4/19

Friday, April 19, 2019 11:32 AM

Last time: Discussed graphs

We defined the languages

HAMPATH

CLIQUE

We showed they are in NP.

Today: NP-Completeness.

First, we need to discuss Boolean formulas.

Boolean expressions

A Boolean (binary) number is 0 or 1.

Important operations

1) NOT ($\neg$) $\neg 0 = 1$, $\neg 1 = 0$
2) AND ($\wedge$) $\{0 \wedge 0 = 0,\ 0 \wedge 1 = 0$ $\quad 1 \wedge 0 = 0,\ 1 \wedge 1 = 1$
3) OR ($\vee$) $\{0 \vee 0 = 0,\ 0 \vee 1 = 1$ $\quad 1 \vee 0 = 1,\ 1 \vee 1 = 1$

A Boolean variable is a variable x which takes either 0 or 1.

$\mathbb{R} \to \mathbb{Z}/2$.

A Boolean expression ϕ is an expression including Boolean variables and NOTs, ANDs and ORs.

Notation: $\bar{x} = \neg x$
$= \text{opposite of } x$
$= 1 - x$.

Example $(x_1 \vee \bar{x}_2 \vee x_5) \wedge (x_3 \vee \bar{x}_5)$

Example In Boolean variable x, y, z.

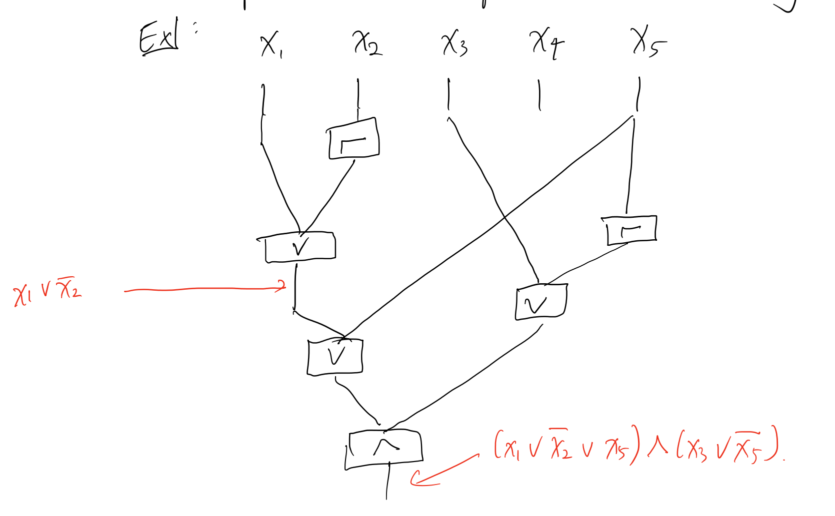
$\phi = x \vee (\bar{x} \wedge y) \vee (\bar{x} \wedge z)$

x	y	z	ϕ
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Satisfiable by $x = 0$, $y = 1$, $z = 1$.

We can represent Boolean expression as circuits using NOT, AND, OR gates

Ex1:



satisfiable

x_1	x_2	x_3	x_4	x_5
1	0	1	0	0

Def^n.

A Boolean expression ϕ in variable $x_1, \ldots, x_n$ satisfiable if $\exists$ assignments of $x_i = 0$ or 1. So that $\phi(x_1, \ldots, x_n) = 1$.

Ex1

$x \wedge \bar{x}$ is not satisfiable.

DEFINE the language.

$SAT = \{\text{Boolean } \phi \text{ which are satisfiable}\}$

Here, we are encoding the Binary expression as a binary string.

