Randomized complexity classes

Today: the Arthur vs. Merlin hierarchy, and interactive proofs
Today

- Arthur vs. Merlin games
- Interactive proofs
- Various characterizations of AM
Arthur vs. Merlin games
Trading Group Theory for Randomness

László Babai
Dept. Algebra
Eötvös University
Budapest
Hungary H-1088

Dept. Computer Science
University of Chicago
1100 E 58th St.
Chicago, IL 60637

Abstract.

In a previous paper [8], we proved, using the elements of the theory of nilpotent groups, that none of the fundamental computational problems in matrix groups has polynomial time complexity.

These problems were also shown to belong, assuming an unproven hypothesis concerning groups, to a random oracle $B$, the mentioned matrix problems belonging to $\mathsf{NP}^B = \mathsf{coNP}^B$.

The aim of this paper is to replace most of and unproven group theory of [8] by elementary arguments. The result we prove is that to a random oracle $B$, the mentioned matrix problems belong to $\mathsf{NP}^B = \mathsf{coNP}^B$.

The problems we consider are membership of a matrix group given by a list of generators. These can be viewed as multilinear pseudorandom vectors relative to the discrete logarithm problem, and $\mathsf{NP}^B$ might be the lowest natural complexity class they may fit in.

We remark that the results remain valid for finite fields of characteristic $p$.

1. Introduction

1.1. Randomness vs. mathematical intractability: a tradeoff


Arthur Merlin Games: A Randomized Proof System, and a Hierarchy of Complexity Classes

LÁSZLÓ BABAI
Eötvös University, Budapest, Hungary and University of Chicago, Chicago Illinois

AND

SHLOMO MORAN
Technion Haifa, Israel

Received June 24, 1986; revised August 3, 1987

One can view $\mathsf{NP}$ as the complexity class that captures the notion of efficient provability by classical (formal) proofs. We consider broader complexity classes (still "just above $\mathsf{NP}$"), in the hope to formalize the notion of efficient provability by overwhelming statistical evidence. Such a concept should combine the nondeterministic nature of (classical) proofs and the statistical nature of conclusions via Monte Carlo algorithms such as a Solovay-Strassen style "proof" of primality. To accomplish this goal, two randomized interactive proof systems have recently been offered independently by S. Goldwasser, S. Micali, and C. Rackoff (GMR system) and by L. Babai (Arthur-Merlin system) (in "Proceedings, 17th ACM Symp. Theory of Comput., Providence, RI, 1985," pp.

Par Schmid, Renate — https://opc.mfo.de/detail?photo_id=14372, CC BY-SA 2.0 de.
https://commons.wikimedia.org/w/index.php?curid=18096981
Arthur vs. Merlin games

❖ Imagine we would like to decide whether \( x \in L \)

❖ We ask Arthur — a mere mortal, who lives only for polynomial time

❖ Arthur can ask Merlin… a supernatural being able to give the answer to any problem (even non-computable)

❖ but Arthur does not trust Merlin…

Arthur: http://lusile17.1.u.pic.centerblog.net/273f716e.jpg
Imagine we would like to decide whether $x \in L$

We ask Arthur — a mere mortal, who lives only for polynomial time

Arthur can ask Merlin for a proof $y$ that $x$ is in $L$

now Arthur can check Merlin’s proof… provided $y$ has polynomial size
Arthur vs. Merlin games

- **INPUT:** $x$
- Merlin answers $y$
- We check whether $(x, y) \in D$ (for some $D$ in $\mathsf{P}$)
- The languages decided this way are just those in $\mathsf{NP}$.

Arthur: [Image](http://lusile17.1.u.pic.centerblog.net/273f716e.jpg)
The class AM

- Now Arthur can also draw (uniform) **random strings**
- INPUT: $x$
- Arthur draws $r$ at random and computes a question $q \leftarrow A(x,r)$
- ... and sends $x\#q\#r$ to Merlin
- Merlin answers $y$
- We check whether $x\#q\#r\#y \in D$ (for some $D$ in $\mathbb{P}$)
- Acceptance condition: if $x \in L$ then succeeds with high prob. if $x \notin L$ then fails with high prob.

