601 Lect. 17: Recall From Last Time

Thm: REACH is complete for **NL**.

Proof:

$$w \in \mathcal{L}(N) \quad \Leftrightarrow \quad \mathrm{CompGraph}(N, w) \in \mathrm{REACH} \qquad \Box$$

Space Hierarchy Thm: Let $f(n) \geq \log n$ be a space constructible function. If

$$\lim_{n \to \infty} \left(\frac{g(n)}{f(n)} \right) = 0$$

Then, $\mathbf{DSPACE}[g(n)] \subsetneq \mathbf{DSPACE}[f(n)]$.

Proof: Diagonalize against all machines using space $f(n)$ and time $2^{f(n)}$. $\Box$

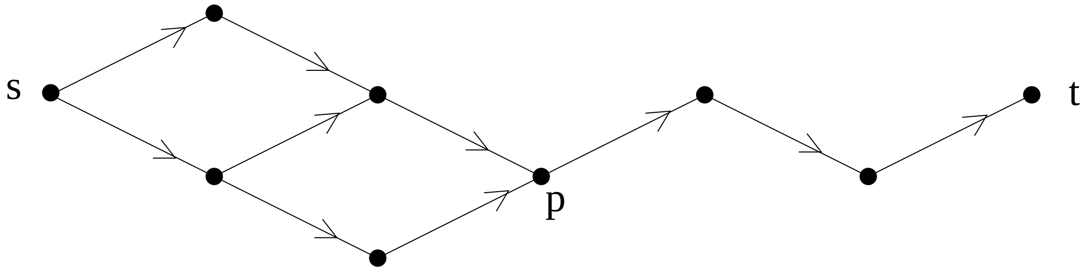
NSPACE vs. DSPACE

Prop:

$$\mathbf{NSPACE}[s(n)] \;\subseteq\; \mathbf{NTIME}\mathbf{[}2^{O(s(n))}\mathbf{]} \;\subseteq\; \mathbf{DSPACE}[2^{O(s(n))}]$$

We can do much better!

Thm: $\text{REACH} \in \textbf{DSPACE}[(\log n)^2]$



Proof:

$$
\begin{aligned}
G \in \text{REACH} \quad &\Leftrightarrow \quad G \models \text{PATH}(s,t,n) \\
\text{PATH}(x,y,1) \quad &\equiv \quad x = y \ \vee \ E(x,y) \\
\text{PATH}(x,y,2d) \quad &\equiv \quad \exists z\, (\text{PATH}(x,z,d) \ \wedge \ \text{PATH}(z,y,d))
\end{aligned}
$$

$S_n(d)$ = space to check paths of dist. d in n-nodegraphs

$$
\begin{aligned}
S_n(n) \quad &= \quad \log n + S_n(n/2) \\
&= \quad O((\log n)^2)
\end{aligned}
$$

☐

Savitch's Thm: For $s(n) \geq \log n$,

$$\textbf{DSPACE}[s(n)] \ \subseteq\ \textbf{NSPACE}[s(n)] \ \subseteq\ \textbf{DSPACE}[(s(n))^2]$$

Proof: Let $A \in \textbf{NSPACE}[s(n)]$; $\quad A = \mathcal{L}(N)$

$$w \in A \qquad \Leftrightarrow \qquad \mathrm{CompGraph}(N, w) \in \mathrm{REACH}$$

$$|w| = n; \qquad |\mathrm{CompGraph}(N, w)| = 2^{O(s(n))}$$

Testing if $\mathrm{CompGraph}(N, w) \in \mathrm{REACH}$ takes space,

$$\begin{aligned} (\log(|\mathrm{CompGraph}(N, w)|))^2 &= (\log(2^{O(s(n))}))^2 \\ &= O((s(n))^2) \end{aligned}$$

From w build $\mathrm{CompGraph}(N, w)$ in $\textbf{DSPACE}[s(n)]$. $\square$

Thm: $\overline{\mathrm{REACH}} \in$ **NL**

Proof: Fix G, let $N_d = \left|\{v \mid \text{distance}(s, v) \le d\}\right|$

Claim: The following problems are in **NL**:

1. $\mathrm{dist}(x, d)$: distance$(s, x) \le d$
2. $\mathrm{NDIST}(x, d; m)$: if $m = N_d$ then $\neg\mathrm{dist}(x, d)$

Proof:

1. Guess the path of length $\le d$ from s to x.
2. Guess m vertices, $v \neq x$, with $\mathrm{dist}(v, d)$.

```
$c := 0$;
for $v := 1$ to $n$ do {    // nondeterministically
    ( $\mathrm{dist}(v, d)$ && $v \neq x$; $c{+}{+}$ )    ||
    ( no-op )
}
if $(c == m)$ then ACCEPT
```

□

Claim: We can compute N_d in **NL**.
Proof: By induction on d.

