LAST TIME:

Discrete Cube Root Problem: Two primes p, q, of the form $3k+2$. $N = pq$, $\varphi(N) = (p-1)(q-1)$. $3 \nmid \varphi(N)$, Given N & x as input output $y \in \mathbb{Z}_N^*$ s.t. $y^3 \equiv x \pmod N$.

DCRA Assumption: $\forall$ poly r, $\nexists$ alg that runs in time $r(n)$ that on inputs N, x, where N defined as above as $n = \#$ bits in N, outputs $y \in \mathbb{Z}_N^*$ that w.p. $\geq \frac{1}{r(n)}$ satisfies $y^3 \equiv x \pmod N$. The probability is over random draws of p, q & x and any internal randomization of alg.

$f_N : \mathbb{Z}_N^* \to \mathbb{Z}_N^*$, $f_N(y) \equiv y^3 \pmod N$ is a bijection.

$f_N^{-1} : \mathbb{Z}_N^* \to \mathbb{Z}_N^*$, $f_N^{-1}(x) \equiv x^d \pmod N$ where d is s.t. $3d \equiv 1 \pmod{\varphi(N)}$ (Euler's Theorem)

Let $((x_i, y_i))_{i=1}^m$, where $x_i \sim_{unif} \mathbb{Z}_N^*$ & y_i s.t. $y_i^3 \equiv x_i \pmod N$.

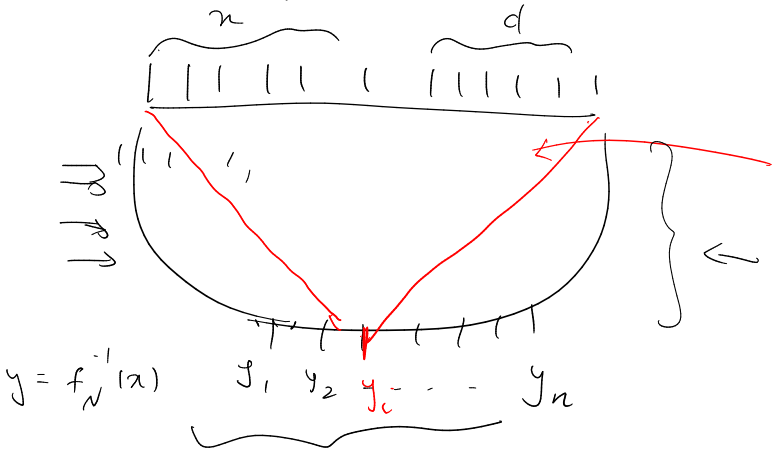
i.e. $y_i = f_N^{-1}(x_i)$

Let $\{ f_{N,i}^{-1} \mid i = 1, \ldots, n \}$ be the functions outputting the i^{th} bit of $f_N^{-1}(\cdot)$.

(At least one of these functions must be hard to learn).

$f_N^{-1}(x) \equiv x^d \pmod N$ (if d is known)

Poly-time Turing machine: $f_N^{-1}(x)$, given x & d as inputs.



x & d are both n bits long

circuit to compute $f_{N,i}^{-1}$

depth = O(Running of TM)

width = O(space used by Turing Machine)

The class of polynomial-sized circuits is not PAC-Learnable, $p(n) = n^{10}$

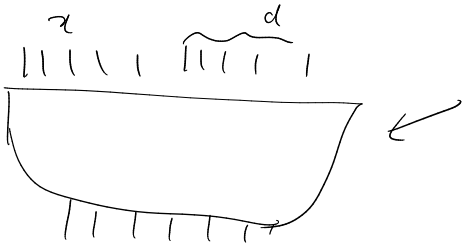
How do we compute x^d efficiently?

x^d : $x^{2^0}, x^{2^1}, x^{2^2}, x^{2^3}, \ldots, x^{2^{\lfloor \log \varphi(N) \rfloor}}$ $\quad d \le \varphi(N)$

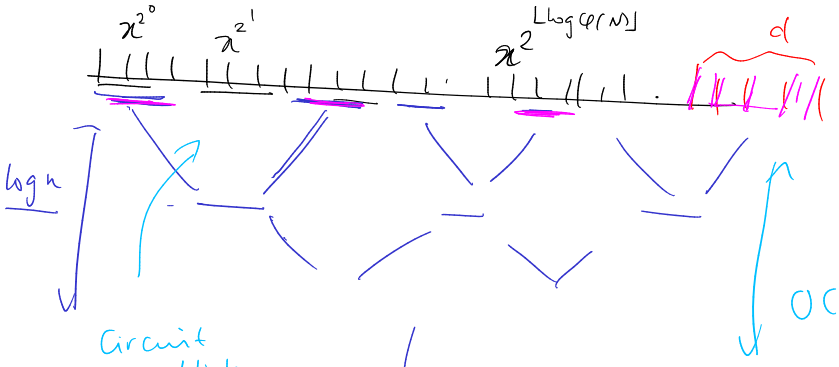
d : $d_{n-1} d_{n-2} \cdots d_1 d_0$ (d in binary)

$$x^d = \prod_{i:\, d_i = 1} x^{2^i}$$

(at most n terms in the product)



circuit computes $x^d \pmod N$



Circuit multiplies 2 n-bit numbers $O(\log n)$ depth.

