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

Proof:

$$w \in \mathcal{L}(N) \quad \Leftrightarrow \quad \text{CompGraph}(N, w) \in \text{REACH} \quad \spadesuit$$

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

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

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

Proof: Diagonalize against all machines using space $3f(n)$ and time $2^{3f(n)}$. ♠

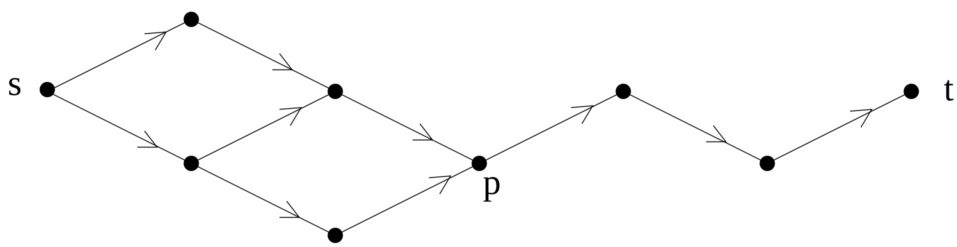
We stated but did not prove the similar *Time Hierarchy Theorem*.

Proposition 17.1

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

We can do much better!

Theorem 17.2 $\quad \text{REACH} \in \textbf{DSPACE}[(\log n)^2]$



Proof:

$$G \in \text{REACH} \Leftrightarrow G \models \text{PATH}(s, t, n)$$

$$\text{PATH}(x, y, 1) \equiv x = y \vee E(x, y)$$

$$\text{PATH}(x, y, 2d) \equiv (\exists z)(\text{PATH}(x, z, d) \wedge \text{PATH}(z, y, d))$$

$S(n, d)$ = space to check paths of distance d in graphs with n nodes.

$$S(n, n) = \log n + S(n, n/2)$$

$$= O((\log n)^2)$$

♠

This can be thought of as a *middle-first search* algorithm for REACH, efficient for deterministic space but lousy for time.

```
boolean isPath (vertex s,
                vertex t, int dist) {
   if (s == t) return true;
   if (dist == 1) return (edge(s, t));
   else for (vertex u = 0; u < n; u++)
      if (isPath (s, u, dist/2) &&
          isPath (u, t, dist - dist/2))
         return true;
   return false;}
```

The call `path(s,t,n-1)` recurses to a depth of at most $\log n$. Each recursive call needs $O(\log n)$ bits on the stack. The running time may be as bad as $n^{\log n}$ since there are in effect $\log n$ nested loops.

Corollary 17.3 (Savitch's Theorem) *For* $s(n) \geq \log n$*,*
$\mathbf{DSPACE}[s(n)] \ \subseteq\ \mathbf{NSPACE}[s(n)] \ \subseteq\ \mathbf{DSPACE}[(s(n))^2]$

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

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

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

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

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

From w build $\text{CompGraph}(N,w)$ in **DSPACE**[s(n)]. ♠

Along with regular languages and CFL's, there is another old-time complexity class called the *context-sensitive languages* that turns out to be **NSPACE**(n). It was asked whether this class is closed under complement (like the regular languages) or not (like the CFL's). Since the general intuition was that they were not, no one looked for a proof that they were. The problem remained open for about twenty years.

In 1987 two researchers, Neil Immerman in the US and Richard Szelepcsényi in Slovakia, simultaneously found a proof that nondeterministic space classes *are* closed under complement. Not only that, the proof is easy to present.

The reason for the simultaneity is probably that a series of results just before this began to suggest that the result might be true. Neil reports that he got the basic idea while out walking his dog.

Theorem 17.4

$$\overline{\text{REACH}} \in \textbf{NL}$$

Proof:

Fix graph G.

$$N_d = |\{v \mid v \text{ reachable from } s \text{ using} \leq d \text{ edges}\}|$$

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

1. $\text{DIST}(x, d)$: $\text{distance}(s, x) \leq d$
2. $\text{NDIST}(x, d; m)$: if $m = N_d$ then $\neg\text{DIST}(x, d)$

Proof:

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

♠

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.

1. $C := 0$;
2. **for** $v := 1$ to n **do** $\{$
3. **OR** $\{\mathrm{DIST}(v, d+1);\quad C++;$
4. $(\forall z)(\mathrm{NDIST}(z, d; N_d) \vee (z \neq v \wedge \neg E(z, v)))$
5. $\}$
6. $\}$
7. $N_{d+1} := C$

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

$\spadesuit$

Corollary 17.5 (Immerman-Szelepcsényi Theorem) *Let* $s(n) \geq \log n$. *Then,*

$$\textbf{NSPACE}[s(n)] \quad = \quad co\text{-}\textbf{NSPACE}[s(n)]$$

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,

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

♠

Definition 17.6 We say that S is *reducible* to T, $S \leq T$, iff $\exists\ f \in F(L)$ such that,

$$(\forall w \in \mathbf{N}) \quad (w \in S) \qquad \Leftrightarrow \qquad (f(w) \in T)$$

♠

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

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

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

$M_{f(n)}$ =	erase input write n	M_n	if 1 then write 17 else loop

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

♠

Theorem 17.7 *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.

