CPSC 421/501

- End of new material:

  A portion of §9.3 of [Sip]

  "How to solve P vs. NP"

  really How people are trying to solve P vs NP...

- 1 Presentation today: Pumping Lemma
- I'll post the topics for Nov 26

  Dec 1 Dec 3 → there will be a little bit of time

[Think: Final 17th at 7pm]

---

§9.3: You have a Boolean function

$f = f(x_1, \ldots, x_n) : \{F, T\}^n \to \{F, T\}$

often $\{0, 1\}^n \to \{0, 1\}$

Sipser: "Parity" $f(x_1, x_2, x_3, x_4) = x_1 \oplus x_2 \oplus x_3 \oplus x_4$

where $\oplus$ = XOR = exclusive or

Min Formula Size, Min Circuit Size

Overview: # Boolean functions on n-variables $2^{(2^n)}$

$$f = f(\underbrace{x_1, \ldots, x_n}_{2^n \; T/F}) = T/F$$

$\Rightarrow$ "Most" Boolean functions on n-variables have minimum circuit size $\geq 2^n / 3n$

- Cook-Levin approach $\Rightarrow$

SAT, 3SAT, ... any L language

if $L \in P$, then there are polynomial size circuits to compute:

given input w, $|w| = n$, there is a circuit of size poly(n) to determine whether or not $w \in L$

Given $f(x_1, \ldots, x_4) = x_1 \oplus x_2 \oplus x_3 \oplus x_4$

$$f(x_1, x_2) = x_1 \oplus x_2$$

"parity functions"

$F \leftrightarrow 0, \quad T \leftrightarrow 1$

$$f(x_1, x_2) = \begin{cases} 1 & \text{if } x_1 = 0, x_2 = 1 \\ & \quad\;\; x_1 = 1, x_2 = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$= (x_1 + x_2) \bmod 2$$

(Next course in complexity theory CPSC 506, but ...)

$$f(x_1, x_2, x_3, x_4) = x_1 \oplus x_2 \oplus x_3 \oplus x_4$$

$$= (x_1 + x_2 + x_3 + x_4) \bmod 2$$

Compute:

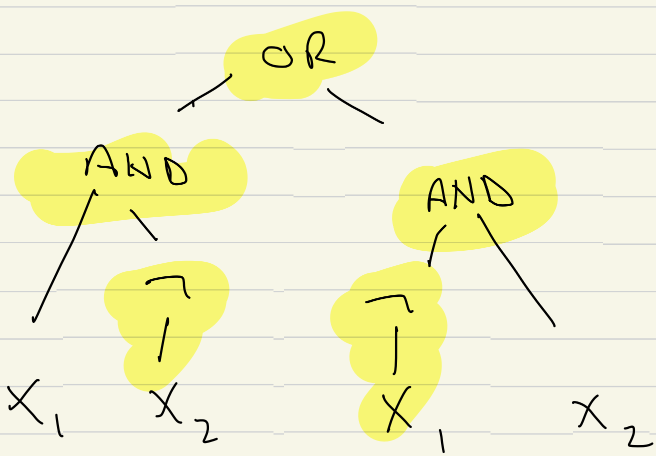
f as formula, say

AND

OR

$\neg$

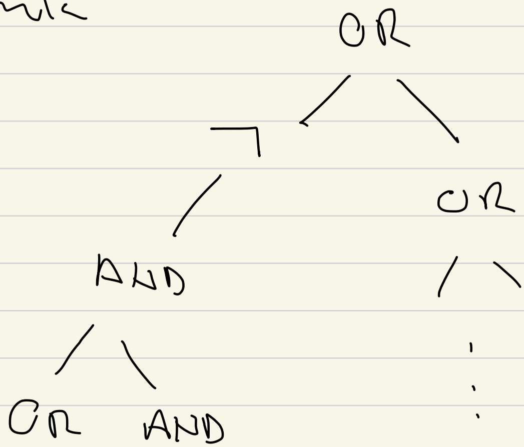
$$x_1 \oplus x_2 = (x_1 \text{ AND } \neg x_2) \text{ OR } (\neg x_1 \text{ AND } x_2)$$

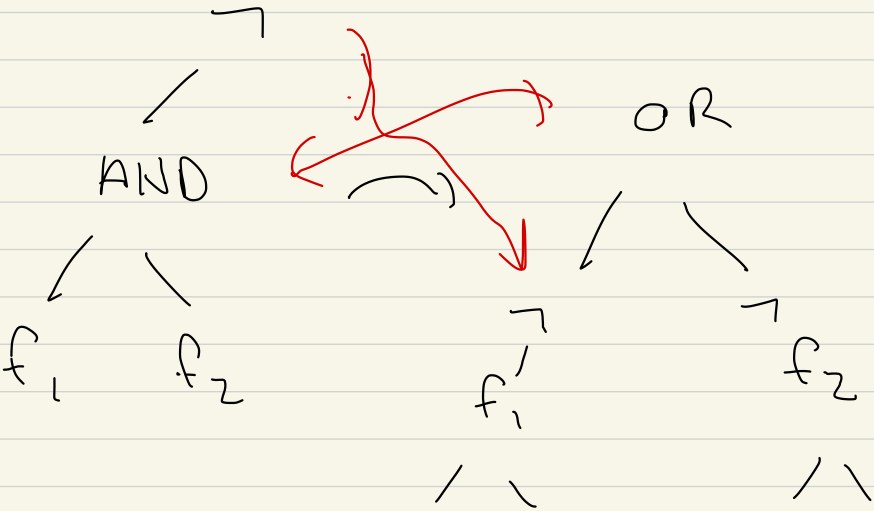


can always $\neg$ at bottom

e.g.