$(x, x^2, x^4, \ldots, x^{2^{\lfloor \log \varphi(N) \rfloor}}, y)$

$\oplus(n^2)$

y s.t. $y^3 \equiv x \pmod N$

$$f_N^{-1} : (\mathbb{Z}_N^*)^n \to \mathbb{Z}_N^*$$

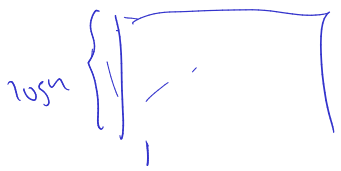
$(f_{N,i}^{-1})$ ← computed by a circuit of depth $O(\log^2 n)$.

(Can do this in $O(\log n)$ depth).

$C = \{ c : X \to \{0,1\} \mid c$ can be represented as a circuit of depth $\leq 10 \log n \}$

(NC^1 is hard to PAC-learn) [under DCRA].

$P, q \sim$ length n bit random primes of the form $3k+2$



$2^{c \log n} \leq \text{poly}(n)$

($VC\text{-dim}(C) \leq \log |C|$ for C finite).

PAC Learning Framework

	DISJUNCTIONS/ CONJUNCTIONS	3-TERM DNF	DT $\lvert T\rvert \le poly(n)$	DNF $\lvert\varphi\rvert \le poly(n)$	Log-Depth Circuits
Efficient/Low sample complexity	✓	✓	✓	✓	✓
Efficient algorithms	✓	✓	?	? (Daniely-Shalev-Schwartz)	X (under DCRA)
Proper Efficient algorithms ($H = C$ output from C)	✓	X (unless RP=NP)	X (unless RP=NP)	X (unless RP=NP)	X

S, "$\exists c \in C$, s.t. c is consistent with S"

Learning using Equivalence & Membership Queries.

Membership Query (Value Query). [Membership Oracle (MQ)]

Target concept is $c \in C$, L can ask: "given x, $c(x) = ?$"

Equivalence Query: [Equivalence Oracle (EQ)]

L submits $h : X \to \{0,1\}$.

if $h \equiv c$ (as a function) returns SUCCESS

else it receives $x \in X$, s.t. $h(x) \neq c(x)$ (counter example)

Defn: (EXACT Learning using MQ+EQ). We say that C is exactly efficiently learnable using membership & equivalence queries, if $\exists$ poly $p(\cdot)$ & and algorithm L, which when given access to MQ & EQ oracles, $\forall c \in C_n$, outputs in time $p(n, \text{size}(c))$ $h \in C_n$ s.t. $h \equiv c$.

PAC+MQ: Everything in PAC learning + access to MQ.

if C is exactly learnable using MQ+EQ. $\Rightarrow$ C is PAC+MQ learnable.

(Exercise on Problem Sheet 3)

- Learning MONOTONE DNF using EQ+MQ (exactly)

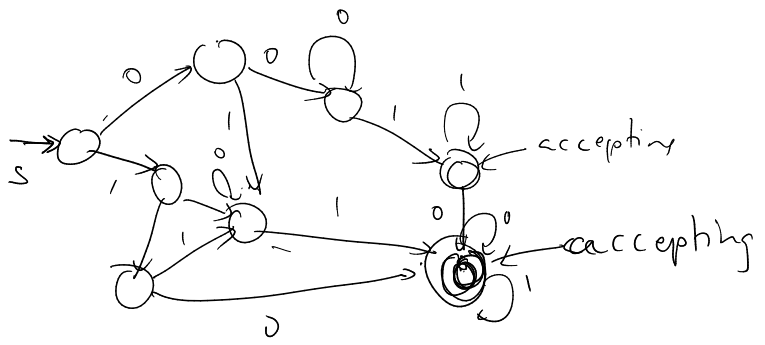
$\varphi \equiv (x_{i_1} \wedge x_{i_2} \wedge \cdots \wedge x_{i_k}) \vee (x \wedge \cdots) \vee \cdots \vee (\cdots)$

No negated literals.

(Only represent monotone f^{n}s from $\{0,1\}^n \to \{0,1\}$)

Learning Deterministic Finite Automata (DFA)

$L \subseteq \{0,1\}^*$.



01011100 1

$ADFA_n$: All accepting paths have length exactly n.

$(ADFA_n)$ is not efficiently PAC learnable under DCRA.

DFA are learnable in EXACT learning with MQ+EQ model.