... with a catch! (in fact, two)
The class AM, formally (1st try)

- $L$ is in AM iff there are:
  - a **poly time** Turing machine $A$
    (used by Arthur to compute questions $q \equiv A(x,r)$)
  - a function $M : \Sigma^* \to \Sigma^*$ producing **poly size** outputs
    (a Merlin map, not necessarily computable)
  - a **poly time** decidable language $D$
    such that:
    - if $x \in L$ then $\Pr_r(x \# q \# r \# y \in D) \geq 2/3$
    - if $x \notin L$ then $\Pr_r(x \# q \# r \# y \in D) \leq 1/3$
    - where $q \equiv A(x,r)$, $y \equiv M(x \# q \# r)$

First catch: when $x \notin L$, we should reject with high prob.
    even if Merlin is **dishonest**, namely **whatever** $y$ it plays

What **honest** Merlin plays, in order to make us accept when $x \in L$
The class AM, formally (2nd try)

- $L$ is in AM iff there are:
  — a poly time Turing machine $A$
  — a Merlin map $M : \Sigma^* \rightarrow \Sigma^*$ producing poly size outputs
  — a poly time decidable language $D$ such that:

- if $x \in L$ then $\Pr_r(x \# q \# r \# y \in D) \geq 2/3$
  where $q \equiv A(x,r)$, $y \equiv M(x \# q \# r)$

- if $x \notin L$ then $\forall$ Merlin map $M'$,
  $\Pr_r(x \# q \# r \# y \in D) \leq 1/3$
  where $q \equiv A(x,r)$, $y \equiv M'(x \# q \# r)$

Second (more benign) catch:
I do not know of any correct proof of error reduction in the literature;
and I do not know of any simple one.
The class AM, formally (final)

- $L$ is in AM iff $\forall$ polynomial $n \rightarrow g(n)$, there are:
  - a poly time Turing machine $A$
  - a Merlin map $M : \Sigma^* \rightarrow \Sigma^*$ producing poly size outputs
  - a poly time decidable language $D$ such that:

- if $x \in L$ then $\Pr_r(x \# q \# r \# y \in D) \geq 1 - 1/2g(n)$
  where $q \equiv A(x,r), y \equiv M(x \# q \# r)$

- if $x \notin L$ then $\forall$ Merlin map $M'$,
  $\Pr_r(x \# q \# r \# y \in D) \leq 1 / 2g(n)$
  where $q \equiv A(x,r), y \equiv M'(x \# q \# r)$
In general, for any word \( w \equiv a_1 a_2 \ldots a_k \in \{A, M\}^* \), there is a class \( w \) (note: boldface), of languages \( L \) such that \( \forall g, \exists A, M, D: \)

- If \( x \in L \) then \( \Pr_r(\text{prot}_w(M; x, r) \text{ accepts}) \geq 1 - 1/2^g(n) \)
- If \( x \not\in L \) then \( \forall M', \Pr_r(\text{prot}_w(M'; x, r) \text{ accepts}) \leq 1/2^g(n) \)

\[
\text{prot}_w(M; x, r_1 r_2 \ldots r_k):
\]
\[
\begin{align*}
\text{inp} &:= x \\
\text{for } j = 1 \ldots k: \\
&\quad \text{if } a_j = A \text{ then } (q_j := A(\text{inp}, r_j); \text{inp} := \text{inp} \# r_j \# q_j) \\
&\quad \text{else } (y_j := M(\text{inp}); \text{inp} := \text{inp} \# y_j) \\
\text{accept} &\text{ if } \text{inp} \in D, \text{ else } \text{reject}
\end{align*}
\]

Arthur’s turn. « draw \( r_j \) at random », compute question \( q_j \), add both to history \( \text{inp} \)

Merlin’s turn. find answer \( y_j \), add it to history \( \text{inp} \)
The Arthur–Merlin hierarchy: the low levels

- When \( w = \epsilon \) (\( k = 0 \)), \( \epsilon = ? \)

In general, for any word \( w = a_1a_2\ldots a_k \in \{A, M\}^* \), there is a class \( \mathcal{w} \) (note: boldface), of languages \( L \) such that \( \forall g, \exists A, M, D: \)

- If \( x \in L \) then \( \Pr_r(\text{prot}_w(M; x, r) \text{ accepts}) \geq 1 - 1/2^{|n|} \)
- If \( x \notin L \) then \( \forall M', \Pr_r(\text{prot}_w(M'; x, r) \text{ accepts}) \leq 1/2^{|n|} \)

\[
\text{prot}_w(M; x, r_1r_2\ldots r_k):
\begin{align*}
\text{inp} &= x \\
\text{for } j = 1 \ldots k: \\
&\quad\text{if } a_i = A \text{ then } (q_i := A(\text{inp},r_i); \text{inp} := \text{inp}\#r_i\#q_i) \\
&\quad\text{else } (y_i := M(\text{inp}); \text{inp} := \text{inp}\#y_i) \\
&\quad\text{accept if } \text{inp} \in D, \text{ else reject}
\end{align*}
\]
The Arthur-Merlin hierarchy: the low levels

- When \( w = \varepsilon (k=0) \): \( \varepsilon = \mathbf{P} \)
- When \( w = A \): \( A = ? \)

