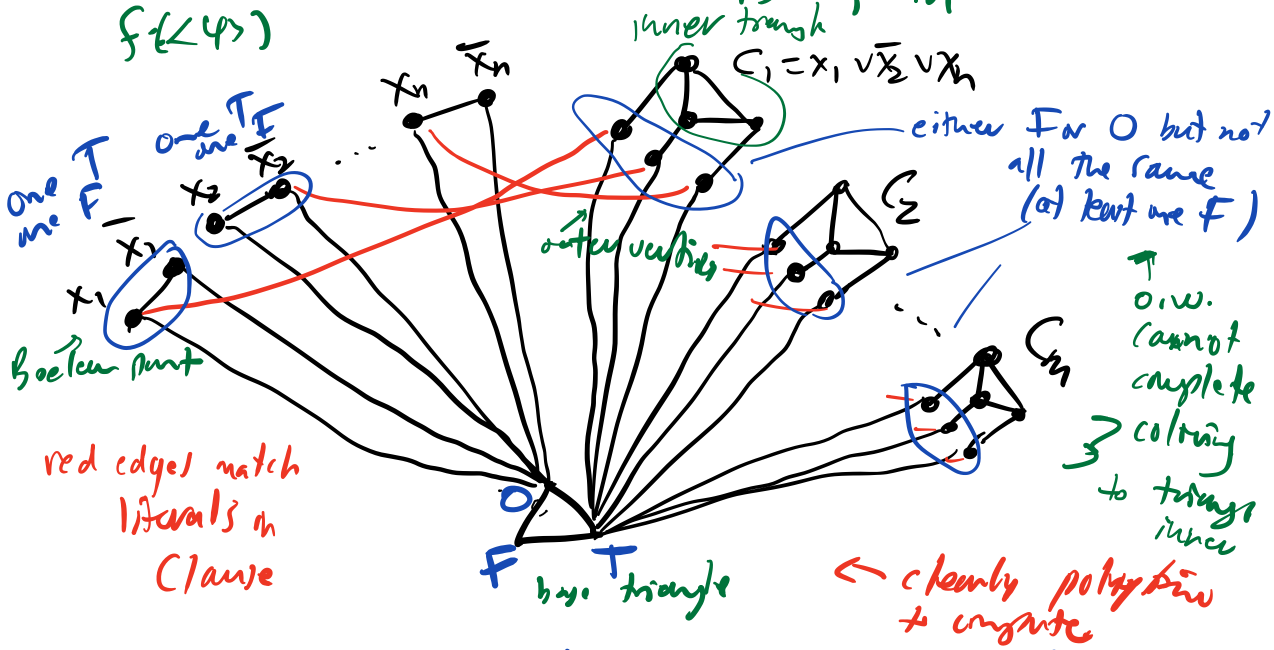
Lecture 20

CSE 431
Intro to Theory of Computation

Last time: $EXACT3SAT \le_m^P 3COLOR$

Given EXACT 3CNF formula $\varphi$ in vars $x_1, \ldots, x_n$ with clauses $C_1, \ldots, C_m$

$f(\langle\varphi\rangle)$




Every 3coloring corresponds to a truth asst for $\varphi$

($\Leftarrow$) if $f(\langle\varphi\rangle)$ is 3-colorable one outer vertex per clause must be an F so corresponding literal node must be a T.
$\therefore$ This has one T literal per clause. so it satisfies $\varphi$.

($\Rightarrow$) if $\varphi$ is SAT, color literal nodes via the truth values. At least one outer node in each clause is opposite a T literal node. color one such outer node with F, the rest with 0 and then complete the inner triangle with 0 opposite the F and T, F opposite the other two

HAMPATH $= \{\langle G \rangle :$ directed graph $G$ has a path touching each node exactly once $\}$

