Left shifting machine:

$q_0 \# w_1 w_2 \ldots w_n \rightarrow q_{acc} w_1 w_2 \ldots w_n$

1. Move right and read character x
2. If $x = '\sqcup'$
   - Move left
   - Write '$\sqcup$'
   - stop
3. If $x \neq '\sqcup'$
   - Move left
   - Write x
   - move right
   - goto 1.

TM which copies a string

$q_0 w_1 w_2 \cdots w_n \longrightarrow q_{acc} w_1 w_2 \cdots w_n w_1 w_2 \cdots w_n$

$\underline{w_1} w_2 \cdots w_n \rightsquigarrow \underline{w_1} w_2 \cdots w_n \#$

$\rightsquigarrow w_1 \underline{w_2} \cdots w_n \# w_1$

$\rightsquigarrow \underline{w_1} w_2 \cdots w_n \# w_1 \cdots w_n$

$\rightsquigarrow w_1 w_2 \cdots w_n \underline{\#} w_1 \cdots w_n$

$\downarrow$ leftshift

$q_{acc} w_1 \cdots w_n w_1 \cdots w_n$

No of operations: $O(|w|^2)$

Ex 3.11 (short version)

$C = \{a^i b^j c^k \mid i \cdot j = k, i, j, k \geq 1\}$

$\cancel{a}$aa bb cccccc

Check for youselves!

Next step: other variants of TM's

Model 1: several tapes

k Tape TM: $\underline{\triangleright} w_1 w_2 \cdots w_n$

$\underline{\triangleright}$

$\underline{\triangleright}$

tape k $\underline{\triangleright}$

NB: # tapes is fixed (k for a k-tape TM)

$\delta(q\ a_1, a_2 \cdots a_k) = (p, b_1, \ldots, b_k, \gamma_1, \gamma_2 \cdots \gamma_k)$

why could several tapes be useful

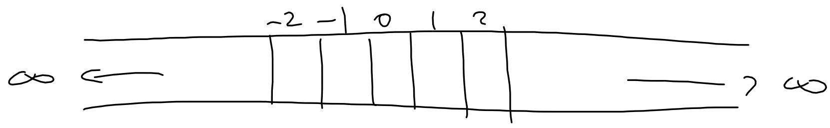
ex copy:

$$\underline{\triangleright} w_1 w_2 \cdots w_n \qquad \triangleright w_1 w_2 \cdots w_n \underline{\sqcup}$$
$$\underline{\triangleright} \quad \rightsquigarrow \quad \underline{\triangleright} w_1 w_2 \cdots w_n$$

$$\rightsquigarrow \triangleright w_1 \cdots w_n w_1 \cdots \underline{w_n} \qquad \underline{\triangleright} w_1 \cdots w_n w_1 \cdots w_n$$
$$\triangleright w_1 \cdots \underline{w_n} \quad \rightsquigarrow \quad \underline{\triangleright}$$

$O(|w|)$ steps

<u>Model 2</u> 2-way ∞ tape



<u>Model 3</u> Non-deterministic TM (NDTM)

loosely: a NDTM can guess

from state q on an a we may have up to $B = |Q| \cdot |\Gamma| \cdot 3$

$\Leftarrow (R, L, S)$

ex where non-determinism is useful:

Given $n \in \mathbb{Z}^+$

is n composite ($\exists?\ n_1, n_2 \geq 2$ s.t. $n_1 \cdot n_2 = n$)

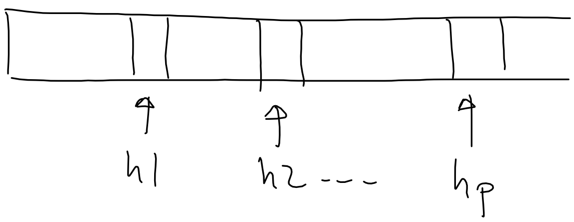
The NDTM guesses n_1 and n_2

calculates $m = n_1 \cdot n_2$ and

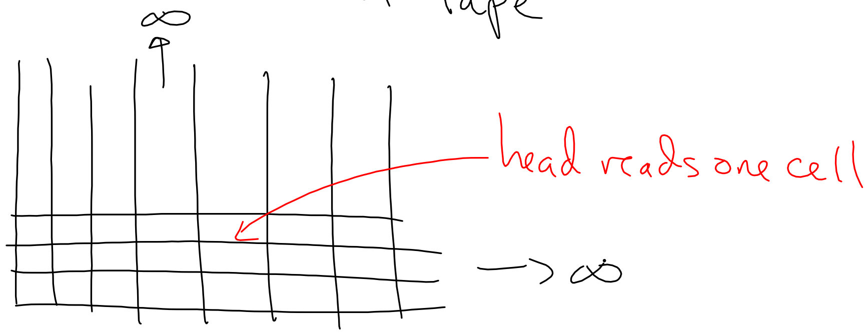
checks whether $m = n$.