In general, for any word \( w = a_1a_2...a_k \in \{A, M\}^* \), there is a class \( w \) (note: boldface), of languages \( L \) such that \( \forall g, \exists A, M, D \):

- If \( x \in L \) then \( \Pr_r(\text{prot}_w(M; x, r) \text{ accepts}) \geq 1 - \frac{1}{2^{2^n}} \)
- If \( x \notin L \) then \( \forall M', \Pr_r(\text{prot}_w(M'; x, r) \text{ accepts}) \leq \frac{1}{2^{2^n}} \)

\[ \begin{array}{l}
\text{prot}_w(M; x, r_1r_2...r_k):
\text{inp} := x
\text{for } j = 1...k:
\quad \text{if } a_j = A \text{ then } (q_j := A(\text{inp}, r_j); \text{inp} := \text{inp}#r_j#q_j)
\quad \text{else } (y_j := M(\text{inp}); \text{inp} := \text{inp}#y_j)
\quad \text{accept if } \text{inp} \in D, \text{ else reject}
\end{array} \]
The Arthur-Merlin hierarchy: the low levels

- When $w=\varepsilon$ ($k=0$): $\varepsilon\in P$
- When $w=A$: $A=BPP$
- When $w=M$: $M=?$

In general, for any word $w = a_1a_2\ldots a_k \in \{A, M\}^*$, there is a class $\omega$ (note: boldface), of languages $L$ such that $\forall g, \exists A, M, D$:

- If $x \in L$ then $Pr_{\omega}(\text{prot}_\omega(M; x, r) \text{ accepts}) \geq 1-1/2^{\omega(n)}$
- If $x \notin L$ then $\forall M', \exists r, \forall(\text{prot}_\omega(M'; x, r) \text{ accepts}) \leq 1/2^{\omega(n)}$

$\text{prot}_\omega(M; x, r_1r_2\ldots r_k)$:

- $\text{inp} := x$
- For $j=1\ldots k$:
  - if $a_j=A$ then $(q_j := \mathcal{A}(\text{inp}, r_j); \text{inp} := \text{inp}\#r_j\#q_j)$
  - else $(y_j := M(\text{inp}); \text{inp} := \text{inp}\#y_j)$
- accept if $\text{inp} \in D$, else reject

Arthur’s turn. «draw $r_j$ at random», compute question $q_j$, add both to history $\text{inp}$

Merlin’s turn. find answer $y_j$, add it to history $\text{inp}$
The Arthur-Merlin hierarchy: the low levels

- When $w=\varepsilon \ (k=0)$: $\varepsilon = \mathbf{P}$
- When $w=A$: $A = \mathbf{BPP}$
- When $w=M$: $M = \mathbf{NP}$
- Then we have $\mathbf{MA}$, $\mathbf{AM}$, $\mathbf{AMAM} = \mathbf{AM}[2]$, $\mathbf{AM}[3]$, ..., $\mathbf{AM}[k]$, ...
Interactive proofs
Interactive proofs

(STOC’1985 aussi!)

The Knowledge Complexity of Interactive Proof-Systems

(Extended Abstract)

Shafl Goldwasser
MIT
Silvio Micali
MIT
Charles Rackoff
University of Toronto

1. Introduction

In the first part of the paper we introduce a new theorem-proving procedure, that is a new efficient method of communicating a proof. Any such method implies, directly or indirectly, a definition of proof. Our “proofs” are probabilistic in nature. On asking a statement, it is correct and right.

We propose to classify languages according to the amount of additional knowledge that must be released for proving membership in them.

Of particular interest is the case where this additional knowledge is essentially 0 and we show that it is possible to interactively prove that a number is quadratic residue.

Moreover, we exhibit the proof that adding information to increase the amount falsified in order

By Rguillou228 - Own work, CC BY-SA 4.0,
https://commons.wikimedia.org/w/index.php?curid=74039300

https://alchetron.com/cdn/charles-rackoff-3aa39129-7251-4443-9d07-4e01fcf9c-resize-750.jpeg
Interactive proofs

- Note that in Arthur-Merlin games, Arthur must communicate not just $q$, but also its random bits $r$ to Merlin.

