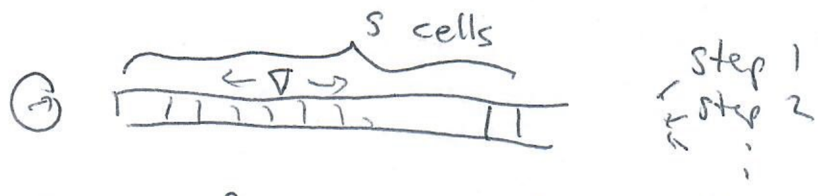
- 8.1, 8.2 (Using an old argument)

Savitch's Thm ⟵ PSPACE = NPSPACE

m : 1-tape TM, deterministic or not



How many configurations of the TM if we only look at the first s cells?

configurations possible : (# states) (s possible tape head positions) $|\Gamma|^s$

$$\boxed{\begin{pmatrix}\text{upper bound for}\\ \text{any space} \le s\\ \text{configuration}\end{pmatrix} \le C_1 \cdot s \cdot C_2^{\,s}}$$

$\left(\begin{matrix} C_1 = |Q| \\ C_2 = |\Gamma| \\ \text{depends on } m \end{matrix}\right)$

Way back when - -

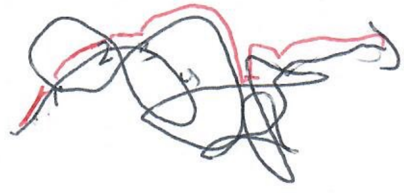
TIME any computation $\ge$ SPACE

TIME any accepting computation $\le C_1 (\text{space})\, C_2^{(\text{space})}$

Reason: In a deterministic halting computation, you see each configuration at most once.

Remark: In a non-det computation, the shortest accepting comput path that accepts any input doesn't see same configuration twice

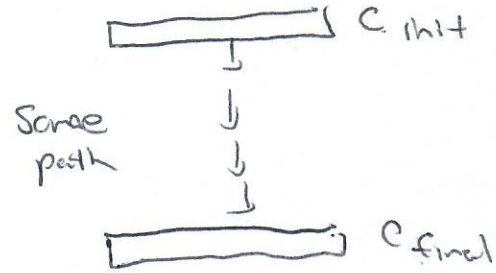
Configuration Set

Progress  vs. 

Regardless of input w, fixed M,

Configs in space s $\leq C_1 \, s \, C_2^{\,s}$

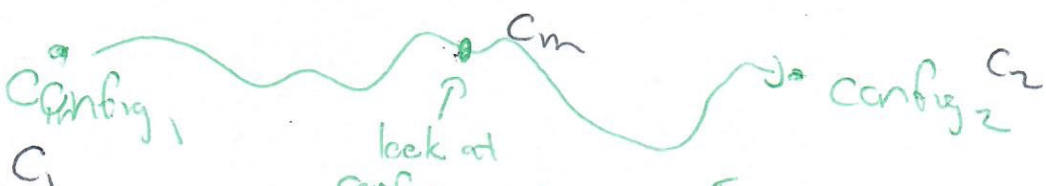
Say fix non-det TM M, on input w, M takes space $\leq S$. If M accepts w



$\Big\}\ t \leq C_1 \, S \, C_2^{\,S}$ ← # steps

Claim: There is a deterministic computation that can check whether one can start at c_{init} and end at <u>some</u> accepting config, in space $C_3 \cdot S^2$.

Recursion:



Canyield (c_1, c_2, t) := we can go from config c_1 to config c_2 in t steps ~~or less~~ or fewer

then

$$\text{Canyield}(c_1,c_2,t) = \bigvee_{c_m \in \text{Configs}} \left(\text{Configs}(c_1, c_m, t/2) \text{ AND } \text{Configs}(c_m, c_2, t/2) \right)$$

Idea: C_{init} —— C_{middle} —— C_{final} (3)

given

remember

element of all configs

some subset of all configs

recurse over all possible c_m

Space of configs that take space $\le S$ store

$\overbrace{\text{state} \mid \text{tape head located} \mid \text{cell 1} \mid \text{cell 2} \mid \cdots \mid \text{cell } S}^{\text{order } S \text{ space (!)}}$

state: int from 1 to $|Q|$; tape head located: int from 1 to S; cell 1: $\in \Gamma$; cell 2: $\in \Gamma$; ...

Remember C_{init}, C_{final}, C_m each take $O(S)$ <u>space</u>

Phase 1: Canyield$(C_{init}, C_m, t/2)$

Phase 2: Canyield$(C_m, C_{final}, t/2)$

Phase 1:

C_{init} —— $C_{new\ middle}$ —— C_m steps $t/2$

some config

$C_{init} \xrightarrow{t/4} C_{new\ middle}$ Remember

$C_{new\ middle} \xrightarrow{t/4} C_m$

$t/8$ $t/8$ $t/8$ $t/8$

Claim: Each config takes $\boxed{O(S)}$ space. # config to remember is $\le \log_2(t)+1$; but $\log_2(t) \le \log_2(C_1 S C_2^S) = \boxed{O(S)}$

Total space is $O(S^2)$.

Cor: $NSPACE(n^{10}) \subset SPACE(n^{20})$, etc. $NPSPACE = PSPACE$.