CS 121: Lecture 25
Randomized Algorithms 2

Adam Hesterberg

https://madhu.seas.Harvard.edu/courses/Fall2020

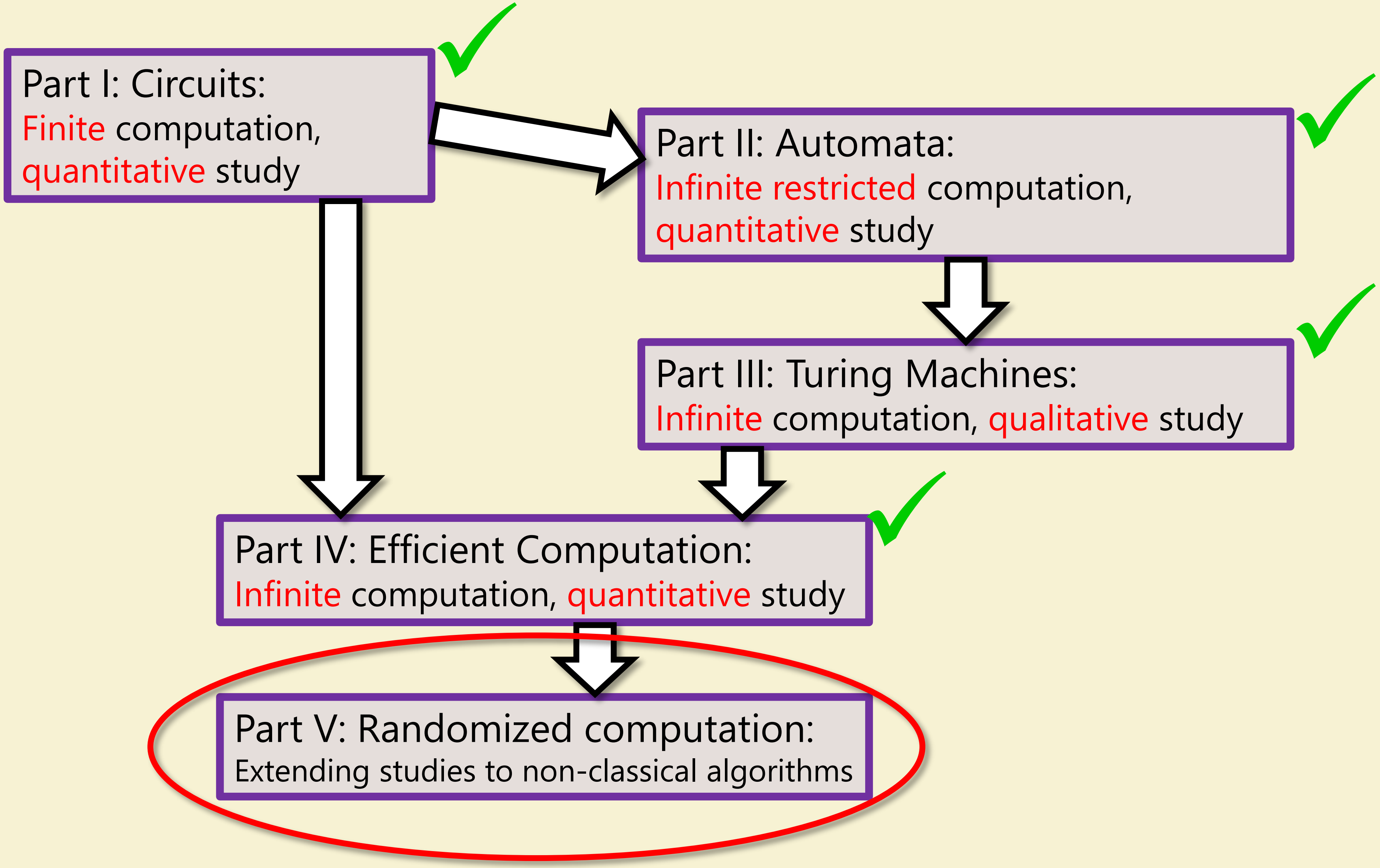
Book: https://introtcs.org

How to contact us
- The whole staff (faster response): CS 121 Piazza
- Only the course heads (slower): cs121.fall2020.course.heads@gmail.com

Announcements:

- Q survey open
- Last Sections this week (through Friday)

Where we are:


Part I: Circuits:
Finite computation, quantitative study
Part II: Automata:
Infinite restricted computation, quantitative study
Part III: Turing Machines:
Infinite computation, qualitative study
Part IV: Efficient Computation:
Infinite computation, quantitative study
Part V: Randomized computation:
Extending studies to non-classical algorithms

Last Lecture

Randomized algorithm ALG computes F if for every input x

$$\Pr[\,ALG(x) = F(x)] \geq \frac{2}{3}$$

Probability over the randomness of the algorithm, not the input

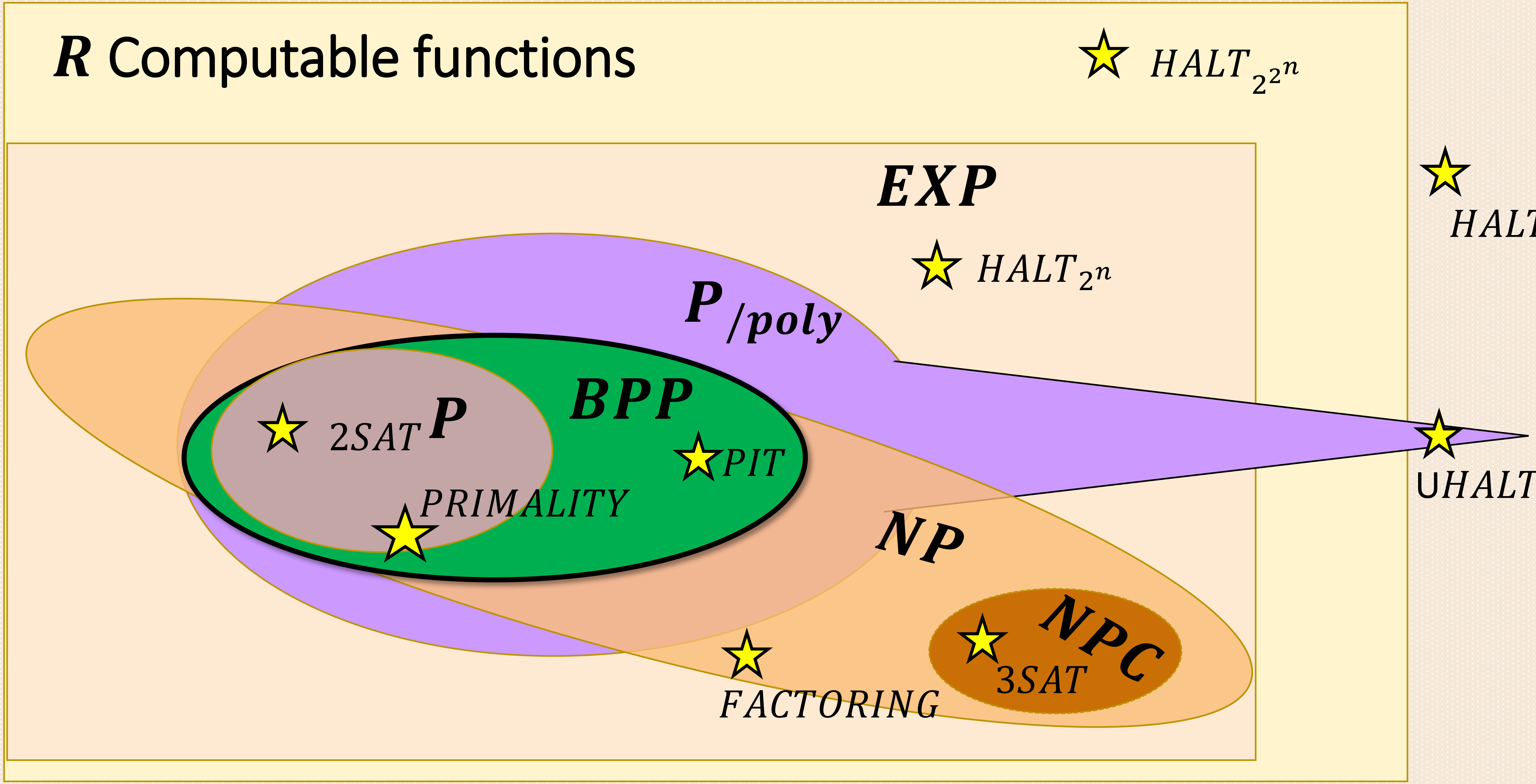
The constant $2/3$ is arbitrary – can be replaced by 0.51, 0.99, even $1 - 2^{-n}$. Not by $1/2$.