- In interactive proofs, Arthur only gives out $q$, and may therefore keep $r$ secret (but is not forced too).
The class $\text{AM} \mapsto \text{IP}[1]$}

- $L$ is in $\text{IP}[1]$ iff $\forall$ polynomial $n \mapsto g(n)$, there are:
  - a poly time Turing machine $\mathcal{A}$
  - a Merlin map $M : \Sigma^* \to \Sigma^*$ producing poly size outputs
  - a poly time decidable language $D$ such that:

- if $x \in L$ then $\Pr_r(x \# q \# r \# y \in D) \geq 1 - 1/2^{g(n)}$
  where $q \triangleq \mathcal{A}(x, r)$, $y \triangleq M(x \# q \# r)$

- if $x \notin L$ then $\forall$ Merlin map $M'$,
  $\Pr_r(x \# q \# r \# y \in D) \leq 1/2^{g(n)}$
  where $q \triangleq \mathcal{A}(x, r)$, $y \triangleq M'(x \# q \# r)$


Note that $r$ still takes part in the final decision (and in Arthur’s computations, of course)
Example: Graph Isomorphism

- Let $V = \{1, \ldots, N\}$ set of vertices,
  $G_N \overset{\text{def}}{=} \text{directed graphs on } V$,
  $S_N \overset{\text{def}}{=} \text{group of permutations of } V$.

- $S_N$ acts on $G_N$ by: $\forall \pi \in S_N$, $\forall G = (V,E) \in G_N$,
  $\pi.G \overset{\text{def}}{=} (V, \{ (\pi(u), \pi(v)) \mid (u,v) \in E \})$

- Two graphs $G_1 = (V, E_1)$, $G_2 = (V, E_2)$ (with the same $V$) are isomorphic ($G_1 \equiv G_2$) iff $\exists \pi \in S_N$, $\pi.G_1 = G_2$. 
Example: Graph Isomorphism

- **Graph isomorphism:**
  INPUT: 2 graphs $G_1 = (V, E_1)$, $G_2 = (V, E_2)$ (with the same $V$)
  QUESTION: are $G_1$, $G_2$ isomorphic?

- Clearly in **NP**
- Not known to be in **P**, nor **NP-complete**...
- We will show, using results on MA, AM, IP[1], etc. that it is **not NP-complete** (unless PH collapses)

(This is only the beginning; Babai gave a super polynomial time algo for GI in 2015; you need to understand first everything in the course to have a hope of understanding it!)
Example: Graph **Non-Isomorphism**

- **GNI** $\triangleq$ complement of **GI**: in **coNP**, not known to be in **P** or **coNP**-complete

- **Prop. GNI** is in **IP**[1].

- **Algorithm.**
  - Arthur draws $i \in \{1,2\}$, $\pi \in S_N$ at random uniformly, sends $q \triangleq \pi.G_i$
  - Merlin answers $j \in \{1,2\}$
  - We accept if $i=j$, reject otherwise.
Prop. GNI is in IP[1].

Proof.
— If \((G_1, G_2) \in \text{GNI}\), there is a unique \(j \in \{1,2\}\) such that \(G_j \equiv \pi.G_i\) (viz., \(i\))
Merlin plays that \(j\), forcing acceptance (always).

— Arthur draws \(i \in \{1,2\}\), \(\pi \in \mathcal{S}_N\) at random uniformly, sends \(q \equiv \pi.G_i\)
— Merlin answers \(j \in \{1,2\}\)
— We accept if \(i=j\), reject otherwise.
Prop. GNI is in IP[1].

Proof.

— If \((G_1, G_2) \notin \text{GNI}\),
then \(G_1 \equiv G_2 \equiv \pi.G_i\) (viz., \(i\))
and Merlin has no information about \(i\)
Whatever Merlin plays, \(\Pr(\text{acceptance})=1/2\).

That is in fact irrelevant to the proof.
But that shows that GNI has a zero-knowledge proof!
Prop. GNI is in $\text{IP}[1]$. 

Error too big ($1/2$).

⇒ Repeat experiments (à la $\text{RP}$), but in parallel.

— Arthur draws $i \in \{1,2\}$, $\pi \in \mathbb{S}_N$ at random uniformly, sends $q = \pi.G_i$
— Merlin answers $j \in \{1,2\}$
— We accept if $i=j$, reject otherwise.

Error $1/2$ now (and still no error if $(G_1, G_2) \in \text{GNI}$).
GNI is in AM

- We will see later that GNI is in AM.
- This is a better result, since AM ⊆ IP[1]
  (Any AM game can be simulated as an IP[1] game
   where Arthur sends both q and r as its question!)
- In fact, AM = IP[1]… but this is a pretty hard result,
  due to Goldwasser and Sipser.
- Meanwhile, let us return to the study of MA, AM, etc.
Other equivalent definitions of AM
1. BP·NP
The **BP · operator**

- **Generalizing BPP.**
  For any class $C$, the class **BP · C**:

  - A language $L$ is in **BP · C** iff there is a language $D$ in $C$, and a poly time TM $M$ such that for every input $x$ (of size $n$):
    - if $x \in L$ then $\Pr_r [M(x,r) \in D] \geq 2/3$
    - if $x \notin L$ then $\Pr_r [M(x,r) \in D] \leq 1/3$.

