

COSE215: Theory of Computation

Lecture 17 — Extensions of Turing Machines

Hakjoo Oh
2019 Spring

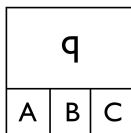
Extensions

Extend the standard Turing machine with

- ① storage in the state
- ② multiple tracks
- ③ a stay-option
- ④ multiple tapes
- ⑤ non-determinism

Storage in the state

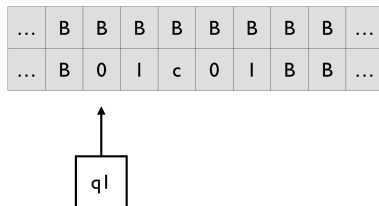
The finite control stores a finite amount of data:



Example

A Turing machine that accepts $01^* + 10^*$.

Multiple Tracks



Example

A Turing machine that accepts $L = \{w cw \mid w \in \{0, 1\}^+\}$.

Turing Machines with a Stay-Option

The tape head can be stationary:

$$(Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

$$\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R, S\}$$

ex) $\delta(q_0, 0) = (q_1, 1, S)$



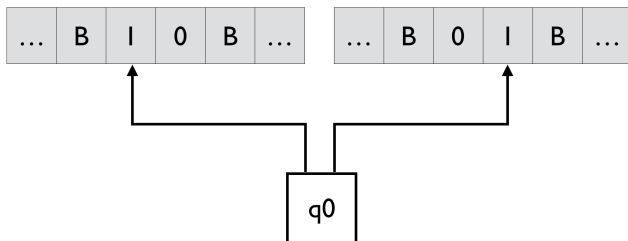
Equivalence

- ① Is the TM with a stay-option is as powerful as the standard TM?
- ② Is the standard TM is as powerful as the TM with a stay-option?

Multitape Turing Machines

Turing machine with

- multiple tapes
- each tape has its own tape head



Multitape Turing Machines

$$(Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

$$\delta : Q \times \Gamma^n \rightarrow Q \times \Gamma^n \times \{L, R\}^n$$

Initially,

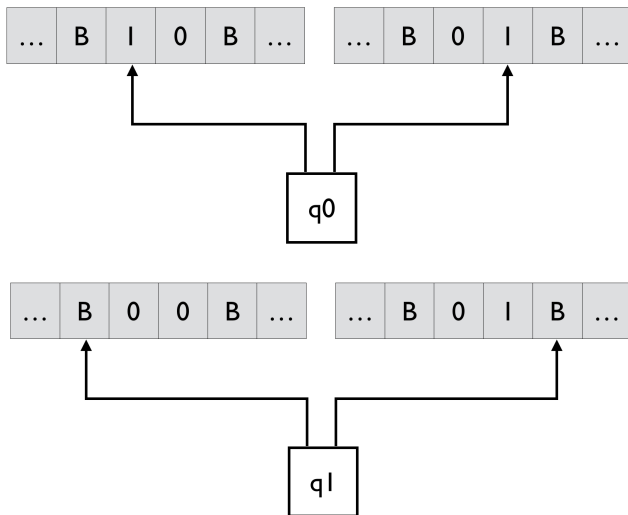
- 1 The input is placed on the first tape.
- 2 All other cells of all the tapes hold the blank.
- 3 The finite control is in the initial state.
- 4 The head of the first tape is at the left end of the input.
- 5 All other tape heads are at some arbitrary cell.

In one move, the multitape TM does the following:

- 1 The control enters a new state.
- 2 On each tape, a new tape symbol is written on the cell scanned.
- 3 Each of the tape heads makes a move independently of each other.

Multitape Turing Machines

ex) $\delta(q_0, 1, 1) = (q_1, 0, 1, L, R)$

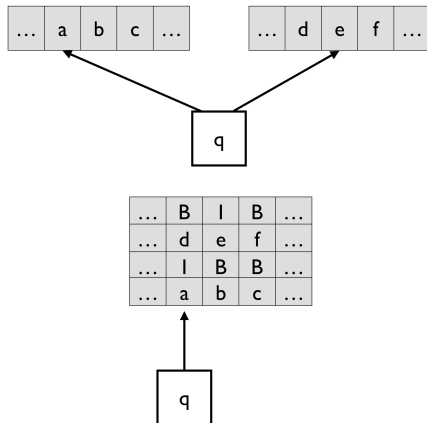


Equivalence

To represent a MTM by a standard TM, we need to represent

- the contents of multiple tapes, and
- the positions of multiple tape heads.

Represent them by a tape with multiple tracks: e.g.,



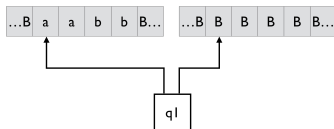
cf) Efficiency

Although the expressiveness is the same, MTM can be more efficient than the standard TM.

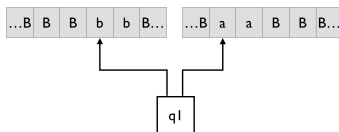
Example

Design a multitape Turing machine that accepts $L = \{a^n b^n \mid n \geq 1\}$.

- In standard TM, repeated back-and-forth movements are required.
- In MTM, copy all a 's to tape 2



and then match b 's on tape 1 against a 's on tape 2



Non-deterministic Turing Machines

$$(Q, \Sigma, \Gamma, \delta, q_0, B, F)$$
$$\delta : Q \times \Gamma \rightarrow 2^{Q \times \Gamma \times \{L, R\}}$$

- E.g., $\delta(q_0, a) = \{(q_1, b, R), (q_2, c, L)\}$
- A NTM accepts w if there is a sequence s.t.

$$q_0 w \vdash^* x_1 q_f x_2$$

with $q_f \in F$.

- Still, equivalent.

Summary

No matter how we extend the standard Turing machines, the expressiveness remains the same.