BPP: {Boolean functions computable by some randomized algorithm}

Polynomial Identity Testing: in BPP, not known if in P.

½-approx to Max Cut: ~~in BPP.~~ Randomized Alg.

All functions $F: \{0,1\}^* \to \{0,1\}$



Today

- $P \subseteq BPP \subseteq EXP$
- $BPP \subseteq P_{/poly}$
  - Proof uses success amplification via the Chernoff bond
- NP vs BPP: Unknown, but *Sipser-Gaacs-Lautemann Theorem:*
  - If $P = NP$ then $BPP = P$
  - BPP contained in a class like, and not much larger than, NP.

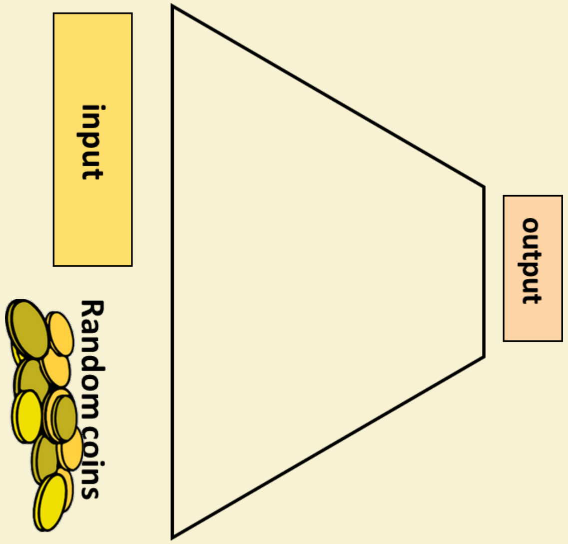
Gács G\'{a}cs ~~Gaacs~~

$P \subseteq BPP \subseteq EXP$

Def 2: $F:\{0,1\}^* \to \{0,1\}$ is in BPP if $\exists$ poly-time deterministic algorithm A , poly $q(n)$ s.t. $\forall n\ \forall x \in \{0,1\}^n$

$$\Pr_{r\sim\{0,1\}^{q(n)}}[A(x;r) = F(x)] \geq \frac{2}{3}$$

possible $q(n)$: $2^{q(n)} \in 2^{n^{O(1)}}$



Q: Prove that $P \subseteq BPP$ A: Ignore randomness

Q: Prove that $BPP \subseteq EXP$ A: Try all possible coin flip results

$BPP \subseteq P_{/poly}$ outline in words

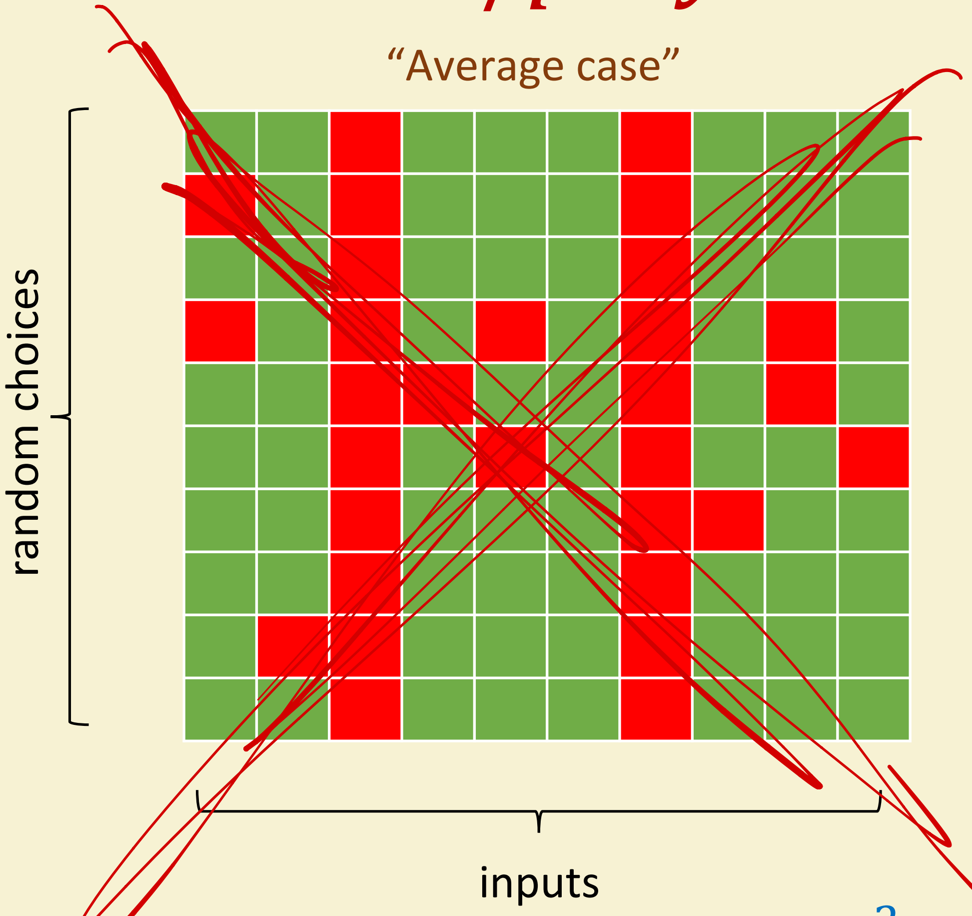
Def: $F: \{0,1\}^* \rightarrow \{0,1\}$ is in BPP if $\exists$ poly-time deterministic algorithm A such that $\forall n$, given a **random** poly-size advice string $q(n)$, $\forall x \in \{0,1\}^n$, A decides $\mathrm{F}(x)$ right, $p > 2/3$.

Def: $F: \{0,1\}^* \rightarrow \{0,1\}$ is in $P/poly$ if $\exists$ poly-time deterministic algorithm A such that $\forall n$, given a **fixed** poly-size advice string $q(n)$, $\forall x \in \{0,1\}^n$, A decides $\mathrm{F}(x)$ right.

Proof idea: Amplify the success probability so much that one $q(n)$ works for every input.

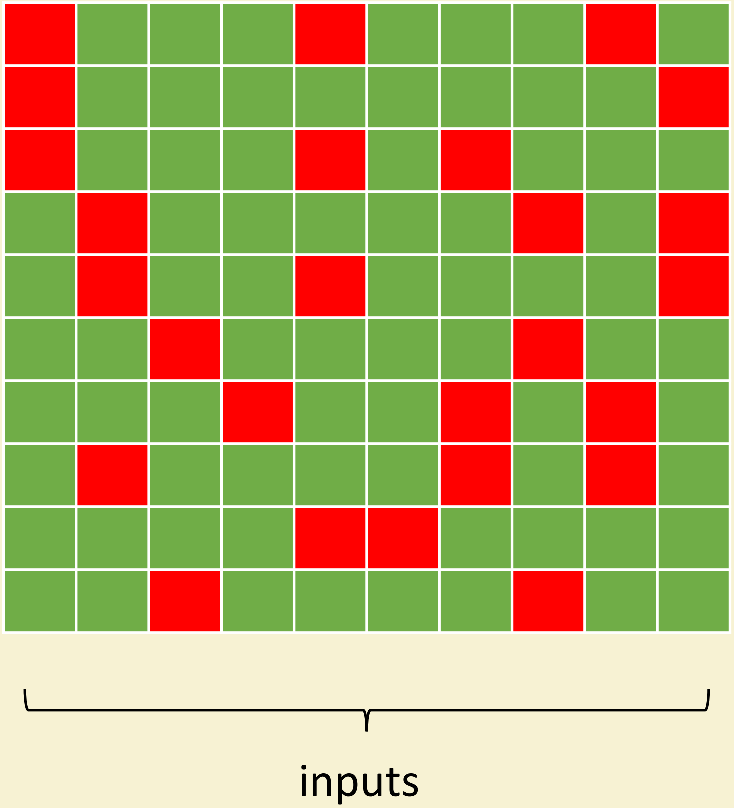
$BPP \subseteq P_{/poly}$ outline in pictures