- In particular, **BP · P = BPP**.
Error reduction: democracy

- As for BPP, we can reduce the error in $\text{BP} \cdot C$ from $1/3$ to $1/2^{g(n)}$ for any polynomial $g(n)$

- ... provided that $C$ is democratic
  (non-standard name; obtained through a vote in class a few years ago)

- Defn. $C$ is democratic iff for every $L \in C$,
  $\{w_1\# ... \# w_k \mid \text{a majority of words } w_i \text{ is in } L\}$ is in $C$.  

A language $L$ is in $\text{BP} \cdot C$ iff there is a language $D$ in $C$, and a poly time TM $M$ such that for every input $x$ (of size $n$):
- if $x \in L$ then $\Pr_r [M(x,r) \in D] \geq 2/3$
- if $x \notin L$ then $\Pr_r [M(x,r) \in D] \leq 1/3$. 

Error reduction through democracy

- Let $L \in \text{BP} \cdot C$, with $D$ as here →
- Let $D' \overset{\text{def}}{=} \{w_1 \# \ldots \# w_k \mid \text{a majority of words } w_i \text{ is in } D\}$
  - $D'$ is again in $C$
- It suffices to decide whether $M(x, r_1) \# \ldots \# M(x, r_{36g(n)\log 2}) \in D'$ (in poly-time)
- Then error is $\leq 1/2^{g(n)}$ (Chernoff!)
We have proved:

**Thm.** Let $C$ be democratic, and $g(n)$ be a polynomial. Then $\text{BP} \cdot C$, is also the class of languages $L$ such that [...]:

- if $x \in L$ then
  $$\Pr_r [M(x,r) \in D] \geq 1 - 1/2g(n)$$
- if $x \notin L$ then
  $$\Pr_r [M(x,r) \in D] \leq 1/2g(n).$$

**Defn.** $C$ is democratic iff for every $L \in C$, \{ $w_1#\ldots#w_k$ | a majority of words $w_i$ is in $L$ \} is in $C$. 

A language $L$ is in $\text{BP} \cdot C$ iff there is a language $D$ in $C$, and a poly time TM $M$, such that for every input $x$ (of size $n$):

- if $x \in L$ then $\Pr_r [M(x,r) \in D] \geq 2/3$
- if $x \notin L$ then $\Pr_r [M(x,r) \in D] \leq 1/3$. 

**Application to voting (4/4)**

- Assume that $\Pr_r (M(x,r) \text{ err}) \leq 1/3$.
  - *how large* should $N$ be so that the probability $P$ that more than $1/2$ of $N$ votes $M(x,r)$ is $\leq 1/2g(n)$?
  - **Answer:** at least $36 \log_2 g(n)$
- $\Pr(\#N=\log_2 g(n)) \leq 1/2g(n)$ if $\exp(-N/36) \leq 1/2g(n)$
- *Proof.* $\exp(-N/36) \leq 1/2g(n)$ iff $-N/36 \leq -g(n)\log_2 2$
Examples of democratic classes

- Fact. \( P \) is democratic. (Easy.)
- Prop. \( NP \) is democratic.
- No, we cannot check whether each \( w_i \) is in \( L \), because if that check fails, then the whole computation fails.
- Instead, we guess a subset \( I \) of indices of \( \geq k/2 \) elements, and we check that \( \forall i \in I, w_i \) is in \( L \). \( \square \)

Defn. \( C \) is democratic iff for every \( L \in C \), \( \{ w_1 \# \cdots \# w_k \mid \text{a majority of words } w_i \text{ is in } L \} \) is in \( C \).
BP · NP has error reduction

Thm. \( L \in \text{BP} \cdot \text{NP} \) iff \( \forall \) poly \( g \),
\( \exists D \in \text{NP}, \) poly time TM \( M / \)
— if \( x \in L \) then
\( \Pr_r \left[ M(x, r) \in D \right] \geq 1 - \frac{1}{2^{g(n)}} \)
— if \( x \not\in L \) then
\( \Pr_r \left[ M(x, r) \in D \right] \leq \frac{1}{2^{g(n)}}. \)
Thm (Prop. 3.5). $\text{AM} = \text{BP} \cdot \text{NP}$.

Proof (1/4). Let $L \in \text{AM}$, as here:

Let $D' \overset{\text{def}}{=} \{ x \# r \mid \exists y, x \# q \# r \# y \in D, \text{ where } q \overset{\text{def}}{=} A(x, r) \}$: $D'$ is in $\text{NP}$.