Base case: $N_0 = 1$

Inductive step: Suppose we have N_d.

```
$c := 0$;
for $v := 1$ to $n$ do {    // nondeterministically
            ( dist$(v, d+1)$; $c++$)    ||
            ($\forall z\,(\mathrm{NDIST}(z, d; N_d) \vee (z \neq v \wedge \neg E(z, v)))$)
}
$N_{d+1} := c$
```

$\square$

$$G \in \overline{\mathrm{REACH}} \quad \Leftrightarrow \quad \mathrm{NDIST}(t, n; N_n)$$

$\square$

Immerman-Szelepcsényi Thm:

If $s(n) \geq \log n$, Then, $\textbf{NSPACE}[s(n)] \ = \ \text{co-}\textbf{NSPACE}[s(n)]$

Proof: Let $A \in \textbf{NSPACE}[s(n)]$; $\quad A = \mathcal{L}(N)$

$$w \in A \qquad \Leftrightarrow \qquad \text{CompGraph}(N, w) \in \text{REACH}$$

$$|w| = n; \qquad |\text{CompGraph}(N, w)| = 2^{O(s(n))}$$

Testing if $\text{CompGraph}(N, w) \in \overline{\text{REACH}}$ takes space,

$$\begin{aligned} \log(|\text{CompGraph}(N, w)|) &= \log(2^{O(s(n))}) \\ &= O(s(n)) \end{aligned}$$

☐

Review of Reductions

Def: S is **reducible** to T, $(S \le T)$, iff $\exists f \in F(L)$ such that, $\forall w \in \mathbf{N}\,(w \in S \;\Leftrightarrow\; f(w) \in T)$.

$$A_{0,17} \quad = \quad \{n \mid M_n(0) = 17\}$$

Claim: $K \le A_{0,17}$.

Proof: Define $f(n)$ as follows:

$$M_{f(n)} \;=\; \boxed{\begin{array}{c}\text{erase input}\\ \text{write } n\end{array}}\boxed{M_n}\boxed{\begin{array}{c}\text{if 1 then write 17}\\ \text{else loop}\end{array}}$$

$$n \in K \quad \Leftrightarrow \quad M_n(n) = 1 \quad \Leftrightarrow \quad M_{f(n)}(0) = 17$$

□

$$A_{0,17} = \{n \mid M_n(0) = 17\} \qquad K = \{n \mid M_n(n) = 1\}$$

Let's look at that equivalence in more detail:

$$M_{f(n)} = \boxed{\begin{array}{c}\text{erase input}\\ \text{write } n\end{array}}\boxed{M_n}\boxed{\begin{array}{c}\text{if 1 then write 17}\\ \text{else loop}\end{array}}$$

Claim: $n \in K \Leftrightarrow M_n(n) = 1 \Leftrightarrow M_{f(n)}(0) = 17 \Leftrightarrow f(n) \in A_{0,17}$

Proof: $(\Rightarrow)$:

$$n \in K \Rightarrow M_n(n) = 1 \Rightarrow M_{f(n)}(0) = 17 \Rightarrow f(n) \in A_{0,17}$$

def of K def of $f(n)$ def of $A_{0,17}$

$(\Leftarrow)$:

$$f(n) \in A_{0,17} \Rightarrow M_{f(n)}(0) = 17 \Rightarrow M_n(n) = 1 \Rightarrow n \in K$$

def of $A_{0,17}$ def of $f(n)$ def of K

$\square$

Fundamental Theorem of Reductions:

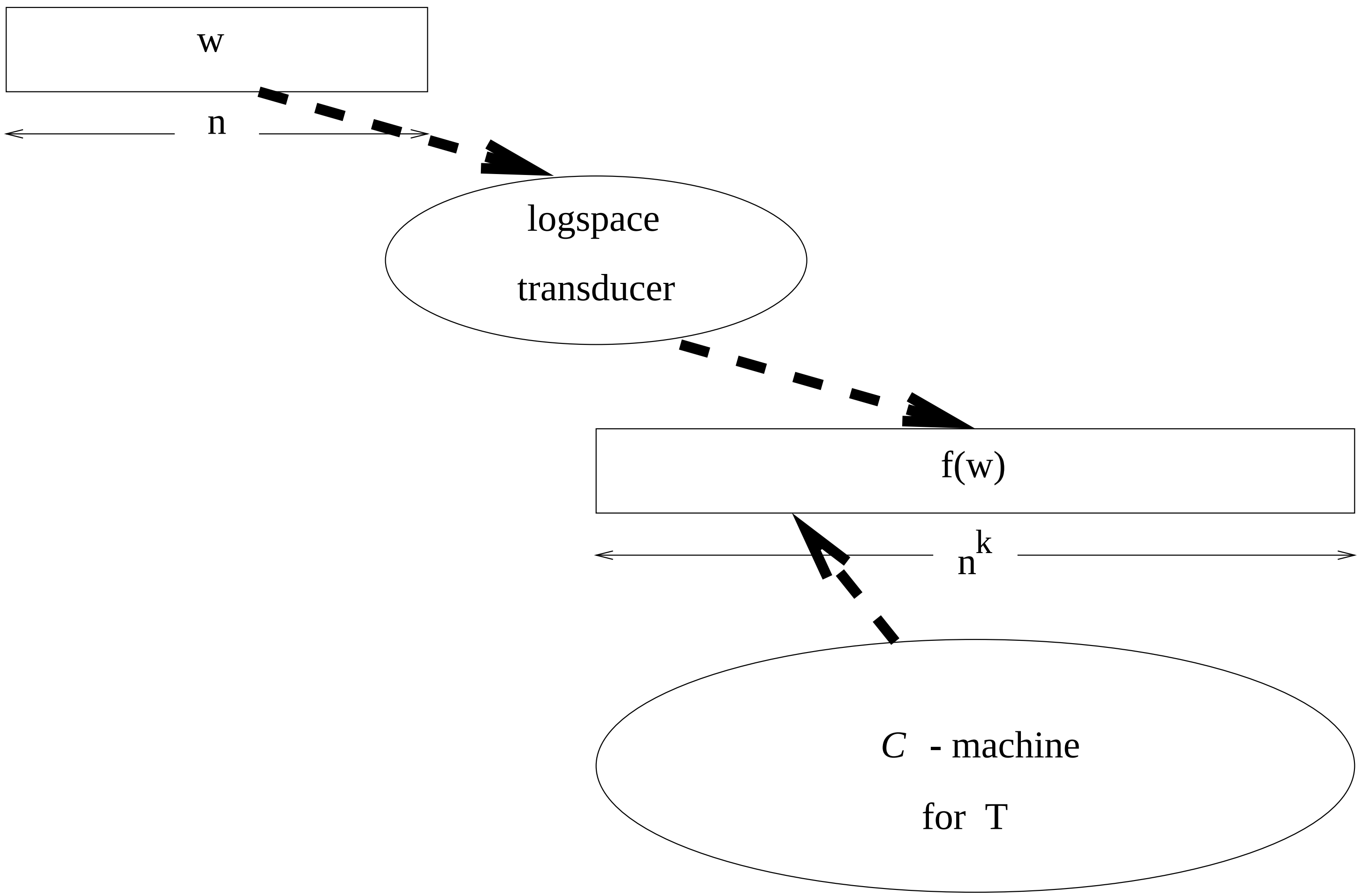
Let $\mathcal{C}$ be one of the following complexity classes: **L**, **NL**, **P**, **NP**, co-**NP**, **PSPACE**, **EXPTIME**, **Primitive Recursive**, **Recursive**, **r.e.**, co-**r.e.**.

Suppose $S \leq T$.

If $T \ \in \mathcal{C}$ Then $S \ \in \ \mathcal{C}$

That is, each of these classes is **closed under reductions**.

Proof: Suppose that $S \leq T$ and $T \in \mathcal{C}$.
We build a $\mathcal{C}$ machine for S: $\quad w \in S \quad \Leftrightarrow \quad f(w) \in T$



w

n

logspace

transducer

f(w)

n^k

C - machine

for T

Reductions are Useful for:

Lower Bounds:
If A is hard and $A \leq B$ Then B is hard.

Upper Bounds:
If B is easy and $A \leq B$ Then A is easy.