"Average case"



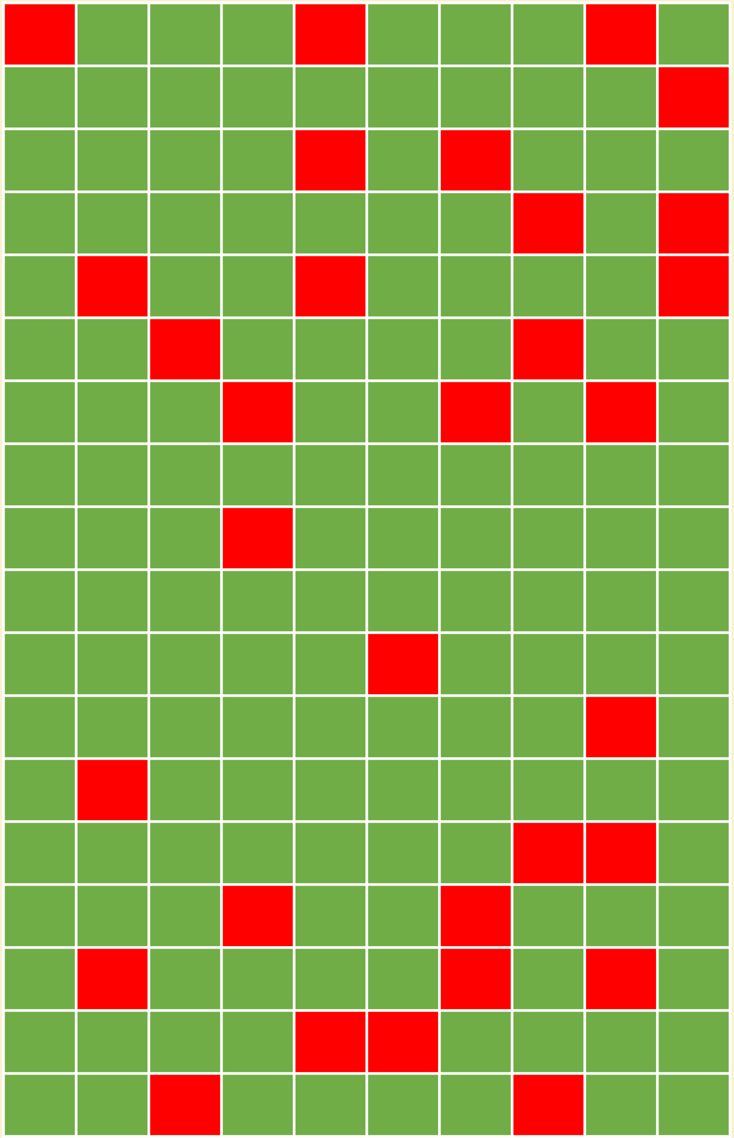
$$\Pr_{x\sim\{0,1\}^n, r\sim\{0,1\}^m}[A(x;r) = F(x)] \geq \frac{2}{3}$$

BPP



$$\forall_{x\in\{0,1\}^n} \Pr_{r\sim\{0,1\}^m}[A(x;r) = F(x)] \geq \frac{2}{3}$$

$BPP, success\ amplified$



Amplification for 2-sided error

Thm: If $F \in BPP$ then $\exists$poly-time algorithm B , poly $q(n)$ s.t. $\forall n\, \forall x \in \{0,1\}^n$

$$\Pr_{r\sim\{0,1\}^{q(n)}}[B(x;r) = F(x)] \geq 1 - 2^{-n^2}$$

Generally: Can amplify success from $\frac{1}{2} + \frac{1}{p(n)}$ to $1 - 2^{-r(n)}$ for all polys p, r

Chernoff Bound: Let $X_0, \ldots, X_{n-1}$ i.i.d. r.v.'s with $X_i \in [0,1]$.
Then if $X = X_0 + \cdots + X_{n-1}$ and $p = \mathbb{E}[X]$, for every $\epsilon > 0$,

$$\boxed{\Pr[\,|X - np| > \epsilon n] < 2^{1-(2\lg e)\epsilon^2 \cdot n}}$$

Thm: If $F \in BPP$ then $\exists$poly-time algorithm B , poly $q(n)$ s.t. $\forall n \, \forall x \in \{0,1\}^n$

$$\Pr_{r \sim \{0,1\}^{q(n)}}[B(x;r) = F(x)] \geq 1 - 2^{-n^2}$$

Proof: Suppose $\Pr[\, A(x;r) = F(x)] \geq 2/3$.

Idea: B will run A $1000n^2$ times and return majority vote.

Define $X_i = \begin{cases} 1, & A(x;r_i) = F(x) \\ 0, & A(x;r_i) \neq F(x) \end{cases}$

$X_1, \dots, X_{1000n^2}$ i.i.d with $\mathbb{E}[X_i] \geq 2/3$

E: $\frac{2}{3}$, diff: $\frac{1}{6}$

By Chernoff, $\Pr\left[\frac{1}{1000n^2}\sum_i X_i < 0.5\right] < 2^{1-\frac{2 \lg e}{36} \cdot 1000n^2} < 2^{-n^2}$

Algorithm B

Input: x

for $i = 1 \dots 1000n^2$:

$\quad r_i \sim \{0,1\}^m$

$\quad y_i \leftarrow A(x;r_i)$

return $Maj(y_1, \dots, y_{1000n^2})$

$\Pr[\, |X - np| > \epsilon n] < 2^{1-(2 \lg e)\epsilon^2 \cdot n}$

$\blacksquare$

$BPP \subseteq P_{/poly}$

If $F \in BPP$ then by amplification $\exists$ poly time algorithm A s.t.

$$\Pr_{r \sim \{0,1\}^m}[A(x;r) \neq F(x)] < 2^{-n}$$

Let $M = 2^m$ be # of random choices

Let $N = 2^n$ be # of inputs

Every column has $< \frac{M}{N}$ "reds"

