

# Computability

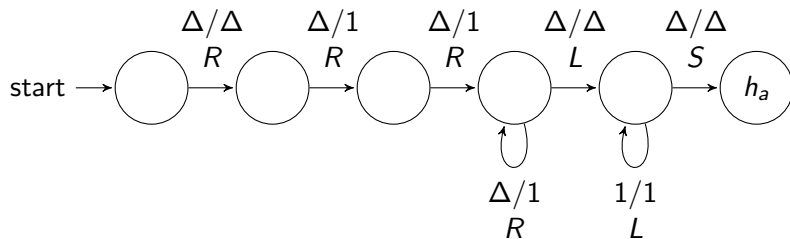
Rasmus Guldberg Pedersen

January 2015

# Non-deterministic Turing Machines as language acceptors

Give a definition of a non-deterministic Turing Machine. Describe how a non-deterministic Turing Machine processes its input and define the language accepted by a non-deterministic Turing machine.

# Nondeterministic Turing Machine (informal)



# Nondeterministic Turing Machine (formal)

$$T = (Q, \Sigma, \Gamma, q_0, \delta)$$

$Q$ , a finite set of states

$\Sigma$ , the input alphabet ( $\Sigma \subseteq \Gamma$ )

$\Gamma$ , the tape alphabet ( $\Delta \notin \Gamma$ )

$q_0$ , the initial state ( $q_0 \in Q$ )

$\delta$ , the transition function

$$\delta \subseteq Q \times (\Gamma \cup \{\Delta\}) \rightarrow (Q \cup \{h_a, h_r\}) \times (\Gamma \cup \{\Delta\} \times \{R, L, S\})$$

# Computational power

A NTM  $T_1$  that acts as input to another  $T_2$  can be thought of as executing  $T_2$  in parallel branches.

# Computational power

A NTM  $T_1$  that acts as input to another  $T_2$  can be thought of as executing  $T_2$  in parallel branches.

This behavior can be simulated by a deterministic TM executing in a breadth-first manner.

# Language accepted by a TM

If  $x \in \Sigma$  then  $x$  is accepted by  $T$  if

$$q_0 \Delta x \vdash_T^* wh_a y$$

$L \subseteq \Sigma^*$  is accepted by  $T$  if  $L = L(T)$  where

$$L(T) = \{x \in \Sigma^* \mid x \text{ is accepted by } T\}$$

The End

The End