1.3 Relations between space and time Complexity

We relate DSPACE, NSPACE, DTIME, NTIME.

1.3.1. Space versus time

Time-efficient simulations of space-bounded computations:

Proposition 23.

If a 2-T-NTM M accepts a language A in space $s(n) \geq \log n$ then M accepts A in time $2^{O(s)}$.

Theorem 24.

Let $s(n) \geq \log n$ be space-constructible. Then,

$$NSPACE(s) \subseteq DTIME(2^{O(s)}).$$

Proof: Let M_1 be a 2-T-NTM accepting A in space s.
Let M_2 be a 2-T-DTM constructing space $s(|x|)$ on input x.
By Lemma ~~9~~ 5, M_2 takes at most $2^{c \cdot s(|x|)}$ steps for some $c > 0$.
If M_1 accepts x on some computation path then there is a comp. path of length $\leq 2^{c \cdot s(|x|)}$ (otherwise there are configurations occurring twice on such paths, cut everything in between out).
We apply dynamical programming: Is k_{acc} reachable from k_{init}?
We consider a list of all poss. configurations of M_1 with space $\leq s(|x|)$; use red color to label configuration k reachable from k_{init} with successor configuration not yet labeled; use green color to label k reachable from k_{init} and all successors labeled.
W.l.o.g. k_{acc} is unique for each x.

Define M to be that TM that, on input x, works as follows:

(1) M runs like M_2 and computes $s(|x|)$

(2) M writes all conf. of M_1 with space $\le s(|x|)$ on tape 1

(3) M labels K_{init} with red color

(4) <u>while</u> there is a red conf. on tape 1 <u>do</u>

(a) M labels leftmost red conf. K with green color

(b) M labels all not-yet-labeled successor conf. of K with red color

(5) M accepts $x \iff K_{acc}$ is a green conf.

Analysis of time complexity:

(1) $\le 2^{c_1 s(|x|)}$ steps

(2) $\underbrace{c_2 \cdot s(|x|)}_{|conf.|} \cdot \underbrace{2^{c_1 s(|x|)}}_{\#\,conf} \le 2^{c_3 \cdot s(|x|)}$

(3) $\le 2^{c_3 \cdot s(|x|)}$

(4) $\le \underbrace{2^{c_1 s(|x|)}}_{iterations} \cdot c_4 \cdot 2^{c_3 s(|x|)}$

(5) $\le 2^{c_3 s(|x|)}$

Overall, sum is $\le 2^{c_5 \cdot s(|x|)}$. $\blacksquare$

<u>Corollary 25.</u>

(1.) $NL \subseteq P$.

(2.) $NLIN \subseteq E$.

(3.) $NPSPACE \subseteq EXP$.

Space-efficient simulations of time-bounded computations:

It holds $DTIME(t) \subseteq DSPACE(t)$, $NTIME(t) \subseteq NSPACE(t)$

Theorem 26.

Let $t(n) \geq n$ be space-constructible. Then,

$$T\text{-}NTIME(t) \subseteq DSPACE(t).$$

Proof: Let M_1 be a T-NTM accepting A in time t.

Let M_2 be a 2-T-DTM constructing space $t(|x|)$ on input x.

Let r be maximum branching degree of M_1, i.e., comp. path is sequence from $\{1,\dots,r\}^{t(|x|)}$.

Define M to be a DTM that, on input x, works as follows:

(1.) M runs like M_2 and writes $1^{t(|x|)}$ on tape 1

(2) until content of w.t. 1 equals $\underbrace{r^{t(|x|)}}_{rr\cdots r}$ do

- (a) M simulates comp. path of M_1 given on tape 1
- (b) if M_1 accepts x then M accepts x and halts
- (c) increment comp. path on w.t. 1

(3.) M rejects x

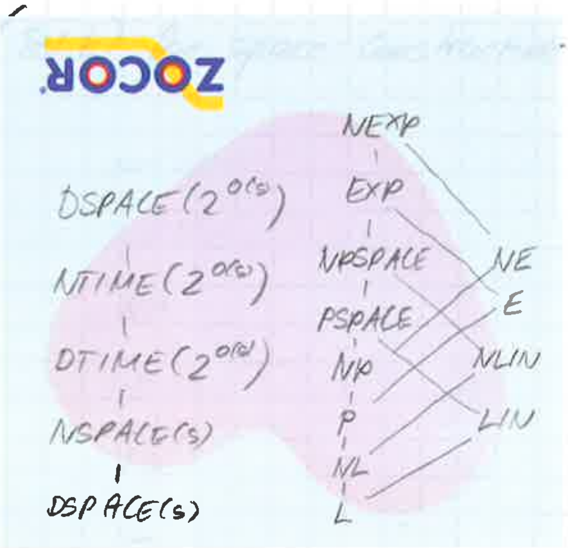
Space complexity of M: $\leq 2 \cdot t(|x|)$, with compression $\leq t(|x|)$.

