<u>Today</u>
- Space Complexity III
- Certificate defn of NL
- NL = coNL
- Ramprasad: PH

CSS.203.1
Computational Complexity
- Lecture #9 (15 Mar '21)
Instructor: Prahladh Harsha & Ramprasad Saptharishi

Recall: Space Complexity
L, NL, PSPACE

- Non-deterministic space can be simulated in det-space w/ at most a quadratic overhead (Savitch's Theorem)

- NL : logspace reductions (polynomial time reduction).
  PATH is NL-complete

<u>Today</u> : coNL / $\overline{PATH}$

NP: 2 defns
- {Non-deterministic TMs
- {Verifier

NL : - {Non-deterministic Space TMs
- {Is there a verifier defn for NL?

NP: Verifier defn

$L \in NP$ if $\exists$ a TM M & two poly p, q s.t.

$$x \in L \iff \exists y \in \{0,1\}^{p(|x|)},\ M(x,y) = 1$$

& M runs in time atmost $q(|x|)$.

Similar verifier defn for NL.

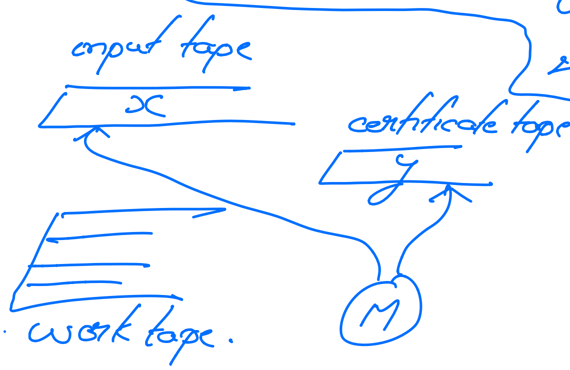
Can, the seq of non-deterministic guesses serve as the certificate y?

Attempt:

$A \in NL'$ if $\exists$ a logspace TM M & a poly $p(\ )$ s.t.

$x \in A \iff \exists y \in \{0,1\}^{p(|x|)}$ s.t. $M(x,y) = 1$

& M runs in logspace on input (x, y)



$3SAT \in NL'$

$\rightarrow$ NL' - too powerful since it can access the entire cert y at any time.

Fix: Make certificate tape READ-ONCE

$A \in NL$ iff $\exists$ a 2-input tape Machine M
(1st input tape - standard read only i/p.
(2nd input tape - read once certificate tape)
& a poly $p(\ )$ s.t.

$$x \in A \iff \exists y \in \{0,1\}^{p(|x|)} \quad M(x,y) = 1$$

& M runs in logspace.

$\overline{PATH} = \{(G,s,t) \mid G$ is a directed graph & there is no path from s to t in $G\}$.

Theorem [Immerman - Szelepcsényi]

$\overline{PATH} \in NL$

Hence, $NL = coNL$

Pf:

s - source vertex

t - target vertex

$C_i = \{v \in V(G) \mid \exists$ a path of length at most i from s to $v\}$.

$C_0 = \{s\}$, $C_n = \{v \mid s \rightsquigarrow v\}$

NL-certificate: "$t \notin C_n$"

"$v \notin C_i$" $\forall\ i \in \{0, .. n\}$ & $v \in V(G)$

(a) Assume $|C_i|$, then there exists an NL-certificate that "$v \notin C_i$".

(b) Assume $|C_{i-1}|$, there exists an NL-certificate for "$|C_i| = c$".

(a). Verifier knows $|C_i| = c$

$C_i := \{v_{i_1} < v_{i_2} < \dots < v_{i_c}\}$

NL-certificate for "$v \notin C_i$" (given $|C_i| = c$)

Certificate for "$v \notin C_i$" :

v_{i_1} - path $s \rightsquigarrow v_{i_1}$

$\wedge$

v_{i_2} - path $s \rightsquigarrow v_{i_2}$

$\wedge$

$\vdots$

$\wedge$

v_{i_c} path $s \rightsquigarrow v_{i_c}$

and the indices of v_{i_j} are increasing

&

$v \notin \{v_{i_1} \dots v_{i_c}\}$

(a). Given $|C_i|$, NL-certificate for "$v \notin C_i$"

(b)': Given $|C_{i-1}|$, NL-certificate for "$v \notin C_i$"

Same as above except check that $v \notin \{v_1 \ldots v_c\}$ & v is not adjacent to $\{v_1 \ldots v_c\}$.

(b) Given $|C_{i-1}|$, NL-certificate for "$|C_i| = c$"

v_1 - $v_1 \notin C_i$ - NL-cert

$\wedge$

v_2. $v_2 \in C_i$ - NL-cert

$\vdots$

$\wedge$

v_n $v_n \notin C_i$.

Hence, $\overline{PATH} \in NL$.

ie, $coNL = NL$.

For any $\mathcal{C}$

$co\mathcal{C} \subseteq \mathcal{C}$.

$\Updownarrow$

$\mathcal{C} \subseteq co\mathcal{C}$.