$\Rightarrow$ # reds < # rows

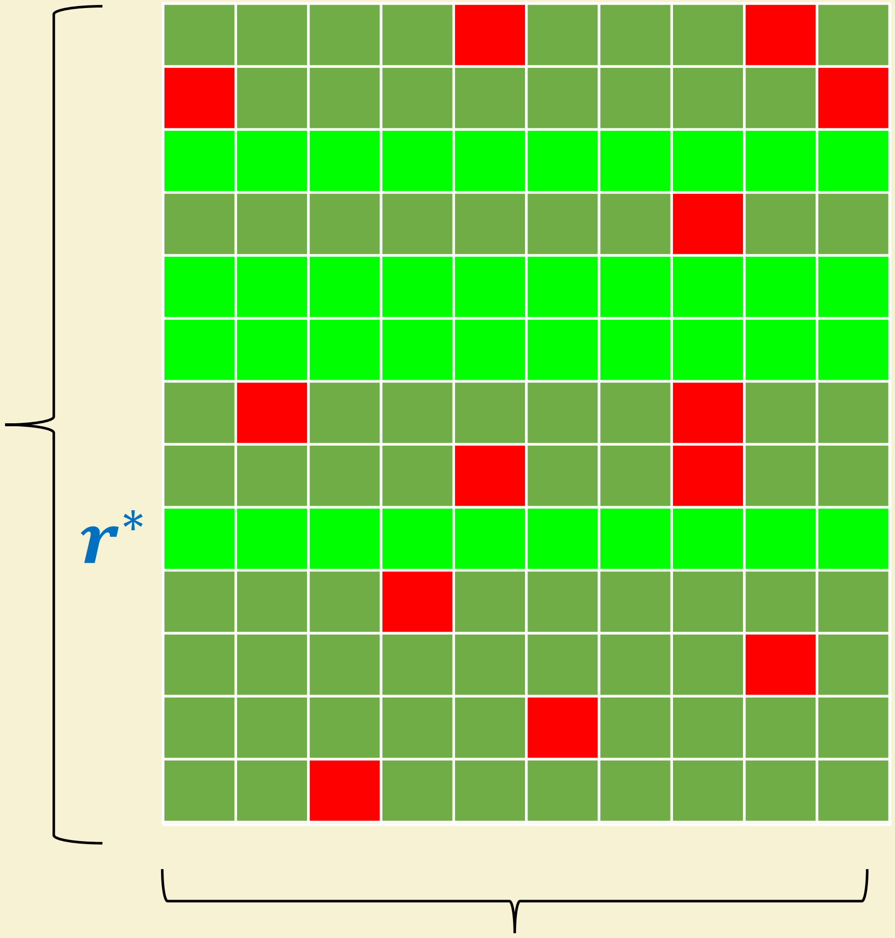
$\Rightarrow$ must be rows with no reds!

A "good choice of randomness" r^* s.t.

$$\forall x \in \{0,1\}^n \; A(x;r^*) = F(x)$$

Use r^* as the P/poly advice string. $\blacksquare$

$M = 2^m$ possible random choices

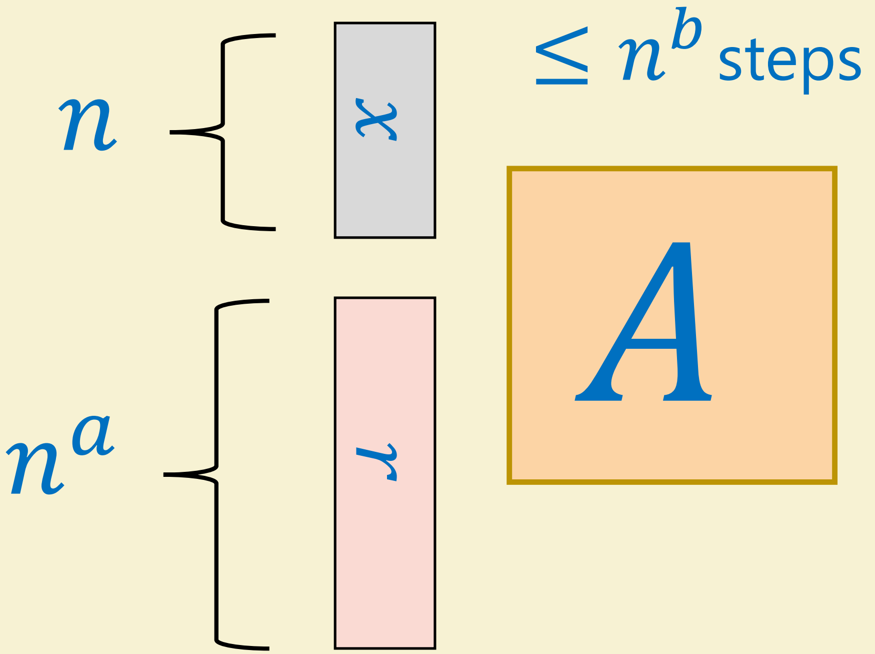


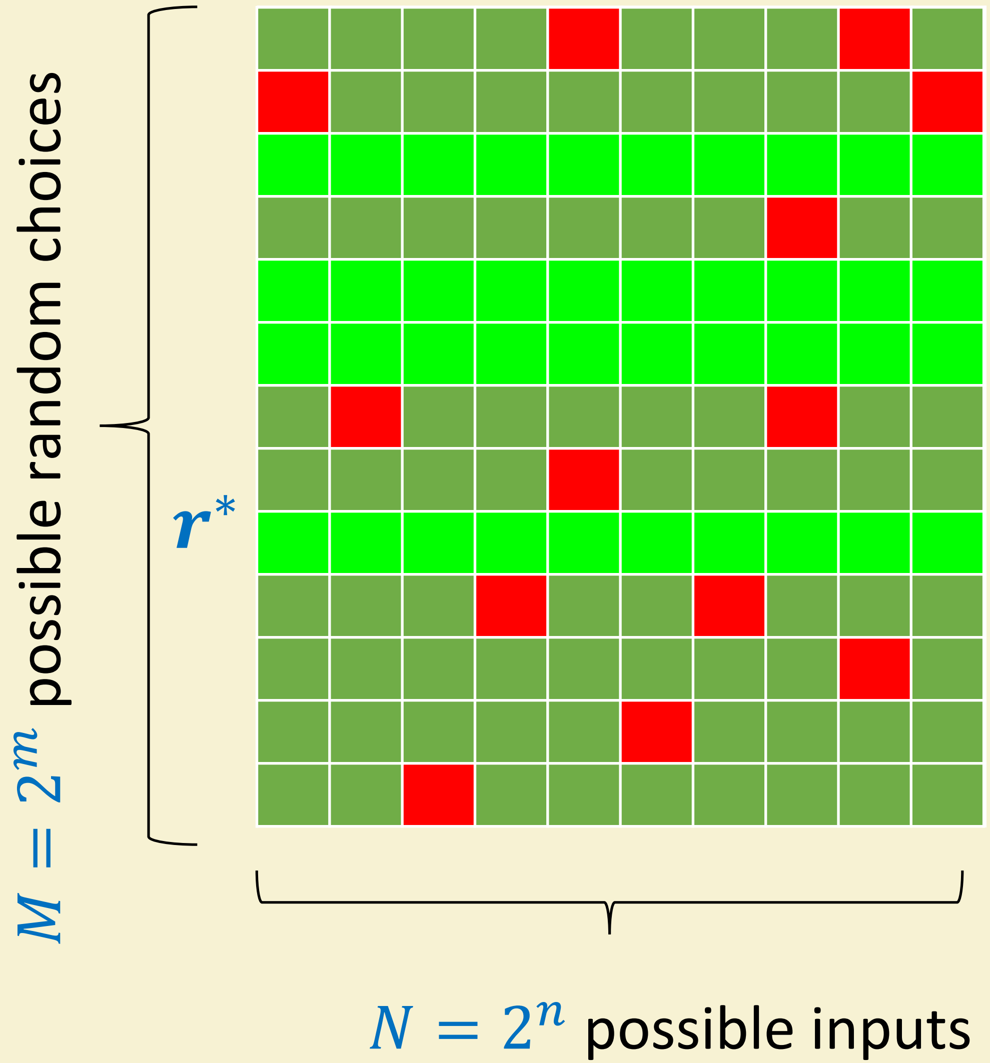
$N = 2^n$ possible inputs

$BPP \subseteq P_{/poly}$ denouement

Proof: Suppose $F \in BPP$ and A is alg using n^a random bits and running in n^b time s.t. $\Pr[\,A(x;r) \neq F(x)] < 0.001 \cdot 2^{-n}$

By $P \subseteq P_{/poly}$ there's circuit C of $\leq n^{4b}$ computing $x, r \mapsto A(x;r)$