HAMCYCLE $= \{\langle G \rangle :$ - - - — — a cycle - - - - - — — - $\}$

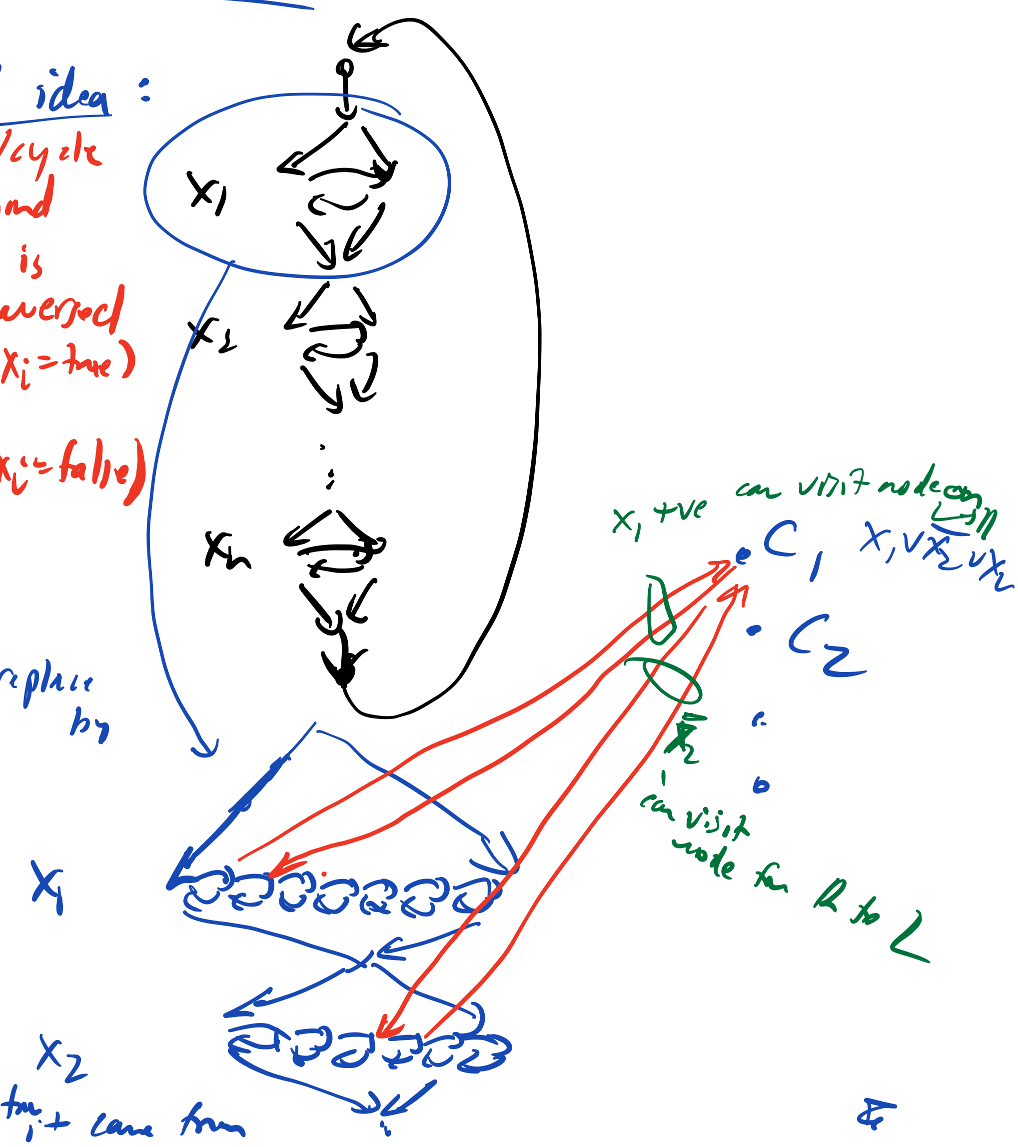
Thm HAMPATH, HAMCYCLE are NP-complete

Pf $\in$ NP easy: each path check - -

$3SAT \leq_m^P$ HAMPATH (CYCLE)

Proof idea:

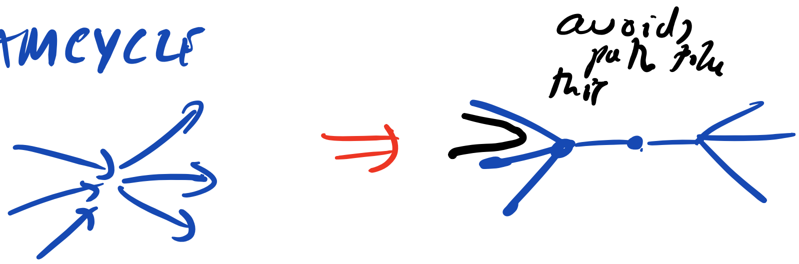
For any path/cycle the diamond for each $x_i$ is either traversed $L \to R$ ($x_i$ = true) or $R \to L$ ($x_i$ = false)




($\Rightarrow$) truth ast can traverse all clauses

($\Leftarrow$) traversal of each $C_i$ node has to return to var it came from

UHAMCYCLE




---

<u>Def</u> $coNP = \{\overline{A} \mid A \in NP\}$ is the analog of coTrec w.r.t NP

coNP problems:
- $UNSAT = \{\langle \varphi \rangle \mid \varphi$ is an unsatisfiable Boolean formula $\}$

almost $\overline{SAT}$ except for the easy case of inputs that are not formulas

- $TAUT = \{\langle \varphi \rangle \mid \varphi$ is a propositional logic tautology $\}$

<u>Note</u>: $\varphi$ is a tautology $\Leftrightarrow \neg\varphi$ is unsatisfiable

$B$ NP-complete $\Rightarrow \forall A \in NP,\ A \leq_m^P B$ & $B \in NP$

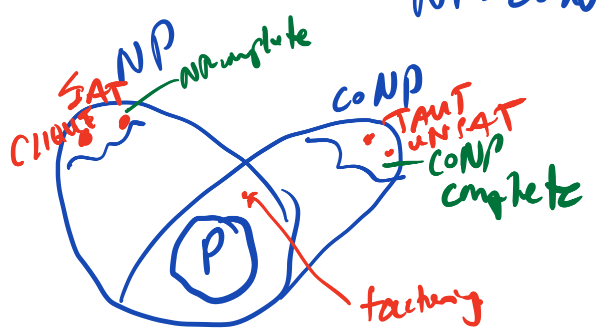
$\Rightarrow \forall \overline{A} \in coNP\ \ \overline{A} \leq_m^P \overline{B}$ & $\overline{B} \in coNP$

$\Rightarrow \overline{B}$ is <u>coNP-complete</u>

TAUT, UNSAT are coNP complete

$\therefore$ $NP \overset{?}{=} coNP$ open

$NP = coNP$ iff every tautology $\varphi$ has a proof of polysize in some proof system




$NP \cap coNP \overset{?}{=} P$

also open

So far we know

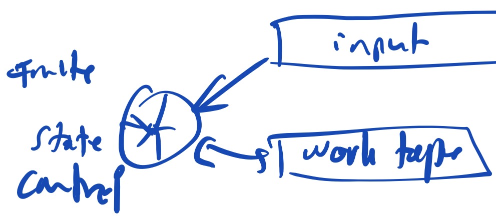
$$P \subseteq \underset{coNP}{NP} \subseteq EXP = \bigcup_k TIME\left(2^{O(n^k)}\right)$$

exponential time

we find an important and natural class of problems in between here

## Space Complexity

We define this using 2-tape NTMs where the input is in read-only memory.




# of work tapes doesn't matter. k-track simulation use bigger alphabet but same # of symbols

**Defn** The space used by such an NTM $M$ is a function $S : \mathbb{N} \to \mathbb{N}$ s.t.

$S(n) = \max\{$ # of work tape cells that $M$ uses on any input $w \in \Sigma^n$ and any computation path $\}$

**Defn** $SPACE(S(n)) = \{A : A$ is decided by TM with read-only input with space used $O(S(n))\}$

$NSPACE(S(n)) = \{A : A$ - - - NTM - - - $O(S(n))\}$

**Note** $A$ is regular $\iff A \in SPACE(1)$

**Thm** $SAT \in SPACE(n)$

**Proof** On input a formula $\langle \varphi \rangle$:

Run brute force algorithm that tries all possible truth assignments $y$ and evaluates $\varphi$ on each one.
$|y| \leq |\langle \varphi \rangle|$. (Copy $\langle \varphi \rangle$ to tape 2)
and reuse space ~~for~~ each a/z t'
total space only a constant factor more than $|\langle \varphi \rangle|$ ∎

