Wed April 25, 2018 - page 1

Tuesday, April 24, 2018 8:28 PM

Another example of the use of randomness.

A "YES" instance of SAT may have many solutions.
Does the difficulty come from not knowing which one to work on?

Suppose you are given ϕ that has 0 or 1 satisfying assignments.
Can you determine which?

OR

Given an algorithm that can distinguish between 0 or 1 satisfying assignments, could we solve general instances efficiently?
Yes, but the only way we know how to do this is with a randomized reduction.

Theorem: (Valiant - Vazirani)
There is a randomized poly-time procedure.
Input: 3CNF formula ϕ
Output: ϕ' such that.
If ϕ is not satisfiable then ϕ' is not satisfiable
If ϕ is satisfiable then w.p. $1/8n$ ϕ' has exactly one satisfying assignment.

Tuesday, April 24, 2018 8:28 PM

If ϕ is not satisfiable
Output = "No".
If ϕ' is satisfiable
Output = "YES" if reduction succeeds ≥ 1 time.

Repeat $8kn$ times:
$\phi \to \phi'$
Use poly-time procedure on ϕ' to determine between 0 & 1 satisfying assignments.
Assignment for ϕ' found? Return (Yes)
Assignment for ϕ' not found? continue.

After loop, reject.

Probability of failure $= \left(1 - \frac{1}{8n}\right)^{k8n} \sim e^{-k}$

$\left[\left[1 - \frac{1}{8n}\right]^{8n}\right]^{k}$ e^{-1}

$\left(1 - \frac{1}{x}\right)^x \sim e^{-1}$ for large x.

Proof: $\phi' = \phi \wedge \theta_1 \wedge \theta_2 \wedge \cdots \theta_k$.

If ϕ not satisfiable then ϕ' not satisfiable.

Suppose ϕ is satisfiable. Let $T=$ set of satisfying assignments.
Assume $x_1 = x_2 \cdots = x_n = 0 \notin T$.

θ_i: Select a random subset S_i of the variables in ϕ.
$S_i \subseteq \{x_1, x_2 \dots x_n\}$
If $|S_i| = k$ add k extra variables $y_1 \cdots y_k$.
$S_i = \{x_{i_1}, x_{i_2} \cdots x_{i_k}\}$

$(y_0) \wedge (x_1 \to y_0 \neq y_1) \wedge (\neg x_1 \to y_0 = y_1)$

Wed April 25, 2018 - page 3

Tuesday, April 24, 2018 8:28 PM

Θ_i : Select a random subset S_i of the variables in Φ.

$S_i \subseteq \{x_1, x_2 \dots x_n\}$

If $|S_i| = k$ add k extra variables $y_1 \dots y_k$.

$S_i = \{x_{i_1}, x_{i_2} \dots x_{i_k}\}$

$y_0 = 1$.

$$(y_0) \wedge (x_{i_1} \to y_0 \neq y_1) \wedge (\neg x_{i_1} \to y_0 = y_1)$$
$$\wedge (x_{i_2} \to y_1 \neq y_2) \wedge (\neg x_{i_2} \to y_1 = y_2)$$
$$\vdots$$
$$\wedge (x_{i_k} \to y_{k-1} \neq y_k) \wedge (\neg x_{i_k} \to y_{k-1} = y_k) \wedge (y_k)$$

$y_j = 1$ iff an even # of variables in $x_{i_1} \dots x_{i_j}$ are $= 1$.

Since y_k must be 1 this is satisfiable if the # of variables in S_i that are $= 1$ is even.

Reduction: On input Φ, select a random $k \in \{0, \dots, n-1\}$

$$\Phi' = \Phi \wedge \Theta_1 \wedge \Theta_2 \dots \wedge \Theta_{k+2}$$

each Θ_i generated randomly

Claim 1: If $|T| > 0$ then

$$\Pr_{k \in \{0, \dots, n-1\}} \left[2^k \leq |T| < 2^{k+1} \right] \geq 1/n.$$

Probability we pick the right k.

