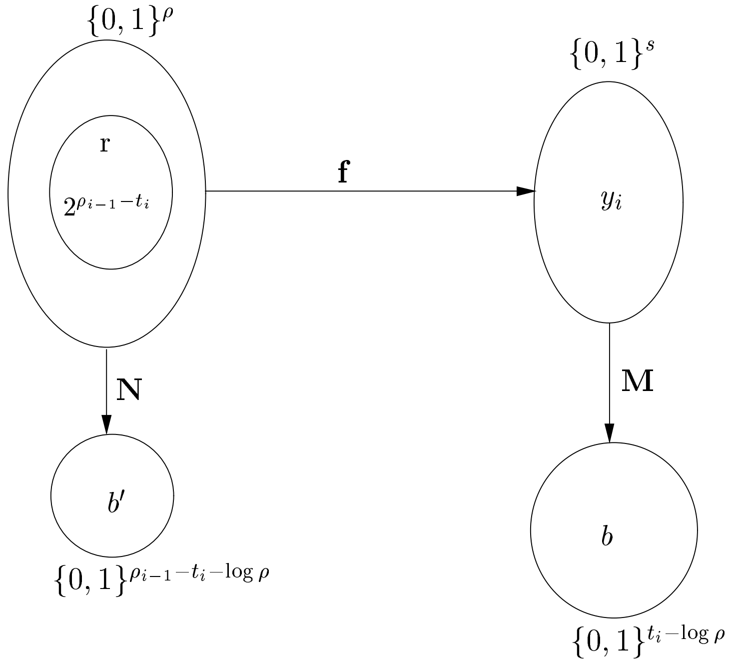

Figure 2: Double hashing

Initially, Merlin sends Arthur t_1, the size of the set of "good" y's. That means that there are 2^{t_1} y's, each with $2^{\rho - t_1}$ r's that map to it. Thus the probability of getting a "good" y is Pr["good" y] $= \frac{1}{2^{\rho - t_1}}$. Arthur challenges Merlin, by picking a random hash function

$M : \{0,1\}^{\rho} \to \{0,1\}^{t_1 - 2\log \rho}$, and a random vector b in the range, and asks Merlin to find y_1. Merlin sends Arthur y_1 such that $M.y_1 = b$. At round i Merlin would have to send y_i which satisfies the (M, b) challenge, but y_i must also be consistent with the history $h_{2i-2} \circ y_i$. The number of random tapes that yield such y_i is $2^{\rho_{i-1} - t_i}$ (the reason is that a potion of the random tape will get fixed during each round of the protocol). Next Arthur challenges Merlin to find a particular r with the (N, b') challenge, which also must be consistent with the current history. Thus at each round Merlin is forced to "guess" a portion of the random tape, and then respond to the request that Merlin produced with an answer z_i. The length of random tape at the end of round i is ρ_i. Then during each round $i+1$ we have:

- $t_{i+1} \in \{1, ..., \rho_i\}$
- There are $2^{t_{i+1}}$ y_i's each with $2^{\rho_i - t_{i+1}}$ random tapes that map to it. Thus probability of getting a "good" y at round i is $\Pr[\text{ good } y] = \frac{2^{\rho_{i-1} - t_i - 2\log \rho}}{2^{\rho - 2\log \rho}}$
- Merlin is forced to come with a random r, consistent with the current history: Arthur selects a random N and b' and challenges Merlin to present an r which is from the set of r whose size as Merlin claimed is $2^{\rho_i - t_{i+1}}$.

At the end of the protocol, there should be 2^{ρ_k} r's, consistent with the history h_{2k}. Merlin presents one such r, and Arthur outputs $V(x, r, h_{2k})$.

2.1 Completeness $[x \in L]$

ρ_i is the length of the random tape:w at the end of round i. Our goal is to show:

Lemma 7. *If the invariant $|Cons(h_{2i})| = 2^{\rho_i}$ holds then*

- $\Pr[y_i | h_{2i-2}] = 2^{-t_i}$.
- *y_i's are uniformly distributed within the set above.*

Claim 8. *Let δ be the probability of an honest Prover (Merlin) failing to convince the Verifier (Arthur) in the original protocol, i.e., $P(0) = 1 - \delta$. An honest Merlin maintains the invariants:*

1. $|Cons(h_{2i})| > 2^{\rho_i}$
2. *$P(h_{2i}) > 1 - 2^i.\delta$, that is if we simulate the original protocol with the history h_{2i}, probability of failure is doubling with each round.*

Proof. By induction. The invariant holds initially, when $i = 0$. Assume invariant holds at the start of the simulation of round i. The history so far is h_{2i-2}, and the consistency is $Cons(h_{2i-2})$. First, throw out y_i such that $\Pr[Cons(h_{2i-1} \circ y_i)] \le 1 - 2^i \cdot \delta$. This leaves at least half of the r's. Thus what we have left is at least $2^{\rho_{i-1}}$ random tapes. Of the y's left we have

$$\sum_{y_i} |Cons(h_{2i-2} \circ y_i)| \ge 2^{\rho_{i-1} - 1}$$

Group the set of y_i's by "how likely they are" as follow $2^{\rho_{i-1}-t_i} \geq |Cons(h_{2i} \circ y_i)| \geq 2^{\rho_{i-1}-t_i-1}$. Then

$$\sum_{t=1}^{\rho} \sum_{y_i \text{as above}} |Cons(h_{2i-2} \circ y_i)| \geq 2^{\rho_{i-1}-1}$$

where each y_i above is such that $2^{\rho_{i-1}-t_i} \geq |Cons(h_{2i} \circ y_i)| \geq 2^{\rho_{i-1}-t_i-1}$.