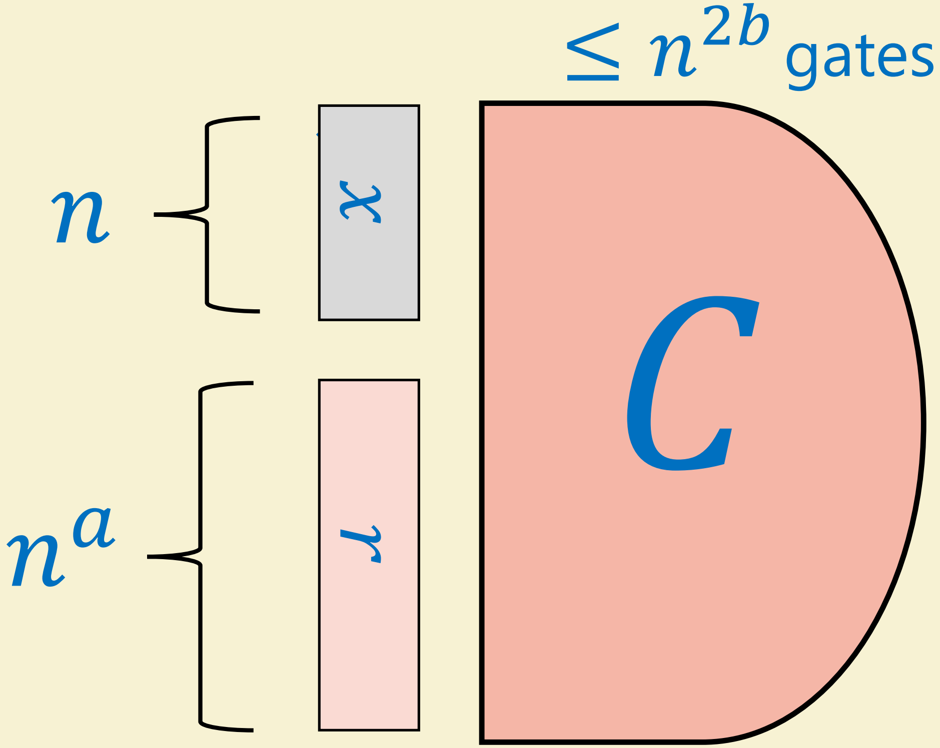


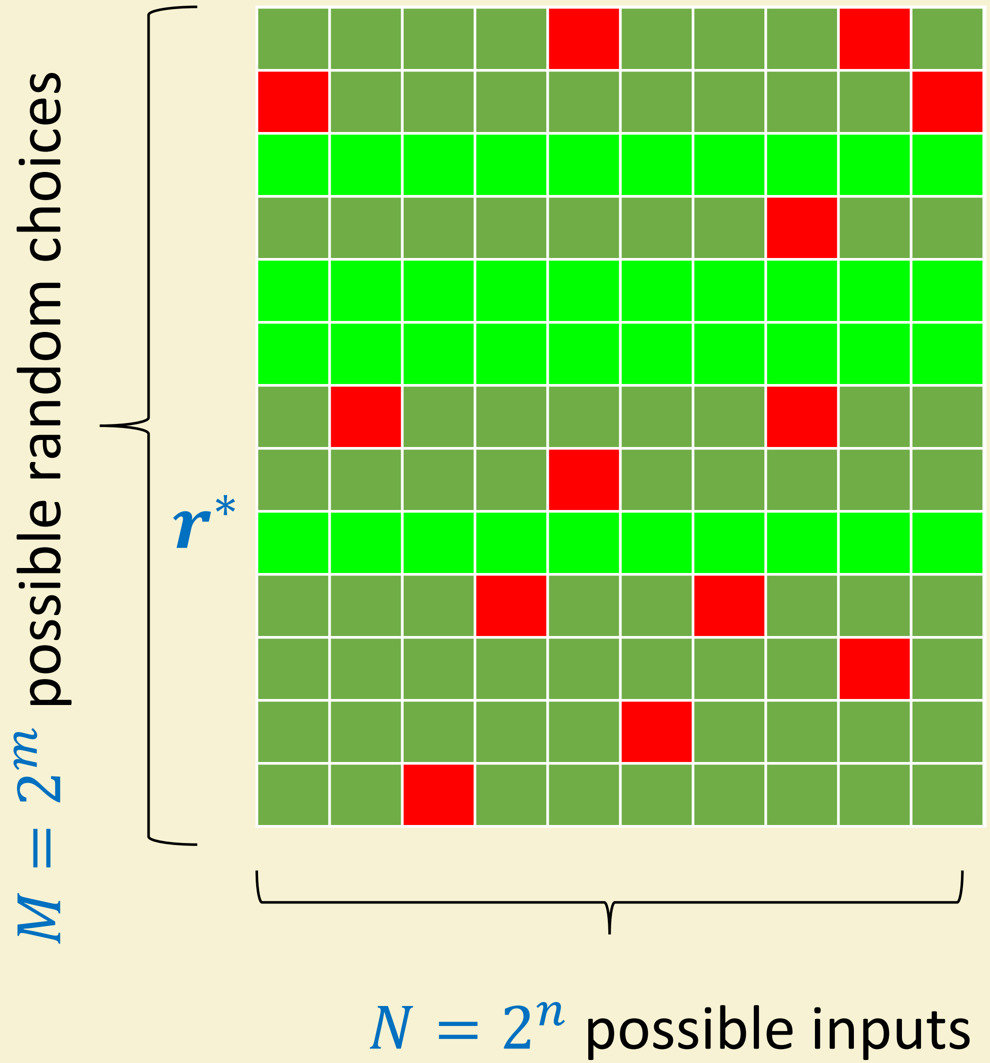


$BPP \subseteq P_{/poly}$ denouement

Proof: Suppose $F \in BPP$ and A is alg using n^a random bits and running in n^b time s.t. $\Pr[\, A(x;r) \neq F(x)] < 0.001 \cdot 2^{-n}$

By $P \subseteq P_{/poly}$ there's circuit C of $\leq n^{4b}$ computing $x, r \mapsto A(x;r)$