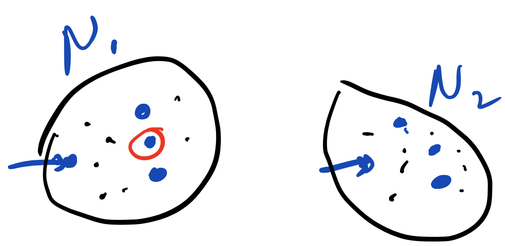
polynomial space

$$PSPACE = \bigcup_k SPACE(n^k)$$

$$NPSPACE = \bigcup_k NSPACE(n^k)$$

**Thm** $\overline{EQ}_{NFA} \in NPSPACE$

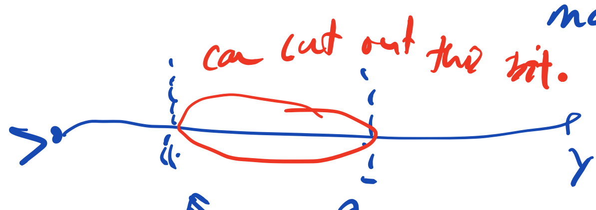
**Proof** On input $\langle N_1, N_2 \rangle$ where $N_1, N_2$ are NFAs with state sets $Q_1, Q_2$ respectively



$L(N_1) \neq L(N_2) \iff \exists$ string $y$ s.t. set of states reachable in $N_1$ on input $y$:
contain a final state of $N_1$
but set of states reachable in $N_2$ on input $y$ does not
(or vice versa)

<u>Claim</u> If such a $y$ exists then one of length $\leq 2^{|Q_1|+|Q_2|}$ exists

<u>Pf of claim.</u> If $y$ is longer than one of the sets of states reachable in the two machines repeats




Idea: Use nondeterminism to guess $y$.
<u>But:</u> $y$ is too long to write down in only $n^{O(1)}$ symbols

<u>Idea:</u> Unlike Time-bounded NTMs can't convert Space-bounded NTMs to guess first form.
- Instead guess $y$ symbol-by-symbol and don't write down the whole thing

Algorithm On input $\langle N_1, N_2 \rangle$
start at $q_0^1, q_0^2$ states of $N_1, N_2$

For $2^{|Q_1|+|Q_2|}$ steps
Guess next symbol of $y$
keeping track of current set of states reached so far on $y$ in both $N_1, N_2$
if one of these sets but not the other contains an accepting state then accept

If $\le 2^{|Q_1|+|Q_2|}$ steps reached but not accepted then reject

Storage
- $|Q_1|+|Q_2|$ bits for sets of states reached
- $|Q_1|+|Q_2|$ bits for a timer.

This is $O(n)$

---

Thm (a) $TIME(T(n)) \subseteq SPACE(T(n))$
$NTIME(T(n)) \subseteq NSPACE(T(n))$

(b) For $S(n) \geq \log_2 n$.
$$SPACE(S(n)) \subseteq TIME(2^{O(S(n))})$$

(c) For $S(n) \geq \log_2 n$
$$NSPACE(S(n)) \subseteq TIME(2^{O(S(n))})$$

Proof (a) If $M$ runs for $T(n)$ steps it can only use $T(n)$ memory cells

(b) Just like algorithm for $A_{LBA}$:

If $M$ has space $S(n)$ then for some $d$ it has only $n \cdot 2^{dS(n)}$ configurations

Why $n \cdot 2^{d\, S(n)}$ ?

read-only input head

possible states × work tape contents and work tape head.

Now for $S(n) \geq \log_2 n$

$$n \leq 2^{S(n)}$$

so total is $2^{d' \cdot S(n)}$

Now just simulate the $S(n)$ space-bounded machine for $2^{d' \cdot S(n)}$ steps if accepted then accept

If not accepted yet then reject

Actually simulation also still takes space $O(S(n))$

- original space + counter

(c) Issues with doing this for NTMs:

- each path has length $2^{d' S(n)}$
- exponential many paths to try

next time.