If $x \in L$, $\Pr_r(x \# r \in D')$

$$= \Pr_r(\exists y, x \# q \# r \# y \in D, \text{ where } q \overset{\text{def}}{=} A(x, r))$$

$$\geq \Pr_r(x \# q \# r \# y \in D, \text{ where } q \overset{\text{def}}{=} A(x, r), y \overset{\text{def}}{=} M(x \# q \# r))$$

$$\geq 1 - 1/2^{g(n)}$$

because $\exists y, P(y)$ is implied by $P(M(x \# q \# r))$
Thm (Prop. 3.5). AM = BP · NP.

Proof (2/4). Let $L \in AM$, as here:

Let $D' \equiv \{x\#r \mid \exists y, x\#q\#r\#y \in D, \text{ where } q \triangleq A(x,r)\}$: $D'$ is in NP.

- If $x \notin L$, then let $M'(x\#q\#r) \triangleq \text{best}$ of Merlin's responses, i.e., some $y$ such that $x\#q\#r\#y \in D$ if one exists

- Then $Pr_r(x\#r \in D')$
  \[= Pr_r(\exists y, x\#q\#r\#y \in D, \text{ where } q \triangleq A(x,r))\]
  \[\leq Pr_r(x\#q\#r\#M'(x\#q\#r) \in D, \text{ where } q \triangleq A(x,r))\]
  \[\leq 1/2^{g(n)}\]

because $M'$ is best:

$(\exists y, x\#q\#r\#y \in D) \Rightarrow x\#q\#r\#M'(x\#q\#r) \in D$
Thm (Prop. 3.5). \( AM = BP \cdot NP \).

Proof (3/4). Let \( L \in BP \cdot NP \), as here:
Let \( D \equiv \{ q \mid \exists y, q#y \in D' \} \), with \( D' \in P \),
\( D'' \equiv \{ x#q#r#y \mid q#y \in D' \} \): in \( P \).

Let \( A(x,r) \equiv M(x,r) \), and \( M(x#q#r) \equiv \) some \( y \) such that \( q#y \in D' \) if one exists,

If \( x \in L \), \( Pr_r(x#q#r#y \in D'', \text{ where } q \equiv A(x,r), y \equiv M(x#q#r)) \)
\( = Pr_r(q#y \in D', \text{ where } q \equiv A(x,r), y \equiv M(x#q#r)) \)
\( \geq Pr_r(\exists y, q#y \in D', \text{ where } q \equiv A(x,r)) \)
\( = Pr_r(M(x,r) \in D) \geq 1 - 1/2^{g(n)} \)

because \( M \) is best:
\( (\exists y, q#y \in D') \Rightarrow q#M(x#q#r) \in D' \)
Thm (Prop. 3.5). AM = BP·NP.

Proof (4/4). Let \( L \in \text{BP} \cdot \text{NP} \), as here:

Let \( D \overset{\Delta}{=} \{ q \mid \exists y, q\#y \in D' \} \), with \( D' \in \text{P} \),
\( D'' \overset{\Delta}{=} \{ x\#q\#r\#y \mid q\#y \in D' \} \): in \( \text{P} \).

Let \( \mathcal{A}(x,r) \equiv M(x,r) \)

If \( x \notin L \), for any \( M' \), \( \Pr_r \left( x\#q\#r\#y \in D'' \right) \), where \( q \equiv \mathcal{A}(x,r), y \equiv M'(x\#q\#r) \)
\[ = \Pr_r \left( q\#y \in D' \right), \text{ where } q \equiv \mathcal{A}(x,r), y \equiv M'(x\#q\#r) \]
\[ \leq \Pr_r \left( \exists y, q\#y \in D' \right), \text{ where } q \equiv \mathcal{A}(x,r) \]
\[ = \Pr_r \left( M(x,r) \in D \right) \leq 1/2^{g(n)} \]

because \( \exists y, P(y) \) is implied by \( P(M'(x\#q\#r)) \)
Other equivalent definitions of AM
2. Extended quantifiers
Lazy Arthur

- Let us say that Arthur is lazy if Arthur does not bother to compute any question: \( A(x,r) = \varepsilon \)

- **Prop (Lemma 3.8).** For every word \( w \in \{A, M\}^* \), the class \( w_{\text{lazy}} \) when Arthur is constrained to be lazy is equal to the class \( w \).

