Space-Bounded Computation

- Input tape: read-only, for free
- working tape: read/write, incurs cost
- output tape: write-only, one-way, for free

$\rightarrow$ *streaming computation*

Def: $A \leqslant_{\mathrm{L}} B$ iff there is some log-space computable $f:\Sigma^*\rightarrow\Sigma^*$ s.t. $\underline{x}\in A \Leftrightarrow f(\underline{x})\in B$.

Call A **NL**-hard if every $B\in$ **NL** satisfies $B \leqslant_{\mathrm{L}} A$,

NL-complete if in addition $A\in$ **NL** holds.

L ⊆ NL ⊆ P ⊆ NP ⊆ PSPACE=NPSPACE ⊆ EXP

Reductions

- polynomial-time *many-one*: $A \leqslant_p^m B$

  $f:\Sigma^*\to\Sigma^*$ $\mathbf{P}$-computable s.t.: $\underline{x}\in A \Leftrightarrow f(\underline{x})\in B$

  – in general $B \not\leqslant_p^m \Sigma^*\setminus B$ (example?)
- log-spacebounded *many-one*: $A \leqslant_L^m B$

  $f:\Sigma^*\to\Sigma^*$ $\mathbf{L}$-computable s.t.: $\underline{x}\in A \Leftrightarrow f(\underline{x})\in B$

  – $A \leqslant_L^m B \Rightarrow A \leqslant_p^m B$ (proof?)
- polynomial-time *Turing* Reduction: $A \leqslant_p^T B$

  A can be solved in polynomial time by aid of oracle queries to *B*: $A \in \mathbf{P}^B$

  – $A \leqslant_p^m B \Rightarrow A \leqslant_p^T B \Rightarrow A \leqslant_p^T \Sigma^*\setminus B$ (proof?)
- Further reductions: *truthtable*, parsimonious…

Theorem: dirPath is **NL**-complete

Let $A \in \mathbf{NL}$, decided by $c \cdot \log n$ space-bounded NTM M

Input: $\underline{w}$; output: dir.Graph G and vertices s,t such that:

M accepts $\underline{w}$ $\Leftrightarrow$ there is a path in G from s to t

$G=(V,E)$, V:=all configurations of M of size $c \cdot \log |\underline{w}|$

$(K_1,K_2) \in E$ $:\Leftrightarrow$ K_2 is a successor config of K_1

s:=start config of M on $\underline{w}$; t:=accept.config (wlog unique)

- M accepts $\underline{w}$ $\Leftrightarrow$ there is a path in G from s to t √
- How large is G? Constructible in logarithmic space?

qed

Immerman-Szelepcsényi

L = NL ? **=coNL ?** **= P ?** Compare „**P** vs. **NP**"

L vs. **NL**: **NL**-complete dirGraph, 2unSAT, nonBipartite

NL vs. **P**: **P**-complete problems

(probably do not admit an efficient parallelization)

NL vs. **coNL**: solved in 1987,

ACM Gödel Prize 1995 !

Theorem (Neil Immerman,

Róbert Szelepcsényi): NL=coNL

Proof: Show dirGraph $\in$ **coNL**

Theorem: For $s(n) \geq \log n$ increasing,

NSPACE($s(n)$)=**coNSPACE**($s(n)$).

dirGraph∈ **coNL**

Given $G=(V,E)$, $s,t\in V=\{1,\ldots,m\}$. Goal:
Logspace NTM accepts iff t not reachable from s.
$A_i := \{ v\in V :$ exists path in G of length $\leq i$ from s to $v\}$,
$c_i := \#A_i$, $i=0,\ldots,m-1$. $A_0=\{s\}$, $c_0=1$. Accept iff $t\notin A_{m-1}$

Definition: NTM computes (partial) $f:\subseteq\Sigma^*\to\Sigma^*$ iff

- $\forall$ inputs $\underline{x}\in \mathrm{dom}(f)$ there is an accepting computation.
- Every accepting computation outputs $f(\underline{x})$.

FNL is closed under composition! (proof?)

Lemma: For each i, $A_i\in$ **NL**.

Given (!) c_i, logspace NTM can even <u>enumerate</u> A_i:

- For each $v\in V$, 'guess' whether $v\in A_i$ (1) or not (0)
- If guessed 1: output, verify (**NL**) and increase counter
- In the end, accept iff counter$=c_i$!!!

dirGraph $\in$ **coNL**

Given $G=(V,E)$, $s,t\in V=\{1,\ldots,m\}$. Goal:
Logspace NTM accepts iff t not reachable from s.
$A_i := \{ v\in V :$ exists path in G of length $\leq i$ from s to $v \}$,
$c_i := \#A_i$, $i=0,\ldots,m\text{-}1$. $A_0=\{s\}$, $c_0=1$. Accept iff $t\notin A_{m-1}$

Lemma: For each i, $A_i\in$ **NL**.
Given (!) c_i, logspace NTM can even enumerate A_i.
Lemma: Given (!) c_i, $A_{i+1}\in$ **coNL**:
Enumerate A_i and, if no edge to v found, accept.
Lemma: Given c_i, logspace NTM can compute c_{i+1}:
For each v, 'guess' whether $v\in A_{i+1}$ holds, and verify
Proof (Theorem): $1=c_0 \rightarrow c_1 \rightarrow c_2 \rightarrow c_3 \rightarrow \ldots \rightarrow c_{m\text{-}1}$