IV. Turing Machine

Yuxi Fu

BASICS, Shanghai Jiao Tong University

Alan Turing

Alan Turing (23Jun.1912-7Jun.1954), an English student of Church, introduced a machine model for effective calculation in

"On Computable Numbers, with an Application to the Entsheidungsproblem",

Proc. of the London Mathematical Society, 42:230-265, 1936.

Turing Machine, Halting Problem, Turing Test

Motivation

What are necessary for a machine to calculate a function?

- The machine should be able to interpret numbers;
- The machine must be able to operate and manipulate numbers according to a set of predefined instructions;

and

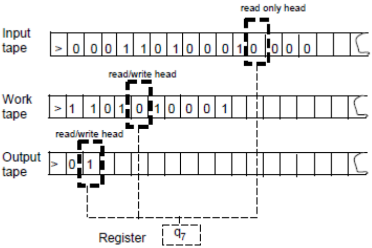
- The input number has to be stored in an accessible place;
- There should be an accessible place for the machine to store intermediate results;
- The output number has to be put in an accessible place.

Turing Machine

A k-tape Turing Machine $\mathbb{M}$ has k-tapes such that

- The first tape is the read-only input tape.
- The other $k-1$ tapes are the read/write work tapes.
- The k-th tape is also used as the output tape.

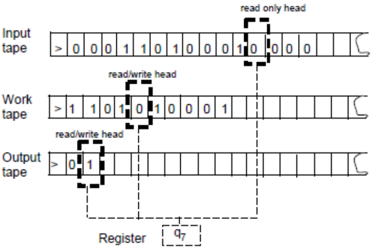
Every tape comes with a read/write head.



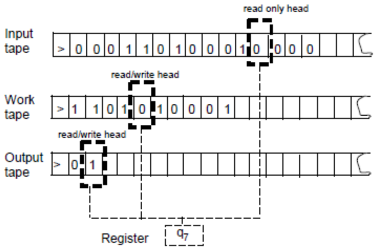
Turing Machine

The machine is described by a tuple (Γ, Q, δ) containing

- A finite set Γ, called alphabet, of symbols. It contains a blank symbol $\square$, a start symbol $\triangleright$, and the digits 0 and 1.
- A finite set Q of states. It contains a start state $q_{\mathtt{start}}$ and a halting state $q_{\mathtt{halt}}$.
- A transition function $\delta : Q \times \Gamma^k \to Q \times \Gamma^{k-1} \times \{\mathtt{L}, \mathtt{S}, \mathtt{R}\}^k$, describing the rules of each computation step.



Computation and Configuration



Computation, configuration, initial/final configuration

A TM for the Palindrome Problem

$Q = \{q_s, q_h, q_c, q_l, q_t\}$; $\Gamma = \{\Box, \rhd, 0, 1\}$; two work tapes.

$\langle q_s, \rhd, \rhd, \rhd \rangle \rightarrow \langle q_c, \rhd, \rhd, \mathtt{R}, \mathtt{R}, \mathtt{R} \rangle$

$\langle q_c, 0, \Box, \Box \rangle \rightarrow \langle q_c, 0, \Box, \mathtt{R}, \mathtt{R}, \mathtt{S} \rangle$
$\langle q_c, 1, \Box, \Box \rangle \rightarrow \langle q_c, 1, \Box, \mathtt{R}, \mathtt{R}, \mathtt{S} \rangle$
$\langle q_c, \Box, \Box, \Box \rangle \rightarrow \langle q_l, \Box, \Box, \mathtt{L}, \mathtt{S}, \mathtt{S} \rangle$

$\langle q_l, 0, \Box, \Box \rangle \rightarrow \langle q_l, \Box, \Box, \mathtt{L}, \mathtt{S}, \mathtt{S} \rangle$
$\langle q_l, 1, \Box, \Box \rangle \rightarrow \langle q_l, \Box, \Box, \mathtt{L}, \mathtt{S}, \mathtt{S} \rangle$
$\langle q_l, \rhd, \Box, \Box \rangle \rightarrow \langle q_t, \Box, \Box, \mathtt{R}, \mathtt{L}, \mathtt{S} \rangle$

$\langle q_t, \Box, \rhd, \Box \rangle \rightarrow \langle q_h, \rhd, 1, \mathtt{S}, \mathtt{S}, \mathtt{S} \rangle$
$\langle q_t, 0, 1, \Box \rangle \rightarrow \langle q_h, 1, 0, \mathtt{S}, \mathtt{S}, \mathtt{S} \rangle$
$\langle q_t, 1, 0, \Box \rangle \rightarrow \langle q_h, 0, 0, \mathtt{S}, \mathtt{S}, \mathtt{S} \rangle$
$\langle q_t, 0, 0, \Box \rangle \rightarrow \langle q_t, 0, \Box, \mathtt{R}, \mathtt{L}, \mathtt{S} \rangle$
$\langle q_t, 1, 1, \Box \rangle \rightarrow \langle q_t, 1, \Box, \mathtt{R}, \mathtt{L}, \mathtt{S} \rangle$

$\{0, 1, \square, \triangleright\}$ vs. Larger Alphabets

Suppose $\mathbb{M}$ has k tapes with the alphabet Γ.