- **Proof.** See lecture notes. Idea: Merlin is so powerful he can reconstruct Arthur’s questions without Arthur’s help. □

- \[ \text{prot}_{w_{\text{lazy}}}(M; x, r_1 r_2 \ldots r_k): \]
  \[ \text{inp} := x \]
  \[ \text{for} \ j=1\ldots k: \]
    \[ \text{if} \ a_j=A \text{ then} (q_j := A(\text{inp}, r_j); \text{inp} := \text{inp}\#r_j\#q_j) \]
    \[ \text{else} (y_j := M(\text{inp}); \text{inp} := \text{inp}\#y_j) \]
  \[ \text{accept} \text{ if } \text{inp} \in D, \text{ else reject} \]
A logical approach

- Model both Arthur and Merlin as quantifiers (over $r, y$)
- ... for « predicates » with values in $[0, 1]$ over finite sets

**Arthur** (expectation):

$$E_{r \in R, F(r)} \overset{\text{def}}{=} \sum_{r \in R} F(r) / \text{card } R$$

**Merlin** (maximize):

$$\exists y \in Y, F(y) \overset{\text{def}}{=} \max_{y \in Y} F(y)$$

(Note: if $F$ takes its values in $\{0,1\}$, this is really the existential quantifier...?)
A small catch

- The notations $E$, $\exists$ are practical, e.g.:
  - $(\exists y \in Y, F(y)) \geq a$  iff  there is a $y \in Y$ such that $F(y) \geq a$
  - But beware that
  - $(\exists y \in Y, F(y)) \leq a$  iff  for every $y \in Y$, $F(y) \leq a$.  

<table>
<thead>
<tr>
<th>Arthur (expectation):</th>
<th>Merlin (maximize):</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathbb{E}<em>{r \in R, F(r)} = \frac{\sum</em>{r \in R} F(r)}{\text{card } R}$</td>
<td>$\exists y \in Y, F(y) \leq \max_{y \in Y} F(y)$</td>
</tr>
</tbody>
</table>
Prop. \( \mathbb{E}_r \in R, \exists y \in Y, F(r,y) \)
\[= \exists f : R \to Y, \mathbb{E}_r \in R, F(r,f(r)) \]

Proof (1/2).
Let \( f(r) \overset{\text{def}}{=} \text{best } y \), viz. some \( y \) that maximizes \( F(r,y) \)
Then \( \exists y \in Y, F(r,y) = F(r,f(r)) \)

Take expectations:
\[ \mathbb{E}_r \in R, \exists y \in Y, F(r,y) = \mathbb{E}_r \in R, F(r,f(r)) \]
\[\ldots \leq \max_{f : R \to Y} \mathbb{E}_r \in R, F(r,f(r)) \]
Prop. $\forall r \in R, \exists y \in Y, F(r,y) = \exists f : R \rightarrow Y, \forall r \in R, F(r,f(r))$

Proof (2/2).
For every $f$, $F(r,f(r)) \leq \max_{y \in Y} F(r,y)$

Take expectations:
$\forall r \in R, F(r,f(r)) \leq \mathbb{E}r \in R, \max_{y \in Y} F(r,y)$

Now take max over $f$. □
« Skolemization »: an example

- $E_{r_1}, \exists y_1, E_{r_2}, \exists y_2, F(x, r_1, y_1, r_2, y_2)$
- $= \exists f_1, E_{r_1}, E_{r_2}, \exists y_2, F(x, r_1, y_1, r_2, y_2)$
  where $y_1 \overset{\text{def}}{=} f_1(r_1)$
- $= \exists f_1, f_2, E_{r_1}, r_2, F(x, r_1, y_1, r_2, y_2)$
  where $y_1 \overset{\text{def}}{=} f_1(r_1), y_2 \overset{\text{def}}{=} f_2(r_1, r_2)$

Arthur (expectation):
$E_{r} \in R$, $F(r) = \sum_{r \in R} F(r) / \text{card } R$

Merlin (maximize):
$\exists y \in Y$, $F(y) = \max_{y \in Y} F(y)$

Prop.
$E_{r} \in R$, $\exists y \in Y$, $F(r, y) = \exists f : R \to Y$, $E_{r} \in R$, $F(r, f(r))$
Let $F$ be $\{0,1\}$-valued (not $[0,1]$) i.e., a **predicate**

Recall that $\exists y \in Y, F(y)$ (=max) is then the existential quantifier (... and is therefore $\{0,1\}$-valued)

Also, $E_r, F(r) = \Pr_r(F(r)=1)$

(« expectation of a predicate = its probability of occurring »)
A-M as E-∃ formulae

- **Prop (3.10).** \( L \in \text{AMAM} \) iff for every polynomial \( g(n) \), there is a poly time predicate \( P \) /— if \( x \in L \), then \( G(x) \geq 1 - 1/2g(n) \)— if \( x \notin L \) then \( G(x) \leq 1/2g(n) \)

  where \( G(x) \triangleq Er_1, \exists y_1, Er_2, \exists y_2, P(x,r_1,y_1,r_2,y_2) \)

- **Proof (1/5).** \( G(x) = \exists f_1, f_2, Er_1, r_2, P(x,r_1,y_1,r_2,y_2) \)

  where \( y_1 \triangleq f_1(r_1), y_2 \triangleq f_2(r_1,r_2) \) « skolemization »