Therefore there exists t_i such that

$$\sum_{y_i \text{ as above}} |Cons(h_{2i-2} \circ y_i)| \geq \frac{2^{\rho_{i-1}-1}}{\rho}$$

Now we claim that

$$\Pr_{M,N}[\, \exists\, y_i \mid (P(h_{2i-2} \circ y_i)\,] \geq 1-2^i\delta) \wedge (|Cons(h_{2i-2} \circ y_i)| \geq 2^{\rho_{i-1}-t_i}) \wedge (M \cdot y_i = b)] \geq 1-O(1/\rho)$$

We show the validity of the claim by reducting the calculation of the probability to variance calculation.

$$\Pr_{M,N}[\, \exists\, y_i \mid (P(h_{2i-2} \circ y_i)\,] \geq 1-2^i\delta) \wedge (|Cons(h_{2i-2} \circ y_i)| \geq 2^{\rho_{i-1}-t_i}) \wedge (M \cdot y_i = b)] =$$

$$1 - \Pr_{M,N}[\, \forall\, y_i \mid (P(h_{2i-2} \circ y_i)\,] \geq 1-2^i\delta) \wedge (|Cons(h_{2i-2} \circ y_i)| \geq 2^{\rho_{i-1}-t_i}) \wedge (M \cdot y_i \neq b)]$$

We want to limit the following probability from above:

$$\Pr_{M,N}[\, \forall\, y_i \mid (P(h_{2i-2} \circ y_i)\,] \geq 1-2^i\delta) \wedge (|Cons(h_{2i-2} \circ y_i)| \geq 2^{\rho_{i-1}-t_i}) \wedge (M \cdot y_i \neq b)]$$

Let X_{y_i} be indicator variables defined as:

$$X_{y_i} = \begin{cases} 1 & \text{if } M \cdot y_i = b \\ 0 & \text{otherwise} \end{cases}$$

Define $X = \sum X_i$. The expectation of X is $E[X] = \sum E[X_{y_i}] \geq \frac{1}{\rho}$. X_{y_i} are pairwise independent thus thus the variance $V[X] = \sum V[X_{y_i}]$ $V[X_{y_i}] = E[X_{y_i}^2] - E[X_{y_i}]^2 \leq E[X_{y_i}]$ Therefore $V[X] \leq E[X]$.
By Chebychev's inequality:

$$\Pr[Y = 0] \leq \Pr[|X - E[X]| \geq E[X]] \leq \frac{V[X]}{E[X]} \leq E[X] = \frac{1}{\rho}$$

Therefore

$$\Pr_{M,N}[\, \exists\, y_i \mid (P(h_{2i-2} \circ y_i)\,] \geq 1-2^i\delta) \wedge (|Cons(h_{2i-2} \circ y_i)| \geq 2^{\rho_{i-1}-t_i}) \wedge (M \cdot y_i = b)] \geq 1-O(1/\rho)$$

$\square$

An honest Merlin sends t_i and then y_i such that with high probability $P(h_{2i-1}) \geq 1 - 2^i \cdot \delta$ and $|Cons(h_{2i-1})| \geq 2^{\rho_i}$ By our standard variance argument Merlin survives the (N, b') challenge with probability $1 - O(1/\rho)$ and makes the optimal move z_i, so that $P(h_{2i-1} \circ z_i) = P(h_{2i-1}) \geq 1 - 2^i \cdot \delta$.

By the invariant, entering the last round, k, we have $P(h_{2k}) \geq 1 - 2^k \cdot \delta$. Now as long as $2^{\rho_k} \cdot \delta < \frac{1}{2}$ we are fine. So the initial δ must be exponentially small. There are more than $2^{\rho_{k-1}}$ $r \in Cons(h_{2k})$, so that $V(x, r, h_{2k}) = 1$ and by the same argument at least one such r will satisfy the last (N, b') challenge[2].

2.2 Soundness

$x \notin L$ implies $P(0) \leq \delta$. Assume $\delta < \rho^{4k^2+4}$. We must show two invariants:

(1∗) $\quad |Cons(h_{2i}| = 2^{\rho_i}$

(2∗) $\quad P(h_{2i}) = \rho^{8i^2}.\delta$

Lemma 9. *If in any round i, $|Cons(h_{2i-1})| < \frac{2^{\rho_i}}{\rho^3}$ then Arthur rejects with probability $1 - O(1/\rho)$.*

Proof. Merlin gets caught with the (N, b') challenge. □

Lemma 10.
If $\quad |Cons(h_{2i-2})| \leq 2^{\rho_i}.\rho^{e_i} \quad$ *(1)*
but $\quad |Cons(h_{2i})| \geq 2^{\rho_i}.\rho^{e_i+8} \quad$ *(2)*
then Arthur rejects with high probability.