formula





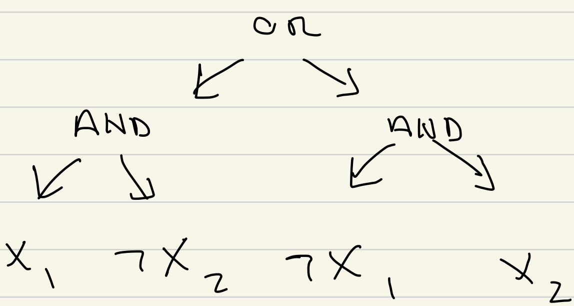
Smallest formula for $f(x_1, \dots, x_n)$

is smallest tree size to compute

f

= # leaves

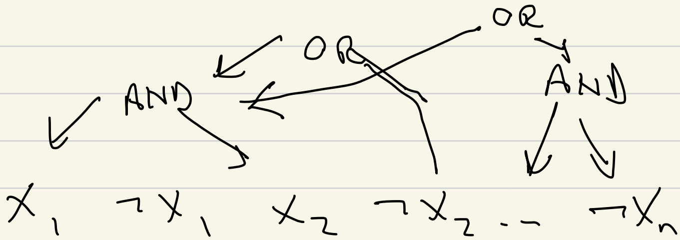
= # literals



$$(x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2)$$

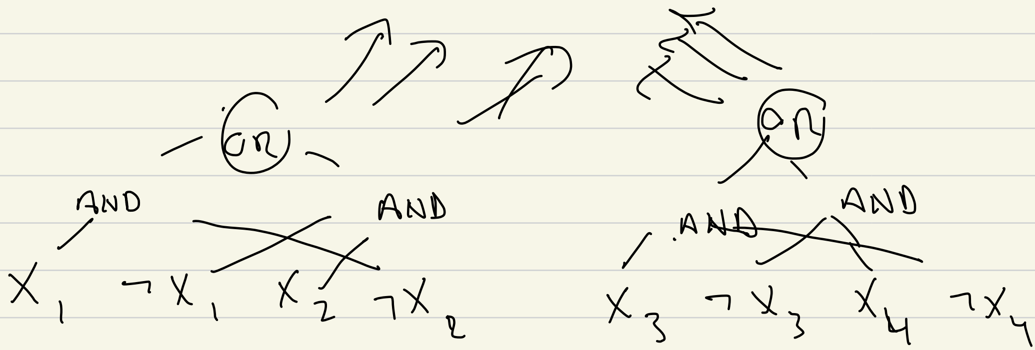
size formula = 4

Circuit to compute f

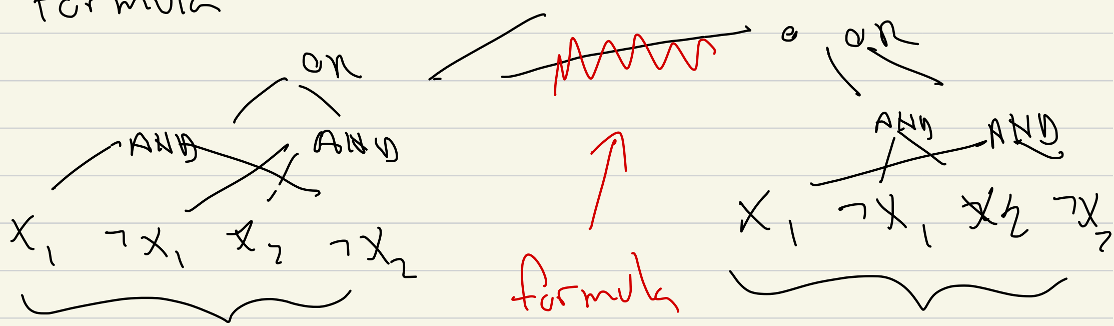


Circuit/Formula to compute f:

$(x_1 \oplus x_2) \oplus (x_3 \oplus x_4)$



formula



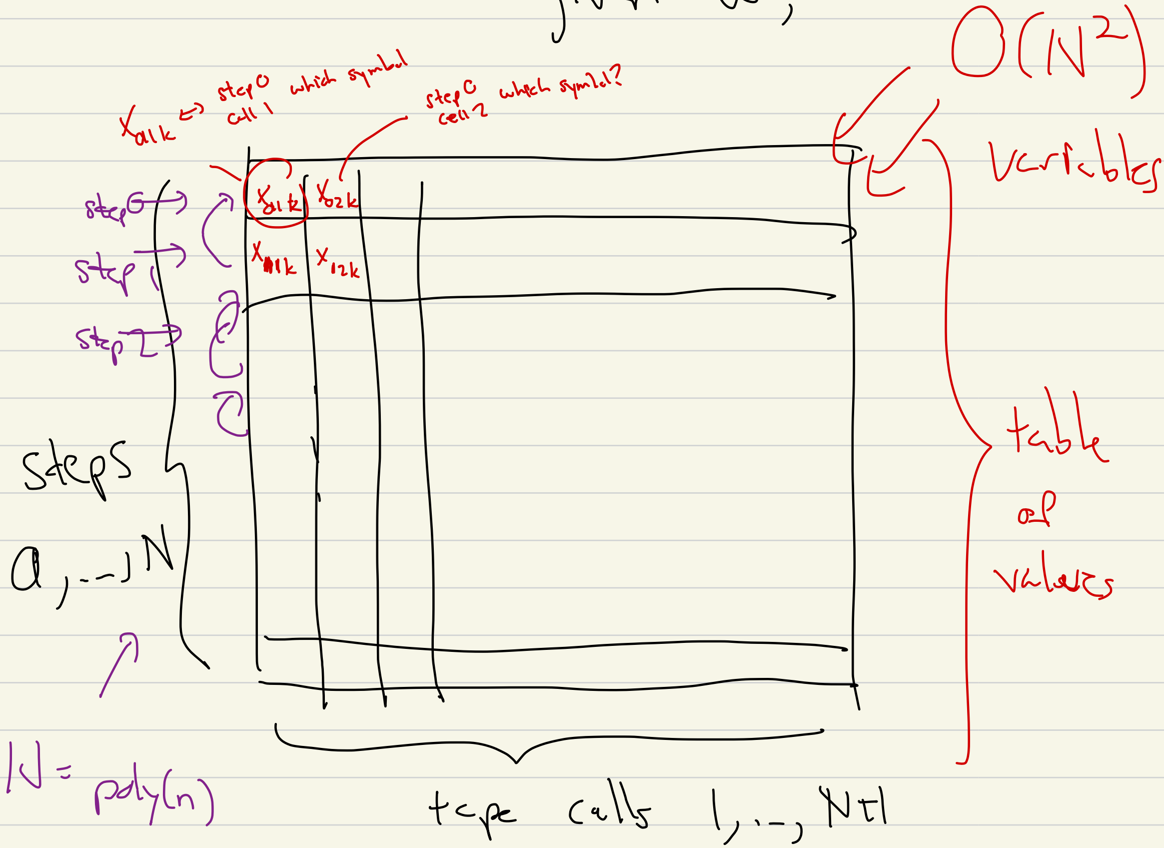
Can't reuse intermediate nodes

Formula $\longleftrightarrow$ Tree ; Circuit $\longleftrightarrow$ Directed Acyclic Graph

Another example:

Say that you have $L \in P$

Cook-Levin Thm: given w,



$$X_{ijk} = \begin{cases} T & \text{if step } i\text{, cell } j \text{ has symbol } k \\ F & \text{otherwise} \end{cases}$$

$\underbrace{X_{ijk}}_{\text{cell contents}}$ # vars: $(N+1)\cdot(N+1)\cdot|\Gamma|$

$\underbrace{Y_{ij}}_{\text{tape head location}}$

$\underbrace{Z_{is}}_{\text{state your time}}$

all these variables

TM non-deterministic $\rightsquigarrow$ SAT, 3SAT instance

TM deterministic!

$\underbrace{X_{i+1,j,k} \quad Y_{i+1,j} \quad Z_{i+1,s}}_{\text{step } i+1}$ j, k, s varying

= some Boolean functions

$\overbrace{X_{ijk}, \cancel{Y}\, Y_{ij}, \quad Z_{i,s}}^{\text{step } i}$

step i-1

step i-2

$\Rightarrow$

$X_{N,j,k}$, $Y_{N,j}$, $Z_{N,s}$

$X_{N-1,j,k}$, - -

X_{N-2}

⋮

$X_{0,j,k}$

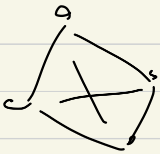
input

most function of n variables require $2^n/n$

$\Rightarrow$ given input, w, $|w| = n$

you have circuit based on $O(|N^2|)$ functions X_{ijk}, Y_{ij}, Z_{is} to see whether or not $w \in L$

If you can prove that given graph size r,



$\binom{r}{2}$ possible edges

$$n = \binom{r}{2}$$

3COLOUR : $\underbrace{T/F}_{} \longrightarrow T/F$

these functions require more than $poly(r) = poly(n)$ circuit gates

$\implies P \neq NP$ $\Leftarrow$ 9.3 of [Sip]