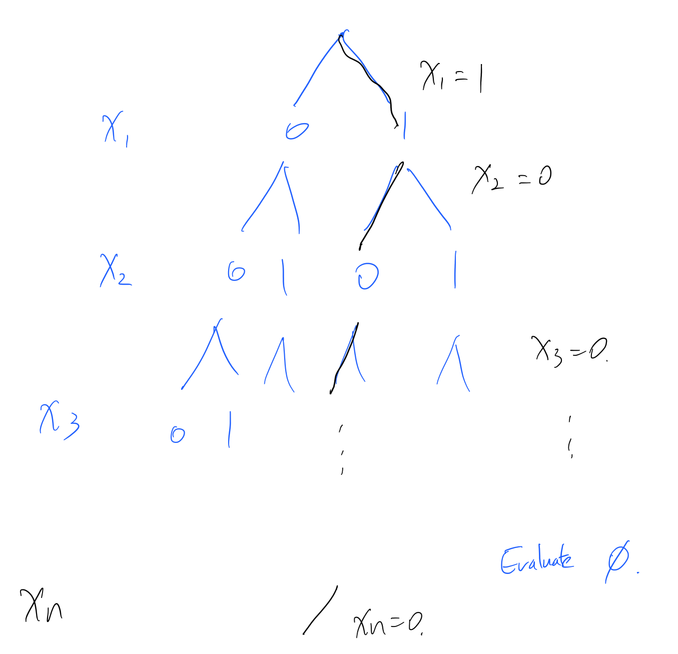
Prop.: $SAT \in NP$

PF: Branch over all possibilities

$x_1 = 0$ or 1, $x_2 = 0$ or $1, \ldots$

check if $\phi(x_1, \ldots, x_n) = 1$

If yes, accept.



running time: $O(n)$

Recall $P \subset NP$

Ques: Is $P = NP$?

Thm (Cook-Levin Thm)

$SAT \in P \iff P = NP$

In other words, if you show $SAT \in P$, you've shown $P = NP$. or $SAT \notin P \Downarrow P \neq NP$.

∗ SAT is as hard as every other problems in NP.

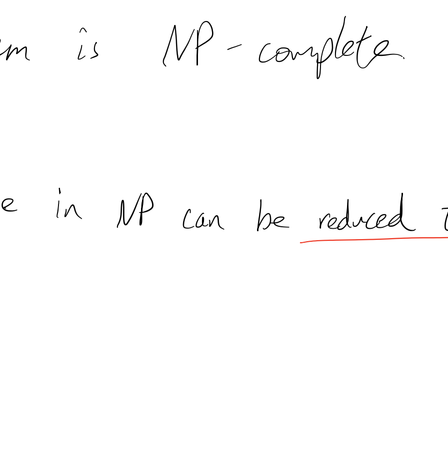
Def^n A language $A \subset \Sigma^*$ is NP complete if the following is true:

$A \in NP$ and

$A \in P \iff P = NP$.

Cook-Levin Thm SAT is NP-complete.

HW: Show HAMPATH, CLIQUE are NP-complete



Ques How can you show that a problem is NP-complete

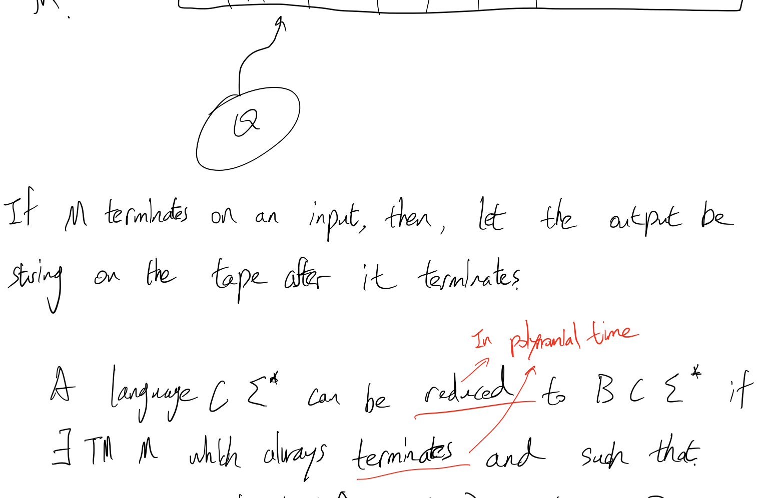
MAIN TECHNIQUE

We'll show that every other language in NP can be reduced to A in polynomial time

Reducing problems to other problems

$x^4 \sin(x^2)$ — chain rules ...

Let's consider a TM



If M terminates on an input, then, let the output be string on the tape after it terminates.

Def^n A language $\subset \Sigma^*$ can be reduced (in polynomial time) to $B \subset \Sigma^*$ if $\exists$ TM M which always terminates (in polynomial time) and such that

input $\in A \iff$ output $\in B$.

Prop. ① If A can be reduced to B,

B decidable $\implies$ A decidable.

② If A can be reduced to B in poly time

$B \in P \implies A \in P$.

PF ① Let M be TM reduces A to B.

If B decidable, let N be a TM decides B.

Design a new TM

Given input w, run M on w.

Run N on the output of M.

$w \in A \xrightarrow{M}$ output $\in B \xrightarrow{N}$ accepts
$w \notin A \xrightarrow{M}$ output $\notin B \xrightarrow{N}$ rejects
$\Big\} = M \circ N$ decides A.

②. Let M be TM reduces A to B in poly time.

If $B \in P$, Let N be a TM decides B in poly time

Design a new TM $\Rightarrow$ in poly time.