MTAT.05.125 Introduction to Theoretical Computer Science

Autumn 2015

Vitaly Skachek
Estonian version by Reimo Palm
English version by Yauhen Yakimenka

Week 10. Turing machines: equivalent models.

Note. Since we use the term "Turing machine" a lot, it will be occasionally shortened as TM.

Variants of Turing machines

What happens if we modify the transition function $\delta\colon Q\times\Gamma\to Q\times\Gamma\times\{\mathsf{L},\mathsf{R}\}$ to be $\delta\colon Q\times\Gamma\to Q\times\Gamma\times\{\mathsf{L},\mathsf{R},\mathsf{S}\}$, where S means "stay on the same place". Does this generalisation allow Turing machines to recognise more languages?

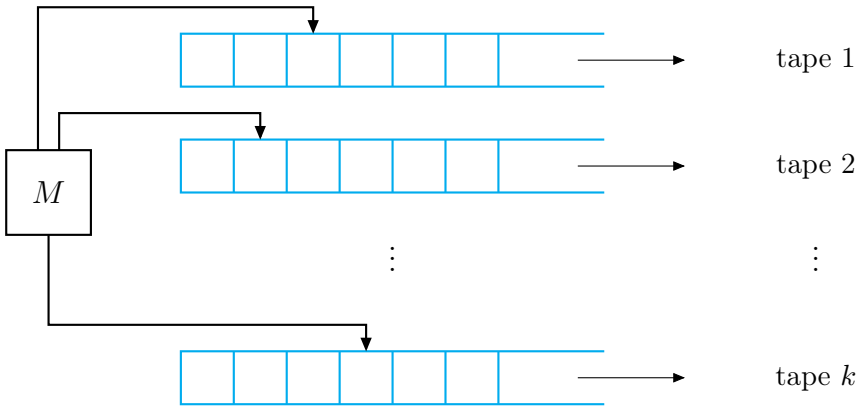
The answer is no. We can convert any Turing machine with "stay on the same place" option into the original model with left and right movements only. We achieve this by replacing "stay on the same place" command by moving to the right (and into a special state), and then moving back to the left.

Turing machines with multiple tapes

It is Turing machine with several read-write tapes (and heads). More formally:

$$\delta\colon Q\times\Gamma^k\to Q\times\Gamma^k\times\{\mathsf{L},\mathsf{R},\mathsf{S}\}^k,$$

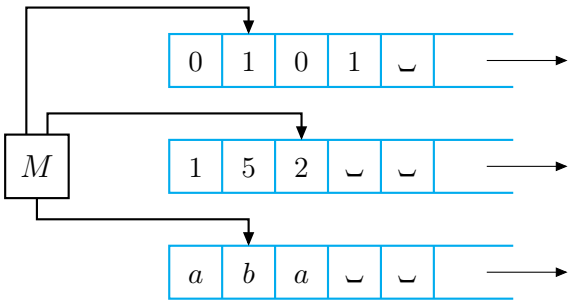
where k is the number of tapes. Now $\delta(q_i,a_1,\ldots,a_k)=(q_j,b_1,\ldots,b_k,\mathsf{L},\mathsf{R},\ldots,\mathsf{L})$ means that if heads $1,2,\ldots,k$ read symbols $a_1,a_2,\ldots,a_k$ respectively, the the machine moves to the state q_j and writes $b_1,b_2,\ldots,b_k$ to the respective tapes.

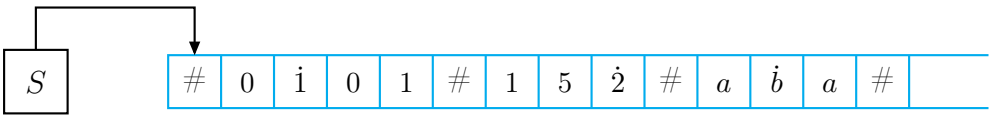


Theorem. *Every multi-tape Turing machine has an equivalent single-tape Turing machine.*

Proof. We convert a multi-tape TM M into an equivalent single-tape machine S.

If M has k tapes, then S simulates these k tapes by storing their content on a single tape. It uses the new special symbol # as a delimiter to separate the content of different tapes. Additionally, it should keep track of the positions of the heads. It is achieved by writing special "dotted" symbols, which will be added to the alphabet.





On input $w = w_1 \dots w_n$, S does the following:

1. First, S puts its tape into the format that represents all k tapes on M. The formatted tape contains[1]

$$\#\dot{w}_1 w_2 \dots w_n \# \dot{\sqcup} \# \dot{\sqcup} \# \dots \#$$

[1]We assume tapes $2, 3, \dots, k$ are empty in the beginning.

2. To simulate a single move, S scans its tape from the first # to the $(k+1)$-st #, in order to determine the symbols pointed by the "virtual heads". S makes the second pass to update the new tapes according to the way that M would do.

3. If at some point S moves one of the "virtual heads" to the right onto #, this means that M has moved the corresponding head onto the previously unread blank position of the tape. Then, S writes a blank symbol on this tape, and shifts the tape content from this place until the right-most #. Then it continues the simulation as before.

□