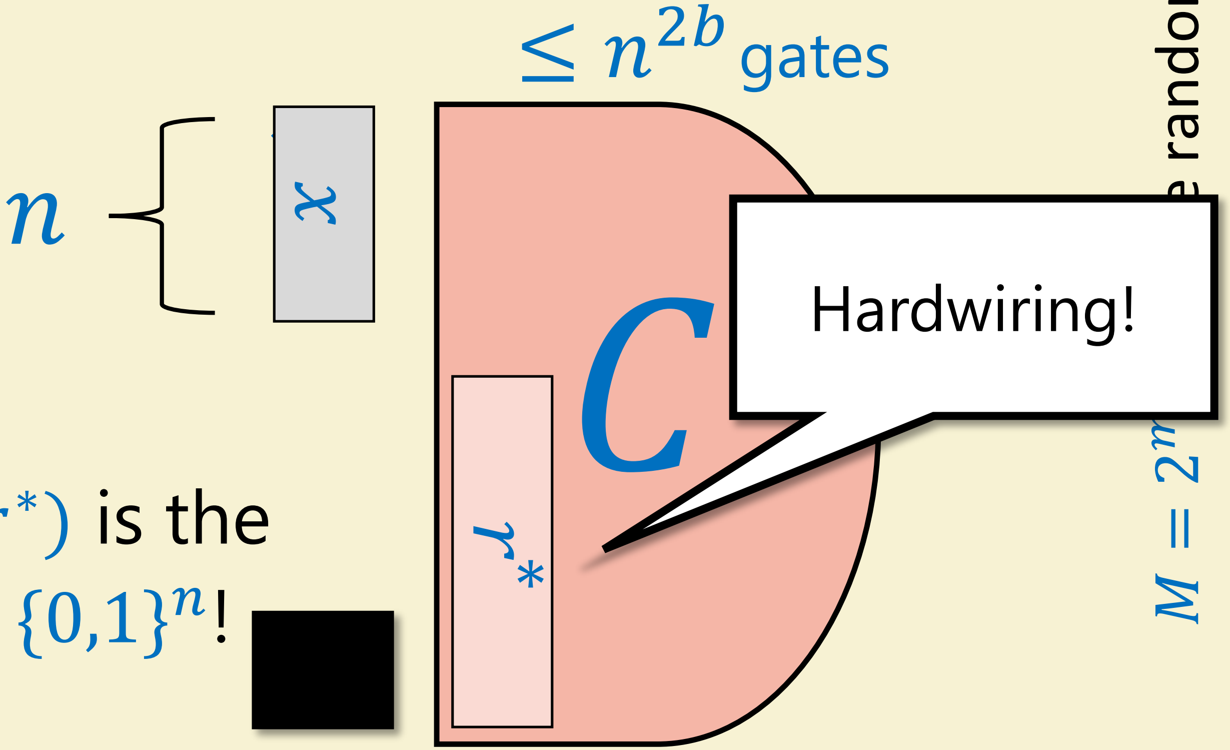


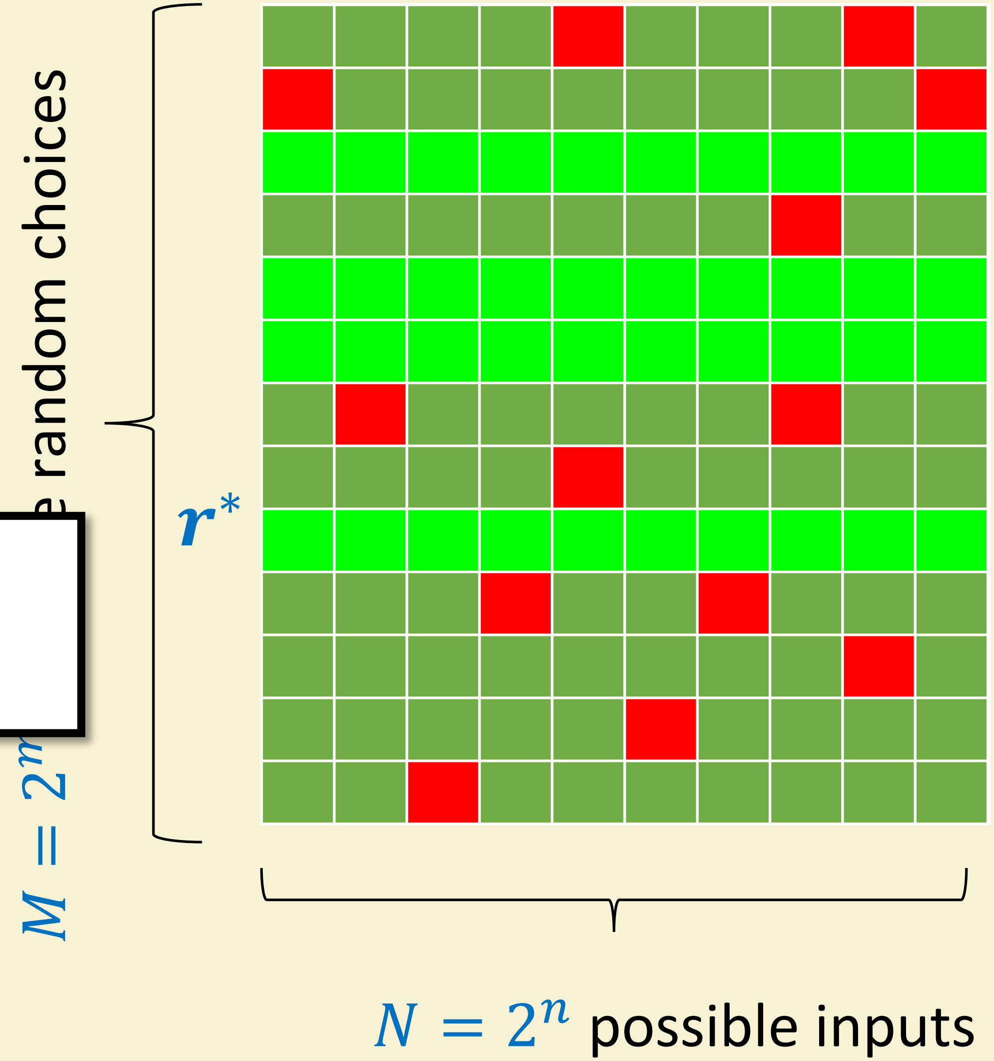


$BPP \subseteq P_{/poly}$ denouement

Proof: Suppose $F \in BPP$ and A is alg using n^a random bits and running in n^b time s.t. $\Pr[\, A(x;r) \neq F(x)] < 0.001 \cdot 2^{-n}$

By $P \subseteq P_{/poly}$ there's circuit C of $\leq n^{4b}$ computing $x, r \mapsto A(x;r)$





$x \mapsto A(x;r^*)$ is the map F on $\{0,1\}^n$!

Recap for now

- $P \subseteq BPP$
- $BPP \subseteq EXP$
- $BPP \subseteq P_{/poly}$
- Unknown if $BPP = P$. Unknown if $BPP = EXP$

Q: Can it be that $P = BPP = EXP$?

Q: Is there a poly-time deterministic algorithm that given randomized alg A for $F \in BPP$ and $n \in \mathbb{N}$ outputs a circuit C_n that computes F on $\{0,1\}^n$?

BPP and NP

Q: Suppose that $F \leq_p G$ and $G \in BPP$. Prove that $F \in BPP$.

$3SAT \in P$ $NP \subseteq P$

Corollary: If $3SAT \in BPP$ then $NP \subseteq BPP$

Unknown: Is $BPP \subseteq NP$? Is $NP \subseteq BPP$? Both? Neither?

Known: Sipser-Gaacs-Lautemann Theorem: If $\mathrm{P} = \mathrm{NP}$ then $BPP = P$

Sipser-Gaacs-Lautemann Thm: If $\boldsymbol{P = NP}$ then $\boldsymbol{BPP = P}$

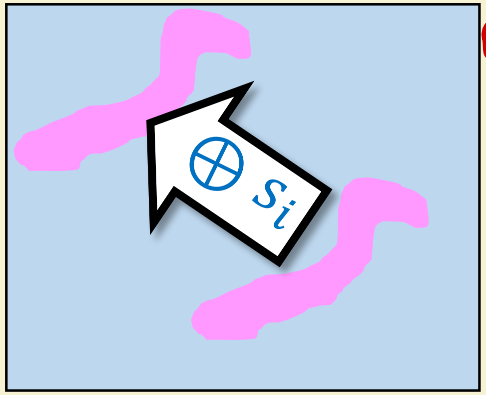
Proof idea: First, amplify like crazy: $m \in n^{O(1)}$

Ensure: $\Pr_{r\sim\{0,1\}^m}[A(x;r) = F(x)] \geq 1 - 2^{-n} > 1 - \frac{1}{1000m}$ one input x

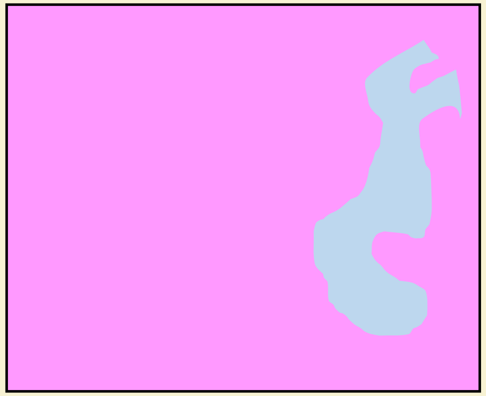
$A(x;r) = 0$

$A(x;r) = 1$

$S_x := \{r \mid A(x;r) = 1\}$



one x



$F(x) = 0 : |S_x| < \frac{1}{1000m} 2^m$ $F(x) = 1: |S_x| > \left(1 - \frac{1}{1000m}\right) 2^m$

MAIN LEMMA: $F(x) = 1$ iff $\exists$ m shifts $s_1, \dots, s_m$ s.t. $\{0,1\}^m = \cup\, (\boldsymbol{S_x} \oplus s_i)$