Model 4 One tape and several heads

p-head TM

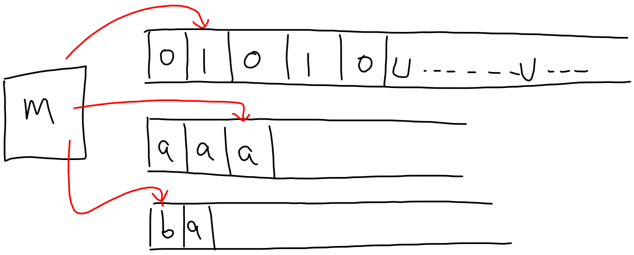


Model 5 2 dimensional tape

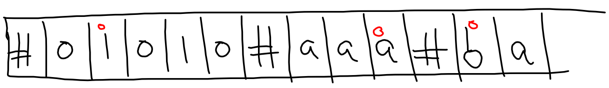


We want to argue that none of these models are stronger than standard TM model, so in particular if $L = L(M)$ for M being e.g a k-tape TM then $\exists$ stand TM M' s.t $L = L(M')$

Simulating a 3-tape TM with a standard TM



M'



read overall description in Sipser.
We concentrate on how to implement one step of M.

<u>Note</u> that if M moves into '⊔' area on any tape, M' can simulate this using shift right (several or on time)

<u>k-tapes</u> for original state q^i of M

We have states $q^i_{(\alpha_1,\dots,\alpha_k)}$ $\alpha_i \in \Gamma \cup \{\_\}$

and states $p^i(\dots\dots, b_1, b_2 \dots b_h, \gamma_1 \dots \gamma_k)$

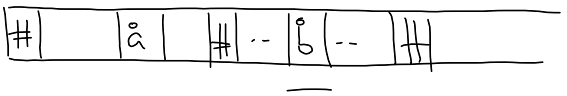
If we are in state $q^i_{(\beta_1 \dots \beta_r, \dots)}$ then
We have collected the symbols being read
on the first r tapes.

<u>Note</u> the # states above is huge, BUT
at most some function $f(k, |\Gamma|)$

How to implement one step of M
(step 2 page 177)

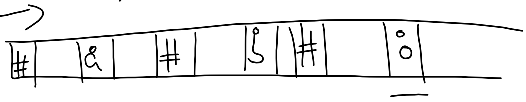
in M $\delta(q_i, a_1 \dots a_k) = (q_j, b_1 \dots b_k, \gamma_1 \dots \gamma_k)$

1. M' starts in state $q^i_{(-,-,\dots,-)}$
2. In state $q^i_{(\beta_1 \dots \beta_r, -, \dots)}$ move head forward to copy of r+1'st tape that is



$q^i_{(a,b,-,-)}$

move right until



shift to state
$q^i_{(a,b,o,\dots)}$

When we reach $q^i_{(\beta_1 \dots \beta_k)}$ we all head characters

move to state $p^j(-,\dots,-, b_1, b_2 \dots b_k, \gamma_1, \gamma_2 \dots \gamma_k)$

In state $p^j_{(b_1,\ldots b_s,\ldots,b_1\ldots b_k,\gamma_1\ldots\gamma_k)}$

move to position of $(s+1)$'st head:

write b_{s+1} and 'move head' according to γ_{s+1}

$\rightsquigarrow$

Finally from state $p^j_{(b_1\ldots b_k,b_1\ldots b_k,\gamma_1\ldots\gamma_k)}$

move to state $q^j_{(-\ldots-)}$

Running time for simulation:

suppose takes t steps on W

- no tape can contain more than t new symbols (M writes atmost one new / step)

simulating one steps takes

- $O(|W|+kt)$ if no shift right call
- $O(k^2t+|W|)$ if we need one or more shiftright

So simulating t steps takes

$$O(k^2t^2+|W|t)$$