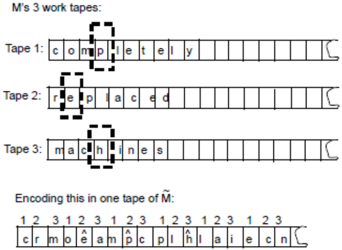
A symbol of $\mathbb{M}$ is encoded by a string $\sigma \in \{0, 1\}^*$ of length $\log |\Gamma|$.

States: A state q is turned into states q, $\langle q, \sigma_1^1, \ldots, \sigma_1^k \rangle$ where $|\sigma_1^1| = \ldots = |\sigma_1^k| = 1$, $\ldots$, $\langle q, \sigma_{\log |\Gamma|}^1, \ldots, \sigma_{\log |\Gamma|}^k \rangle$ where $|\sigma_{\log |\Gamma|}^1| = \ldots = |\sigma_{\log |\Gamma|}^k| = \log |\Gamma|$.

A computation step of $\mathbb{M}$ is simulated in $\widetilde{\mathbb{M}}$ by $\log |\Gamma|$ steps to read, $\log |\Gamma|$ steps to write, and $\log |\Gamma|$ steps to relocate the heads.

One Tape vs. Many Tapes

The basic idea is to interleave k tapes into one tape.
The first $n+1$ cells are reserved for the input.



Every symbol a of $\mathbb{M}$ is turned into two symbols $a, \widehat{a}$ in $\widetilde{\mathbb{M}}$, with $\widehat{a}$ used to indicate head position.

One Tape vs. Many Tapes

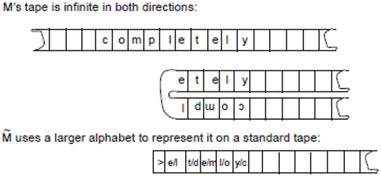
The machine $\widetilde{\mathbb{M}}$ copies the input bits to the first imaginary tape. The head then moves left to the $(n+2)$-th cell.

Sweeping the tape cells from left to right. Record in the register the k symbols marked with the hat $\widehat{\_}$.

Sweeping the tape cells from right to left to update using the transitions of $\mathbb{M}$.

One Unidirectional Tape vs. Bidirectional Tape

The idea is that $\widetilde{\mathbb{M}}$ makes use of the alphabet $\Gamma \times \Gamma$.



Every state q of $\mathbb{M}$ is turned into $\overline{q}$ and $\underline{q}$.

Turing Machine Model is extremely robust.

Function Definable by Turing Machine

A k-ary function $f : \underbrace{\{0,1\}^* \times \ldots \times \{0,1\}^*}_{k} \to \{0,1\}^*$ is

Turing definable if there is a TM $\mathbb{M}$ such that whenever k binary numbers $n_1, \ldots, n_k$ are written on the input tape, the following statements are valid:

- If $f(n_1, \ldots, n_k)$ is defined, then $\mathbb{M}$ terminates in a configuration with $f(n_1, \ldots, n_k)$ written on the output tape.
- If $f(n_1, \ldots, n_k)$ is undefined, then the execution of $\mathbb{M}$ is infinite.

Lambda Definability vs. Turing Definability

Theorem. All λ-definable functions are Turing definable.

A. Turing, A.

Computability and λ-definability.

Journal of Symbolic Logic, **2**:153-163, 1937.

Lambda Definability vs. Turing Definability

The theorem can be proved by designing a Turing Machine for each closed λ-term.

- The bound variables are treated using de Bruijn notation.
- The machine uses the leftmost reduction strategy.

A uniform approach that transforms a closed λ-term to a Turing Machine is left as an exercise.

Gödel was reported as saying that Turing machines offer the most convincing formalization of mechanical procedures.

Turing remarked that the λ-calculus is more convenient.

Church pointed out that Turing Machines have the advantage of making the identification with effectiveness in the ordinary sense evident immediately.

Exercise. Describe an algorithm that transforms a closed λ-term to a Turing Machine. The machine should meet the specification: Whenever it starts with a closed λ-term written on its input tape, it carries out the leftmost reduction, and stops if the reduction terminates.