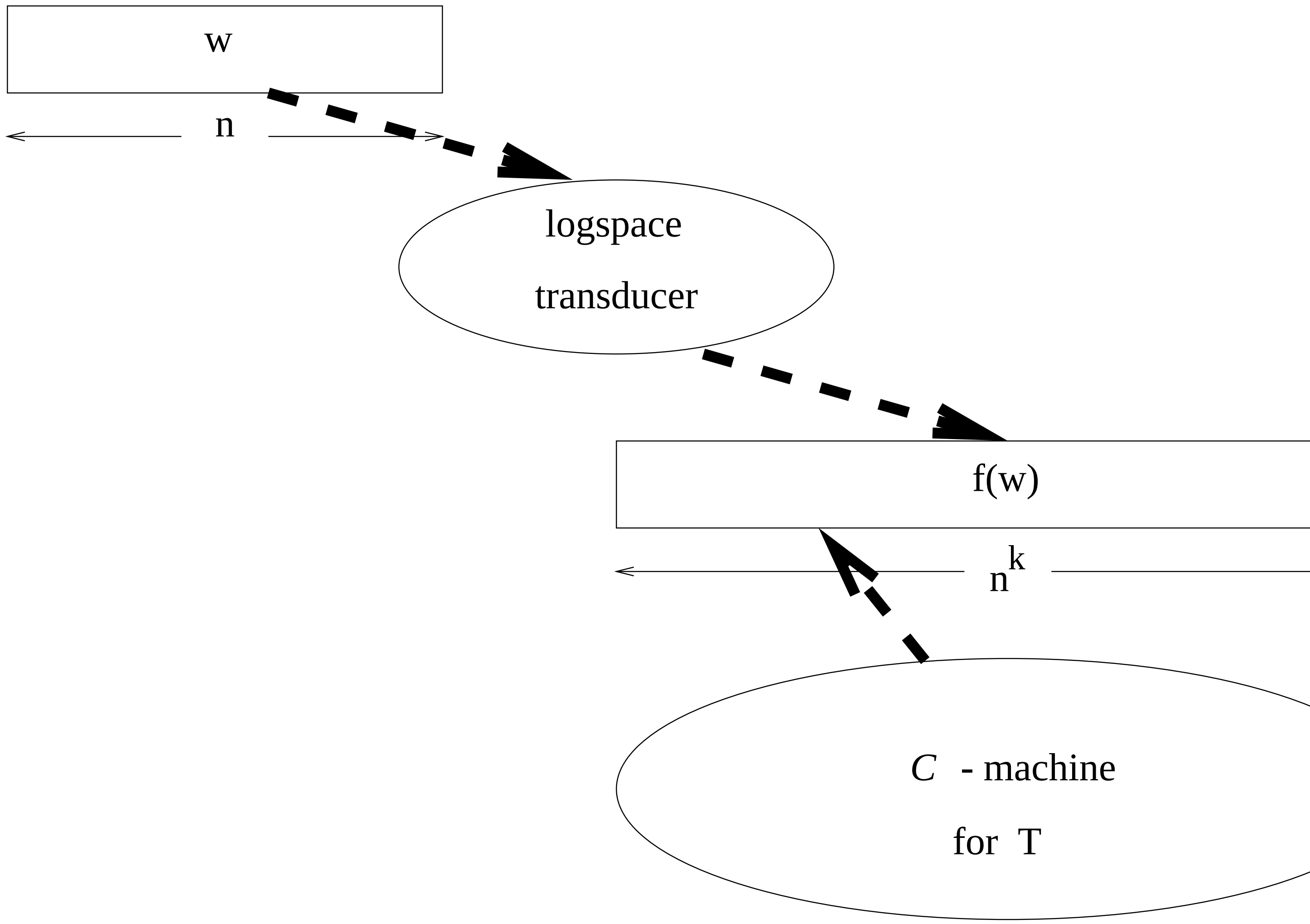
Suppose $S \leq T$.

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: $\quad w \in S \quad \Leftrightarrow \quad f(w) \in T$



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.

A Nontrivial Fact About Logspace Reductions:

If $A \leq B$ and $B \leq C$, then $A \leq C$.

It looks obvious at first, but draw the picture of the two machines! The output tape of the first machine becomes the input tape of the second. In the two original machines neither counts against the space bound, but in the new machine *this becomes a worktape*!

On HW#6 you'll prove that this fact is actually true.