Corollary 27.

(1.) $NTIME(Pol\ t) \subseteq DSPACE($

$t(n) \geq n$.

(2.) $NP \subseteq PSPACE$



1.3.2 Nondeterministic space versus determ. space

Lemma 28.

For all space-constructible $s(n), t(n) \geq \log n$, the following holds: If a language A is accepted by a 2-T-NTM in space s and time 2^t then L is accepted by a 2-T-DTM in space $s \cdot t$.

Proof later.

Theorem 29. (Savitch 1970)

Let $s(n) \geq \log n$ be space-constructible. Then,

$$NSPACE(s) \subseteq DSPACE(s^2).$$

Proof: Let M be an NTM accepting A in space s. So, M accepts A in time $2^{c \cdot s}$. By Lemma 28, there is a DTM accepting A in space $s \cdot (c \cdot s) = c \cdot s^2$, with compression in space s^2. ■

Corollary 30.

(1.) $NL \subseteq DSPACE(\log^2 n)$

(2.) $NLIN \subseteq DSPACE(n^2)$

(3.) $NPSPACE = PSPACE$.

Proof of Lemma 28:

Let M be a 2-T-NTM accepting A in space s and time 2^t.
On input x, there are at most $2^{c \cdot s(|x|)}$ conf. of M;
$k_1, k_2, \ldots, k_{2^{c s(|x|)}}$. Let k_{init} be the initial conf.; let k_{acc} accepting halting conf. (w.l.o.g. unique).

We want to answer the following question: Is k_{acc} reachable from k_{init} within $2^{t(|x|)}$ steps?
(That is; $x \in A \iff$ answer is „yes")

Define following predicate:

$R(i,j,\tau) = 1 \iff_{def}$ M on input x reaches k_j from k_i within 2^τ steps

That is: $x \in A \iff R(k_{init}, k_{acc}, t(|x|)) = 1$

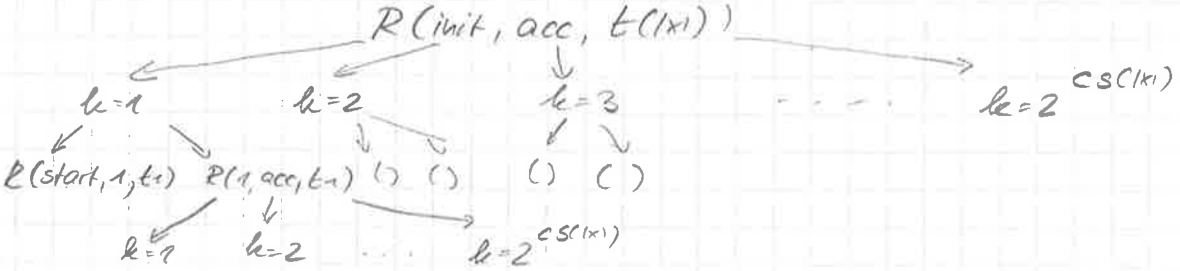
How to compute $R(i,j,\tau)$?

Idea: $R(i,j,\tau) = 1 \iff (\exists k)\left[R(i,k,\tau-1) = R(k,j,\tau-1) = 1\right]$

Consider recursive algorithm for $R(i,j,\tau)$:

(1) $R := 0$
(2) if $\tau = 0$ then
(3) if k_j is successor of k_i then $R := 1$ else $R := 0$
(4) else
(5) for $k = 1, 2, \ldots, 2^{c s(|x|)}$ do
(6) if $R(i,k,\tau-1) = R(k,j,\tau-1) = 1$ then $R := 1$

Algorithm applied to $(k_{init}, k_{acc}, t(|x|))$:

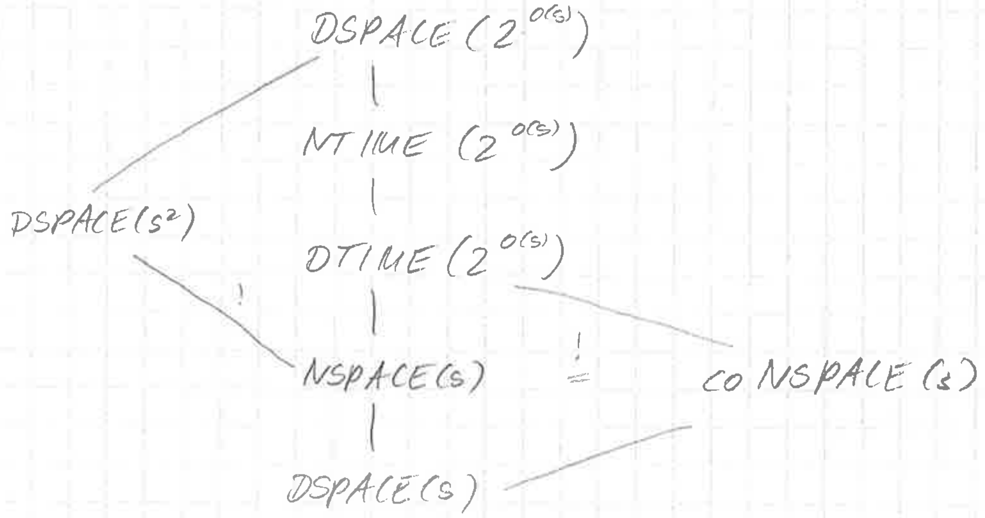


For traversing the tree using (preorder) depth-first search, it is enough to store conf. k for each level

So, space complexity is

$\leq$ |configuration| $\cdot$ heigth of tree

$\leq s(|x|) \cdot t(|x|)$



1.3.3. Complementing nondeterministic space

For each class $\mathcal{K}$, define

$$\mathrm{co}\mathcal{K} =_{\mathrm{def}} \{ \overline{A} \mid A \in \mathcal{K} \}$$

Remark: To show $\mathcal{K} = \mathrm{co}\mathcal{K}$, it is enough to show $\mathcal{K} \subseteq \mathrm{co}\mathcal{K}$ or $\mathrm{co}\mathcal{K} \subseteq \mathcal{K}$.

Theorem 31. (Szelepczényi, Immerman 1987)

Let $s(n) \geq \log n$ be space-constructible. Then,

$$\mathrm{coNSPACE}(s) = \mathrm{NSPACE}(s).$$

Proof (Inductive counting)

$\boxed{2}$ Let $A \in \mathrm{NSPACE}(s)$, i.e., there is 2-T-NTM M accepting A in space $s(n)$. Let $k_1, k_2, \dots, k_{2^{cs(|x|)}}$ be all poss. conf. of M on input x. Let k_{init}, k_{acc} denote the unique initial, acc. halt. conf.

Define $N =_{\mathrm{def}} 2^{c \cdot s(|x|)}$

We use $k \vdash_t^{M,x} k'$ to denote the fact that M, on input x, reaches k' from k in $\leq t$ steps. So,

$$M \text{ accepts } x \iff k_{init} \vdash_N^{M,x} k_{acc}.$$

Further define

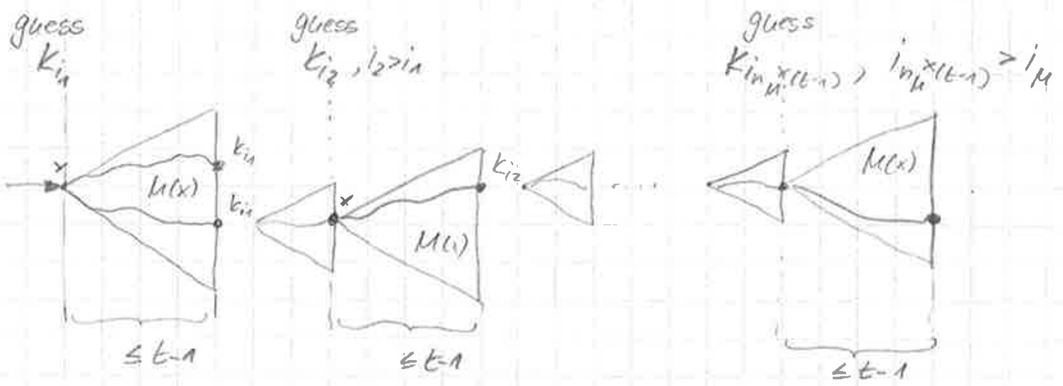
$$n_M^x(t) =_{\mathrm{def}} \| \{ j \mid 1 \leq j \leq N \wedge k_{init} \vdash_t^{M,x} k_j \} \|$$

Define M' to be a 2-T-NTM that, on input x, works in $N+1$ stages and computes $n_M^x(t)$ for $t \in \{0, 1, \dots, N\}$:

- stage 0: $n_M^x(0) = 1$.
- stage $t > 0$: Suppose we know value $n_M^x(t-1)$.

  How to check if some k is among $n_M^x(t-1)$ conf. reachable in $t-1$ steps?