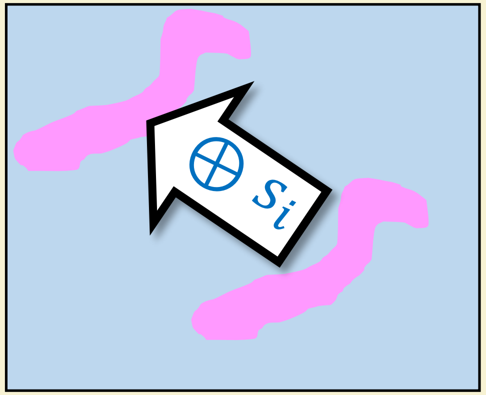
$F(x) = 1$ iff $\exists_{s_1,\dots,s_m \in \{0,1\}^m} \forall_{z \in \{0,1\}^m} \exists_{i \in [m]} \exists_{r \in \{0,1\}^m}: (A(x;r) = 1) \wedge (z = r \oplus s_i)$

Ensure: $\Pr_{r\sim\{0,1\}^m}[A(x;r) = F(x)] \geq 1 - 2^{-n} \geq 1 - \frac{1}{1000m}$

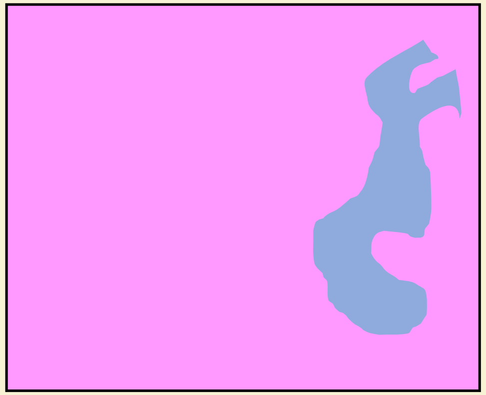
$S_x := \{r | A(x;r) = 1\}$

$A(x;r) = 0$

$A(x;r) = 1$



$F(x) = 0 : |S_x| < \frac{1}{1000m} 2^m$



$F(x) = 1 : |S_x| > \left(1 - \frac{1}{1000m}\right) 2^m$

MAIN LEMMA: $F(x) = 1$ iff $\exists$ m shifts $s_1, \dots, s_m$ s.t. $\{0,1\}^m = \cup\, (\boldsymbol{S_x} \oplus s_i)$

CLAIM 1 ($\Leftarrow$): If $|S| < \frac{1}{1000m} 2^m$ then $\forall_{s_1,\dots,s_m \in \{0,1\}^m}\ |\cup_i (S \oplus s_i)| < 2^m$

Proof: $|S \oplus a| = |S|$

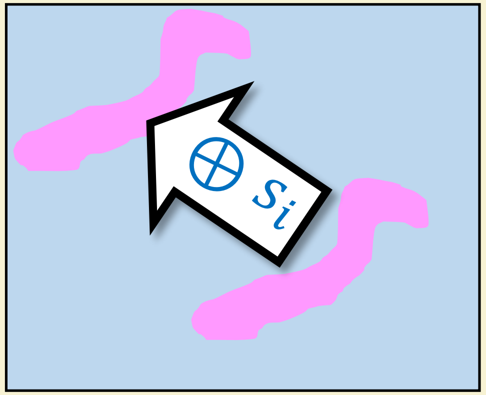
$$|\cup_i (S \oplus s_i)| < m \cdot \frac{1}{1000\, m} 2^m < 2^m \quad \blacksquare$$

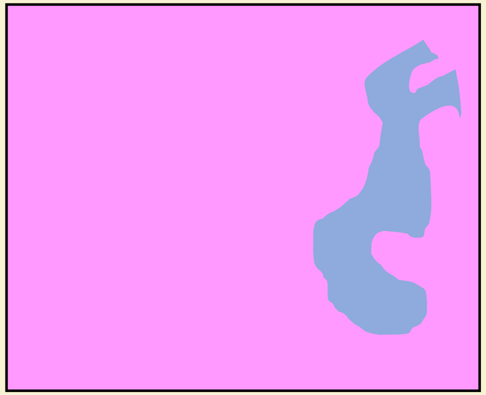
Ensure: $\Pr_{r\sim\{0,1\}^m}[A(x;r)=F(x)] \geq 1-2^{-n} \geq 1-\frac{1}{1000m}$

$S_x := \{r \mid A(x;r)=1\}$

$A(x;r)=0$

$A(x;r)=1$





$F(x)=0: |S_x| < \frac{1}{1000m}2^m$ $\quad$ $F(x)=1: |S_x| > \left(1-\frac{1}{1000m}\right)2^m$

MAIN LEMMA: $F(x)=1$ iff $\exists$ m shifts $s_1,\ldots,s_m$ s.t. $\{0,1\}^m = \cup(\boldsymbol{S_x} \oplus s_i)$

CLAIM 2 ($\Rightarrow$): If $|S| > \frac{2}{3}2^m$ then $\exists_{s_1,\ldots,s_m\in\{0,1\}^m}$ s.t. $\cup_i (S \oplus s_i) = \{0,1\}^m$

Proof: For every $z \in \{0,1\}^m$ $\Pr_s[z \notin S \oplus s] = \Pr[s \notin S \oplus z] < \frac{1}{3}$

$\Rightarrow$ For every $z \in \{0,1\}^m$ $\Pr_{s_1,\ldots,s_m}[\wedge_{i=1}^m z \notin S \oplus s_i] < \left(\frac{1}{3}\right)^m < 2^{-m}$

$\Rightarrow \Pr_{s_1,\ldots,s_m}\left[\exists_{z\in\{0,1\}^m} \wedge_{i=1}^m z \notin S \oplus s_i\right] < 1$ $\Rightarrow \exists s_1,\ldots,s_m$

Sipser-Gaacs-Lautemann Thm: If $P = NP$ then $BPP = P$

MAIN LEMMA: $F(x) = 1$ iff $\exists$ m shifts $s_1, \dots, s_m$ s.t. $\{0,1\}^m = \cup\, (\boldsymbol{S_x} \oplus s_i)$

$F(x) = 1$ iff $\exists_{s_1,\dots,s_m \in \{0,1\}^m} \forall_{z \in \{0,1\}^m} \exists_{i \in [m]} \exists_{r \in \{0,1\}^m}: (A(x;r) = 1) \wedge (z = r \oplus s_i)$

$F(x) = 1$ iff $\exists_{s_1,\dots,s_m} \neg \exists_z \neg \exists_i \exists_r: (A(x;r) = 1) \wedge (z = r \oplus s_i)$

$NP = \Sigma_1$

$\downarrow$

Σ_3

In NP, so replace with P alg (no $\exists_i \exists_r$)

$f(s_1, \dots, s_m, z) = 1$

iff $\exists i \exists r \dots$

Also in P

In NP, so replace with P alg (no $\exists_z$)