$0 < |T| < 2^n$

0 1 2 4 . 2^{n-1} 2^n

Wed April 25, 2018 - page 4

Tuesday, April 24, 2018 8:28 PM

Claim 1: if $|T| > 0$ then

$$\Pr_{k \in \{0,\dots,n-1\}} \left[2^k \le |T| < 2^{k+1} \right] \ge 1/n.$$

Claim 2: if $2^k \le |T| \le 2^{k+1}$ then the probability that

$\phi \wedge \theta_1 \wedge \theta_2 \wedge \cdots \wedge \theta_{k+2}$

has exactly one satisfying assignment $\ge \frac{1}{8}$

$\Rightarrow$ probability of success $= \frac{1}{8n}$.

Assume the all 0 assignment does not satisfy ϕ

$\Rightarrow$ Let $t \in T$ $\quad$ Prob[t satisfies θ_i] $= 1/2$

1	0	1	1	0	1	1	0	0
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9.

$= t$.

pick S_i

even or odd $\rightarrow$

$\Rightarrow$ Let $t + t' \in T$. $\quad$ Prob[t + t' both satisfy θ_i] $= 1/4$

	x_ℓ	x_k	
t	0	1	odd / even
t'	1	1	odd / even.

Find a variable x_ℓ where $t + t'$ differ

Say $t_\ell = 0$ $\quad$ $t'_\ell = 1$.

There must be a variable x_k where $t_k = 1$.

Wed April 25, 2018 - page 5

Tuesday, April 24, 2018 8:28 PM

Regardless of whether $t'_k = 0$ or 1

Only one possibility for $x_\ell \in / \notin S_i$ $x_k \in / \notin S_i$

will cause t & t' to both satisfy θ_i.

Since the S_i are all chosen independently:

$$\text{Prob}[t \text{ satisfies } \underbrace{\phi \wedge \theta_1 \wedge \cdots \wedge \theta_{k+2}}_{\phi_k}] = \left(\tfrac{1}{2}\right)^{k+2}.$$

$$\text{Prob}[t \,\&\, t' \text{ both satisfy } \phi \wedge \theta_1 \wedge \theta_2 \cdots \wedge \theta_{k+2}] = \left(\tfrac{1}{4}\right)^{k+2}$$

Prob[t uniquely satisfies ϕ_k] =

$$\underbrace{\text{Prob}[t \text{ satisfies } \phi_k]}_{\left(\frac{1}{2}\right)^{k+2}} - \underbrace{\text{Prob}[t \text{ and some other } t_i \text{ both satisfy } \phi_k]}_{\text{upper bound.}}$$

$2^k \le |T| \le 2^{k+1}$

$\left(\frac{1}{4}\right)^{k+2}.$

$$\le (|T|-1)\,\text{Prob}[t \,\&\, t' \text{ both satisfy } \phi_k].$$

$$\ge \left(\tfrac{1}{2}\right)^{k+2} - \frac{2^{k+1}}{\underbrace{4^{k+2}}_{2^{k+2}\cdot 2^{k+2}}} = \left(\tfrac{1}{2}\right)^{k+2}\left[1 - \frac{2^{k+1}}{2^{k+2}}\right] = \frac{1}{2^{k+3}}$$

$$\text{Prob}[\exists\, t \text{ that uniquely satisfies } \phi_{k+2}] \ge \frac{|T|}{2^{k+3}} \ge \frac{2^k}{2^{k+3}} = \frac{1}{8}. \quad //$$

$$\text{Prob}[A \text{ or } B] = \text{Prob}[A \cup B]$$
$$\leq \text{Prob}[A] + \text{Prob}[B]$$

Prob [t + some other t' both satisfy Θ_i]

Prob [t + t_1 sat Θ_i or t + t_2 sat Θ_i ... or t + t_n sat Θ_i].

$|T| - 1 \cdot$ prob [t + t' sat Θ_i].

Tuesday, April 24, 2018 8:28 PM

Randomized Complexity Classes.

model probabilistic TM
deterministic TM w/ an additional read-only input tape containing coin flips.