Proof. Look at the set of y_i's $\{y_i$ such that $|Cons(h_{2i})| \geq 2^{\rho_i}.\rho^{e_i+8}\}$. How big is this set? It can't be too big, because (1) must hold, and (1) limits the number of such y_i's.

$$|\{y_i \text{ such that } |Cons(h_{2i})| \geq 2^{\rho_i}.\rho^{e_i+8}\}| \leq \frac{|Cons(h_{2i-2})|}{2^{\rho_i}.\rho^{e_i+8}} \leq \frac{2^{\rho_{i-1}}.\rho^{e_i}}{2^{\rho_i}.\rho^{e_i+8}} = \frac{2^{\rho_{i-1}-\rho^i}}{\rho^8} = \frac{2^{t_i}}{\rho^8}$$

Therefore

$$\Pr[\exists\, y_i, s.t. (My_i = b) \wedge (|Cons(h_{2i-1})| \geq 2^{\rho_i}.\rho^{e_i+8})] \leq \frac{1}{\rho^8}$$

□

Lemma 11. *Assume (1*) holds, that is $2^{\rho_{i-1}.\rho^{8(i-1)}} \geq |Cons(h_{2i-2})| \geq \frac{2^{\rho_{i-1}}}{\rho^3}$ then with high probability either $P(h_{2i}) \leq \rho^{16}.P(h_{2i-2}))$ or Arthur rejects.*

Proof.

$$|Cons(h_{2i-2})| = \sum_{y_i \text{ s.t. } P(h_{2i-2} \circ y_i) \geq \rho^{16i}.P(h_{2}i)} |Cons(h_{2i-2} \circ y_i)|$$

[2] Are the y's uniformly distributed? - If t_i's are fixed in advance then y_i's must be uniformly, and randomly distributed, otherwise Merlin will be forcing a specific r so that $2^{t_i - 2\log\rho}$ must be a reasonable fraction.

By an avaraging argument it follows that:

$$\Pr[|Cons(h_{2i-2})| = \sum_{y_i \text{ s.t. } P(h_{2i-2} \circ y_i) \geq \rho^{16i}.P(h_2 i)} |Cons(h_{2i-2} \circ y_i)|] \leq \frac{1}{\rho^{16i}}.|Cons(h_{2i-2})|$$

The number of y_i's such that $|Cons(h_{2i-2} \circ y_i)| \leq \frac{2^{\rho_i}}{\rho^3}$ and $P(h_{2i-2} \circ y_i) \geq \rho^{16i}.P(h_2 i)$ is smaller than $\frac{2^{\rho_{i-1}}.\rho^3}{\rho^{8i}.2^{\rho_i}} = \frac{2^{t_i}}{\rho^{8i-3}}$.
Thus

$$\Pr[\exists y_i \text{ as above, and } My_i = b] \leq \frac{1}{\rho^{8i-5}} = O(\rho^{-3})$$

□

What we have shown so far is that one of the following three cases occurs:

Case 1. WHP Arthur rejects, because Merlin did not meet the (M, b) challenge.

Case 2. or $P(h_{2i-2} \circ y_i) \leq \rho^{8i}.P(h_{2i-i})$

Case 3. or $|Cons(h_{2i-2} \circ y_i)| \leq \frac{2^{\rho_i}}{\rho^3}$

If Case 3 happens then WHP Arthur rejects after (N, b') challenge.

Theorem 12. *If $x \notin L$ then Arthur rejects with high probability.*

Proof. By Lemma **??** and **??** either Arthur has rejected or at round k, the consistency $|Cons(h_{2k})| \leq 2^{\rho_k} \cdot \rho^{8k}$ and $P(h_{2k}) \leq \rho^{16k^2} \cdot \delta$

$$|\{r \text{ s.t. } r \in Cons(h_{2k}) \wedge (V(x, r, h_{2k}) = 1)\}| \leq 2^{\rho_k} \cdot \rho^{8k+16k^2} \cdot \delta << \frac{2^{\rho_k}}{\rho^3}$$

$$\Pr_{b'}[\exists r \text{ as above such that } Nr = b' \,] \leq \frac{1}{\rho}$$

□

A Pairwise independent family of hash functions

A family of pairwise independent hash functions $\mathcal{H}$ is defined as $h \in \mathcal{H} : \{0,1\}^n \to \{0,1\}^m$ and satisfies:

1. For $a \in \{0,1\}^m$, and $\forall x \in \{0,1\}^n : \Pr_{h \in \mathcal{H}}[h(x) = a] = \frac{1}{2^m}$
2. $\forall x \neq y, \Pr_{h \in \mathcal{H}}[h(x) = a | h(y) = b] = \frac{1}{2^m}$

References

[1] S. Goldwasser, M. Sipser. *Private Coins versus Public Coins in Interactive Proof Systems,* FOCS 1986

[2] V. Vazirani *Approximation Algorithms,* Springer 2001