$f(s_1, \dots, s_m) = 1$

iff $\exists z \dots$

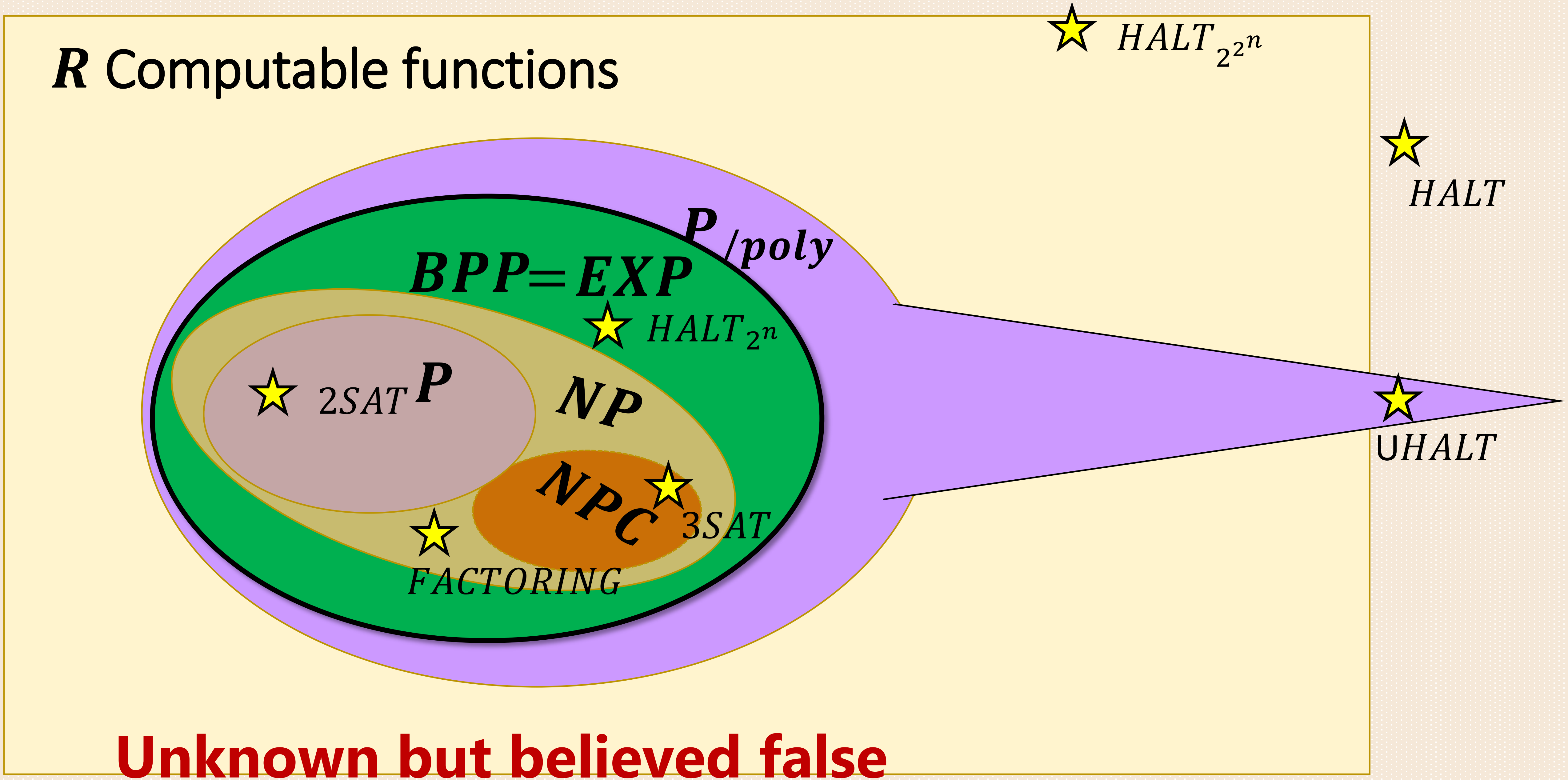
Also in P

In NP, so replace with P alg (no $\exists_{s_1,\dots,s_m}$) ■

BPP and NP recap

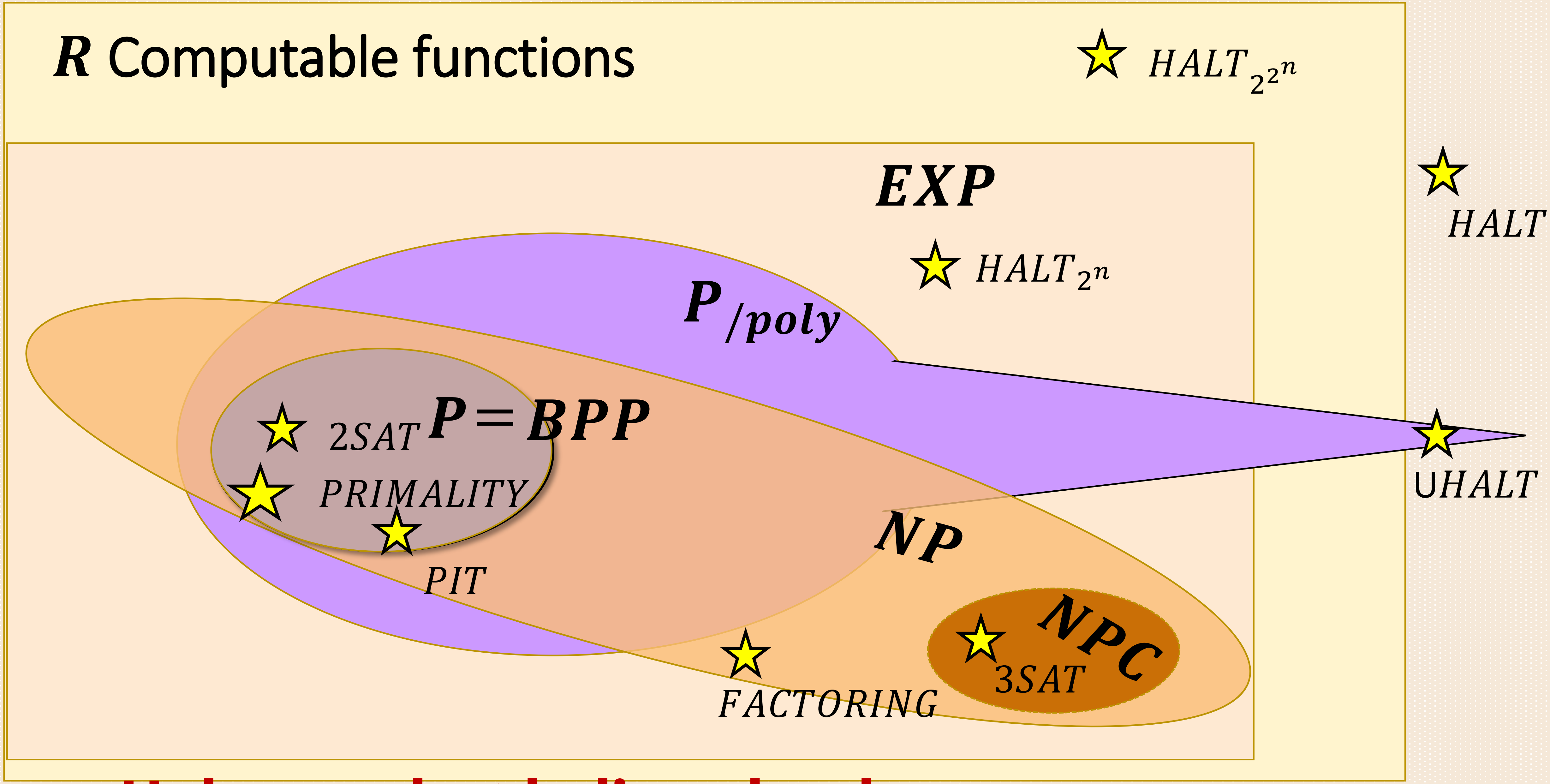
- If $3SAT \in BPP$ then $NP \subseteq BPP$: All theory of NP completeness stays the same if we use BPP as our model of "efficient computation".
- If $P = NP$ then $BPP = P$
- If (as widely believed) $3SAT \notin P_{/poly}$ then $NP \nsubseteq BPP$

All functions $F:\{0,1\}^* \to \{0,1\}$



Unknown but believed false

All functions $F: \{0,1\}^* \to \{0,1\}$



Unknown but believed to be true

BPP recap

- $P \subseteq BPP \subseteq EXP$
  Unknown if either inclusion strict but can't have $P = BPP = EXP$
- $BPP \subseteq P_{/poly}$
- If BPP contains an NP-complete problem then $NP \subseteq BPP$
- Relation with NP unknown
- If $P = NP$ then $BPP = P$
- It is believed that $P \neq NP$ (of course) but it is also believed that $BPP = P$.

Next Lecture: Wrap-up

- Quantum Computation
  - Most credible challenger to Strong Church-Turing Thesis.
  - Less-Strong Church-Turing Thesis: SCTT but with quantum computers.
- Cryptography, Society
- Exam info

Bonus topics:

- One sided error algorithms: $coRP, RP$
- "Zero sided error" (Las Vegas): ZPP
- Known that $ZPP = RP \cap coRP$ and that $RP \cup coRP \subseteq BPP$
- Known that $RP \subseteq NP$ and $coRP \subseteq coNP$
- Pseudorandom generators
- relation between counting and sampling.
- Randomized reductions.