BPP : Bounded Error Probabilistic Poly-time
$L \in BPP$ if $\exists$ p.p.t. TM M (p.p.t. = probabilistic poly-time)

$\frac{1}{2} + \frac{1}{poly(n)}$.

$x \in L \Rightarrow Prob_y[M(x,y) \text{ accepts}] \geq 2/3$
$x \notin L \Rightarrow Prob_y[M(x,y) \text{ rejects}] \geq 2/3$
(2-sided error).

		co-RP
RP :	$x \in L$ $Prob_y[M(x,y) \text{ accepts}] \geq 1/2$	1
	$x \notin L$ $Prob_y[M(x,y) \text{ rejects}] = 1$.	1/2

ZPP : (zero error, prob poly-time) $ZPP = RP \cap co\text{-}RP$
$Prob_y[M(x,y) \text{ outputs "I don't know"}] \leq 1/2$
otherwise it outputs the correct answer.

Or runs in expected poly-time but always outputs the correct answer.

These classes capture "efficiently computable" better than P.

→ The $1/2$ in the definition of ZPP, RP or co-RP can be replaced by $\frac{1}{\text{poly}(n)}$.

→ The $2/3$ in the definition of BPP can be replaced by $1/2 + 1/\text{poly}(n)$.
(via error reduction).

Suppose we have L + p.p.t. TM M.

$$x \in L \Rightarrow \Pr[M \text{ accepts}] \geq \epsilon$$
$$x \notin L \Rightarrow \Pr[M \text{ rejects}] = 1$$

M': run M k/ϵ times, each run uses independent coin flips.

- accept if any run of M accepts.
- reject otherwise.

if $x \in L$ prob a particular run is "bad" $\leq (1-\epsilon)$
prob all runs are bad: $\leq (1-\epsilon)^{k/\epsilon}$

$$= \left[\underbrace{(1-\epsilon)^{1/\epsilon}}_{e^{-1}}\right]^k \sim e^{-k}.$$

if $\epsilon = 1/\text{poly}(n)$.
$\frac{n}{\epsilon} = \text{poly}(n)$.
⇒ prob error e^{-n}

if $x \notin L$
$\Pr[M' \text{ rejects}] = 1$.

Prob M accepts $\geq 1 - e^{-k}$.

Tuesday, April 24, 2018 8:28 PM

Error Reduction for BPP :

$x \in L$ $\Pr[M \text{ accepts}] \geq 1/2 + \epsilon$.

$x \notin L$ $\Pr[M \text{ rejects}] \geq 1/2 + \epsilon$.

Simulate M $\frac{k}{\epsilon^2}$ times with independent coin flips.

take the majority answer.

x_i : random variable $= 1$ if i^{th} answer is correct
$= 0$ otherwise.

$\Pr[x_i = 0] \leq 1/2 - \epsilon$ $\Pr[x_i = 1] \geq 1/2 + \epsilon$

$E[x_i] \geq 1/2 + \epsilon$. x_i's are mutually independent

$X = \sum_i^t x_i$ $\mu = E[X] = (1/2 + \epsilon)\frac{k}{\epsilon^2}$ $m = \frac{k}{\epsilon^2}$

$E[X] = \frac{m}{2} + \epsilon m$.

Chernoff :

$\text{Prob}[X \leq m/2] \leq 2^{-\Omega(\epsilon^2 m)} \ldots 2^{-\Omega(k)}$

$\Rightarrow$ If $\epsilon > \frac{1}{\text{poly}(n)}$ & k is $\text{poly}(n)$

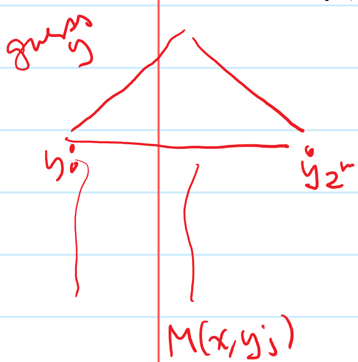
$\Rightarrow$ $\frac{k}{\epsilon^2}$ is $\text{poly}(n)$ & error is exp small.