Consider the following sub-TM that re-computes all $n_M^x(t-1)$ conf. in ascending order



So, for $j = 1, \dots, N$: $\quad m = n_M^x(t-1)$

- M' guesses predecessor conf. $k_1', \dots, k_m'$ of k_j.
- M' runs sub-TM above to check if one of the conf. k_i' is reachable in $\leq t-1$ steps
- M' adds k_j to $n_M^x(t)$, if so

Finally, we know the value $n_M^x(t)$.

- stage $N+2$:
  - M' runs sub-TM above to check if k_{acc} is reachable in $\leq N$ steps.
  - M' accepts x iff k_{acc} is <u>not</u> reachable

Correctness: note that there is exactly one comp. path of sequence along with correct numbers $n_M^x(0), n_M^x(1), \dots, n_M^x(N)$. So,

$$M \text{ accepts } x \iff k_{init} \vdash_N^{M,x} k_{acc}$$
$$\iff M' \text{ rejects } x$$

That is, M' accepts $\bar{A}$.

Space complexity: $\leq \underbrace{2 \cdot c \cdot s(|x|)}_{|\text{config.}|} + \underbrace{\log t}_{|\text{clock}|} \leq_{ae} c' \cdot s(|x|)$

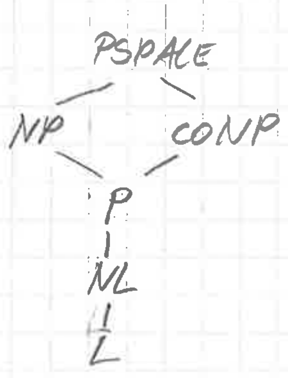
Corollary 32.

(1.) $co\, NL = NL$

(2.) $co\, NLIN = NLIN$.

1.3.4 Open problems in complexity theory

General Problem	Special cases
$DSPACE(s) \overset{?}{=} NSPACE(s)$	$\boxed{L \overset{?}{=} NL}$ $LIN \overset{?}{=} NLIN$
$NSPACE(s) \overset{?}{=} DTIME(2^{O(s)})$	$\boxed{NL \overset{?}{=} P}$ $NLIN \overset{?}{=} E$ $PSPACE \overset{?}{=} EXP$
$DTIME(Pol\, t) \overset{?}{=} NTIME(Pol\, t)$	$\boxed{P \overset{?}{=} NP}$ $E \overset{?}{=} NE$ $EXP \overset{?}{=} NEXP$
$NTIME(Pol\, t) \overset{?}{=} DSPACE(Pol\, t)$	$\boxed{NP \overset{?}{=} PSPACE}$ $NE \overset{?}{=} DSPACE(2^{O(s)})$ $NEXP \overset{?}{=} DSPACE(2^{Pol\, s})$
$NTIME(Pol\, t) \overset{?}{=} coNTIME(Pol\, t)$	$\boxed{NP \overset{?}{=} co\, NP}$ $NE \overset{?}{=} co\, NE$ $NEXP \overset{?}{=} co\, NEXP$



Theorem 33.

Let s be space-constructible. Let t be time-constructible, i.e., $t \circ len \in FDTIME(Pol\, t)$.

(1.) $L = NL \Rightarrow DSPACE(s) = NSPACE(s)$ for $s(n) \geq \log n$

(2.) $NL = P \Rightarrow NSPACE(s) = DTIME(2^{O(s)})$ for $s(n) \geq \log n$

(3.) $P = NP \Rightarrow DTIME(Pol\, t) = \overset{NTIME}{\cancel{DSPACE}}(Pol\, t)$ for $t(n) \geq n$

(4.) $NP = PSPACE \Rightarrow NTIME(Pol\, t) = DSPACE(Pol\, t)$ for $t(n) \geq n$

(5.) $NP = coNP \Rightarrow NTIME(Pol\, t) = coNTIME(Pol\, t)$ for $t(n) \geq n$

<u>Proof</u>: Use padding technique.

(1.) Choose $r =_{def} 2^s$. We have: PACE(s).

? A, $A \in NSPACE(s) \Rightarrow A_r \in NSPACE(\log s) = NL$

$\Rightarrow A_r \in DSPACE(\log s) = L$

$\Rightarrow A \in DSPACE(s)$

(2) - (5.) Analogous. ∎

Corollary 34.

(1.) $L = NL \Rightarrow LIN = NLIN$

(2.) $NL = P \Rightarrow NLIN = E \Rightarrow PSPACE = EXP$

(3.) $P = NP \Rightarrow E = NE \Rightarrow EXP = NEXP$

(4.) $NP = PSPACE \Rightarrow NE = DSPACE(2^{O(n)})$

$\Rightarrow NEXP = DSPACE(2^{Pol\, n})$