- **Hence** \( G(x) = \exists f_1, f_2, Pr_{r_1, r_2}(x\#r_1\#y_1\#r_2\#y_2 \in D) \)

  where \( D \triangleq \{ x\#r_1\#y_1\#r_2\#y_2 \mid P(x,r_1,y_1,r_2,y_2)=1 \} \) (note: \( D \in \mathbb{P} \))
Proof (2/5). If \( L \in \text{AMAM} \) with Merlin map \( M \) and a lazy Arthur,

- if \( x \in L \) then let \( f_1(r_1) \stackrel{\text{def}}{=} M(x \# r_1) \) (in short, \( y_1 \))

\[
f_2(r_1,r_2) \stackrel{\text{def}}{=} M(x \# r_1 \# f_1(r_1) \# r_2)
\]

Then \( G(x) = \exists f_1, f_2, \Pr_{r_1, r_2}(x \# r_1 \# y_1 \# r_2 \# y_2 \in D) \geq \Pr_{r_1, r_2}(x \# r_1 \# y_1 \# r_2 \# y_2 \in D) \geq 1 - 1/2^g(n) \)
Proof (3/5). If \( L \in \text{AMAM} \) with Merlin map \( M \) and a lazy Arthur,

- if \( x \notin L \) then for all maps \( f_1, f_2, \)
  let \( M'(x \# r_1) \overset{\text{def}}{=} f_1(r_1), M'(x \# r_1 \# y_1 \# r_2) \overset{\text{def}}{=} f_2(r_1, r_2) \)
  and \( M' \) of anything else be arbitrary (e.g., \( \varepsilon \))

- Then \( G(x) \leq \Pr_{r_1}(x \# r_1 \# y_1 \# r_2 \# y_2 \in D) \leq 1/2^{g(n)} \)
  where \( y_1 \overset{\text{def}}{=} M'(x \# r_1), y_2 \overset{\text{def}}{=} M'(x \# r_1 \# y_1 \# r_2) \)
Proof (4/5). If $L$ is as here →

for each $x \in L$ there are maps $f_1, f_2$
such that

$$\Pr_{r_1, r_2}(x \# r_1 \# y_1 \# r_2 \# y_2 \in D) \geq 1-1/2^{g(n)}$$

where $y_1 \equiv f_1(r_1), y_2 \equiv f_2(r_1, r_2)$

Let $M(x \# r_1) \equiv f_1(r_1), M(x \# r_1 \# y_1 \# r_2) \equiv f_1(r_1, r_2)$, else arbitrary

If $x \in L$ then $\Pr_r(x \# r_1 \# y_1 \# r_2 \# y_2 \in D) \ (y_1 \equiv M(x \# r_1), y_2 \equiv M(x \# r_1 \# y_1 \# r_2))$

$$= \Pr_{r_1, r_2}(x \# r_1 \# y_1 \# r_2 \# y_2 \in D) \ (y_1 \equiv f_1(r_1), y_2 \equiv f_2(r_1, r_2))$$

$$\geq 1-1/2^{g(n)}$$
Proof (5/5). If $L$ is as here

- If $x \notin L$ then
  for every Merlin map $M'$,
  \[
  f_1(r_1) \overset{\text{def}}{=} M'(x \# r_1) \quad \text{(in short, $y_1$)}
  \]
  \[
  f_2(r_1, r_2) \overset{\text{def}}{=} M'(x \# r_1 \# f_1(r_1) \# r_2) \quad (y_2)
  \]

- Then $\Pr_r(x \# r_1 \# y_1 \# r_2 \# y_2 \in D) \ (y_1 \overset{\text{def}}{=} M'(x \# r_1), y_2 \overset{\text{def}}{=} M'(x \# r_1 \# y_1 \# r_2))$
  \[
  = \Pr_{r_1, r_2}(x \# r_1 \# y_1 \# r_2 \# y_2 \in D) \ (y_1 \overset{\text{def}}{=} f_1(r_1), y_2 \overset{\text{def}}{=} f_2(r_1, r_2))
  \]
  \[
  \leq G(x) \leq 1 / 2^g(n). \quad \square
  \]
Next time...
The Arthur-Merlin hierarchy collapses!

- We will see that the whole Arthur-Merlin hierarchy looks like this!

\[
\begin{align*}
&\subseteq \quad \subseteq \quad \subseteq \\
&P \quad \subseteq \quad \subseteq \quad \subseteq \\
&NP \quad \subseteq \quad \subseteq \quad \subseteq \\
&MA \quad \subseteq \quad \subseteq \quad \subseteq \\
&AM \quad \subseteq \quad \subseteq \quad \subseteq \\
&\text{(All other classes w equal to AM)}
\end{align*}
\]