Tuesday, April 24, 2018 8:28 PM

RP, co-RP, BPP & ZPP all contain P.
(the TM can always ignore the random string).

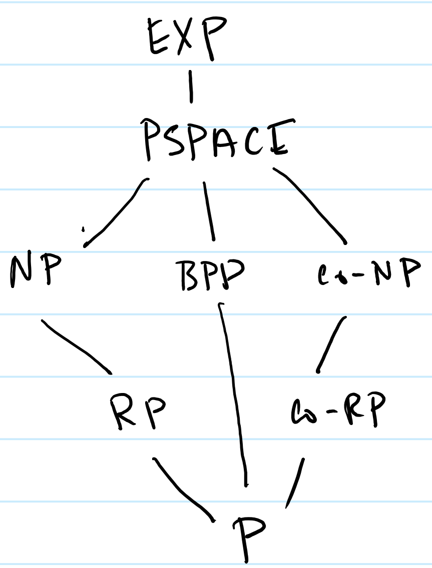
All contained in PSPACE:
Exhaustively try all y and count the #
of accepting computations. $\Pr[\text{accept}] = \dfrac{\#\, y \text{ s.t. } M(x,y) = \text{acc}}{\#\text{ all possible } y\text{'s}}$.

Also $RP \subset NP$ (and co-RP $\subseteq$ co-NP).
An NTM can guess y then compute $M(x,y)$



if $x \notin L$ $\forall y$ $M(x,y)$ rejects.
if $x \in L$ $M(x,y)$ accepts for most of the y's.

$\to$ NP only requires ≥ 1 y to accept.



How powerful is BPP?

There is an example of a problem in BPP that we only know how to do in EXP.

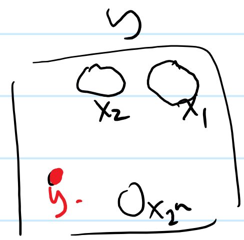
Don't know if BPP = EXP
(or even NEXP).

Strong hints that BPP $\neq$ EXP however

Tuesday, April 24, 2018 8:28 PM

Is there a deterministic simulation of BPP that does better than brute-force search?

Yes, but only if we allow non-uniformity.



Theorem: $BPP \subset P/poly$ (Adleman).

Take $L \in BPP$

Error reduction gives TM M s.t.

$\exists y\ \forall x$ $M(x,y)$ is correct.

if $x \in L$ $|x| = n$ $Pr_y[M(x,y) \text{ accepts}] \geq 1 - (1/2)^{n^2}$

if $x \notin L$ $|x| = n$ $Pr_y[M(x,y) \text{ rejects}] \geq 1 - (1/2)^{n^2}$

y is "bad" for x if $M(x,y)$ gives the wrong answer.

Fix x. $Pr_y[y \text{ is bad for } x] \leq (1/2)^{n^2}$

$Pr_y[y \text{ is bad for some } x,\ x = |n|] \leq 2^n \left(\frac{1}{2}\right)^{n^2} < 1$

$Pr_y[y \text{ is good for all } x\text{'s } |x| = n] > 0$

Summing up over all x.

There exists a y which is good for all inputs x of length n.

This y is the hint for all inputs of length n. (hard code y into C_n).

If $BPP = EXP$ then $EXP \subset P/poly$.

Wednesday, April 25, 2018 9:16 AM

Does BPP have complete problems?

Determining if a TM is an NTM is easy.
Determining if a TM is in BPTIME is undecidable
since it requires that every string is accepted
w/ probability $\leq 1/3$ or $\geq 2/3$.

$M(x, y)$

A natural candidate for a BPP-complete problem would be

$A = (M, x, 1^t)$ M accepts x w.p. $\geq 2/3$ in time t.

This problem is BPP-hard. $L \in BPP$ $L \leq A$.

However is $A \in BPP$?
A BPP machine can't just simulate M on input x
because it could be that M accepts x
with probability ½ on input x.

If BPP = P (conjectured to be true)
then BPP does have complete problems
because P has complete problems.