Lesson 10: Non-deterministic Turing machines

Theme: Non-deterministic Turing machines.

A non-deterministic Turing machine (NTM) \( M = \langle \Sigma, \Gamma, Q, q_0, q_{\text{acc}}, q_{\text{rej}}, \delta \rangle \) is defined as the standard Turing machine, with the exception that \( \delta \) is now a relation:

\[
\delta \subseteq (Q - \{ q_{\text{acc}}, q_{\text{rej}} \}) \times \Gamma \times Q \times \Gamma \times \{ \text{Left, Right, Stay} \}
\]

As before, we write an element of \( \delta \) is in the form:

\[
(q, a) \rightarrow (p, b, \alpha).
\]

The initial configuration of \( M \) on input word \( w \) is \( q_0w \). For two configurations \( C, C' \), the notion of “one step computation” \( C \vdash C' \) is defined similarly as in the standard Turing machine. A run of \( M \) on input \( w \) is a sequence:

\[
C_0 \vdash C_1 \vdash \cdots
\]

where \( C_0 \) is the initial configuration on \( w \). A run is accepting/rejecting, if it ends up in an accepting/rejecting configuration, respectively. However, due to non-determinism, for each \( C \) there can be a few configurations \( C' \) such that \( C \vdash C' \), thus, there can be many runs. Some are accepting, some are rejecting, and some other do not halt.

Important definitions:

- An NTM \( M \) accepts \( w \), if there is an accepting run of \( M \) on \( w \).
- An NTM \( M \) rejects \( w \), if all runs of \( M \) on \( w \) are rejecting.
- A language \( L \) is decided by an NTM \( M \), if
  - for every \( w \in L \), \( M \) accepts \( w \);
  - for every \( w \notin L \), \( M \) rejects \( w \).
- A language \( L \) is recognized by an NTM \( M \), if
  - for every \( w \in L \), \( M \) accepts \( w \);
  - for every \( w \notin L \), \( M \) does not accept \( w \).

Recall that the standard TM is always deterministic. To avoid potential confusion, we will use the abbreviation DTM to mean deterministic Turing machine.

Theorem 10.1 For every language \( L \), the following holds.

- If \( L \) is recognized by an NTM \( M \), then there is a DTM \( M' \) that recognizes \( L \).
- If \( L \) is decided by an NTM \( M \), then there is a DTM \( M' \) that decides \( L \).
Appendix

A An informal definition of non-deterministic algorithm

One can view a “non-deterministic” algorithm as an algorithm as defined in the appendix in Lesson 7, with an additional special variable $z$ and an instruction of the following form:

$$z := 0 \parallel 1;$$

(1)

This instruction means “randomly assign variable $z$ with either 0 or 1.”

A non-deterministic algorithm $A$ “accepts” an input word $w$, if on every instruction of the form $[z]$, variable $z$ can be assigned with 0 or 1 such that $A$ will “return true.” Note that the instruction $[z]$ can be encountered more than once during the execution of algorithm $A$. For example, it may appear inside a while-loop.