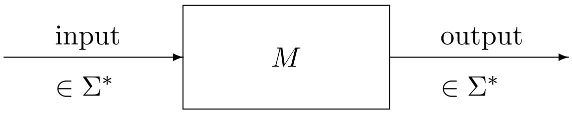
6.5 Turing Machines

The mathematical results of complexity theory consist almost exclusively of asymptotic cost estimates, and in almost all cases these estimates are upper bounds. Complexity theory in its various flavours relies on diverse models of computation. In this section we shortly sketch the common formalism by Turing machines.



Here Σ (as usual) denotes a finite alphabet. The input is a finite string on a tape that is infinite in both directions. The Turing machine M can assume states from a finite set that also contains a state "halt". Depending on the state the machine executes certain operations, for instance reads one character from the tape, changes its state, writes one character to the tape, moves the reading head by one position to the left or to the right. If M reaches the state "halt", then the current string on the tape is the output.

Let $L \subseteq \Sigma^*$ be a language. If M reaches the "halt" state after a finite number of steps for all inputs $x \in L$, then we say that M **accepts the language** L. If $f\colon L \longrightarrow \Sigma^*$ is a function, and M reaches "halt" after finitely many steps for each $x \in L$ with output $f(x)$, then we say that M **computes** f.

With some effort, and not too overwhelming elegance, we can describe all algorithms by Turing machines. Then by counting the steps we may express their complexities in the form: for input x the machine M takes τ_x steps until reaching "halt".

Usually we consider "worst case" complexity. Let $L_n := L \cap \Sigma^n$. Then the function

$$t_M\colon \mathbb{N} \longrightarrow \mathbb{N}, \quad t_M(n) := \max\{\tau_x \mid x \in L_n\},$$

is called **(time) complexity** of the Turing machine M (for L).

The subset $\mathbf{P}$ ("polynomial time") of the set of all functions from L to Σ^* consists of the functions $f\colon L \longrightarrow \Sigma^*$ for which there exists a Turing machine M and an integer $k \in \mathbb{N}$ such that

(i) M computes f,

(ii) $t_M(n) \leq n^k$ for almost all $n \in \mathbb{N}$.

Remark Equivalent with (ii) is the statemant: There is a polynomial $p \in \mathbb{N}[X]$ with $t_M(n) \leq p(n)$ for all $n \in \mathbb{N}$.

For if there is such a polynomial $p = a_r X^r + \cdots + a_0$ (with $a_r \neq 0$), then

$$\begin{array}{rcl} a_r n^r & \geq & a_{r-1} n^{r-1} + \cdots + a_0 \quad \text{for } n \geq n_0, \\ p(n) & \leq & 2a_r n^r \quad \text{for } n \geq n_0, \\ p(n) & \leq & n^{r+1} \quad \text{for } n \geq n_1 = \max\{2a_r, n_0\}. \end{array}$$

Conversely if $t_M(n) \leq n^k$ for $n \geq n_0$, then we choose $c \in \mathbb{N}$ with $t_M(n) \leq c$ for the finitely many $n = 0, \ldots, n_0 - 1$. Then $t_M(n) \leq p(n)$ for all $n \in \mathbb{N}$ with $p = X^k + c$.

Analogously we define the set **EXPTIME** ("exponential time"): f is in **EXPTIME** if there exist a TURING machine M, an integer $k \in \mathbb{N}$, and real numbers $a, b \in \mathbb{R}$ with

(i) M computes f,

(ii) $t_M(n) \leq a \cdot 2^{bn^k}$ for almost all $n \in \mathbb{N}$.

Obviously $\mathbf{P} \subseteq$ **EXPTIME**.

Examples with $\Sigma = \mathbb{F}_2$.

1. Assume

$$L := \{(p, z) \in \mathbb{N}^2 \mid p \text{ prime } \equiv 3 \pmod 4,\ z \in \mathbb{M}_p^2\}$$

is coded as a subset of Σ^* by a suitable binary representation. Let $f(p, z) =$ the square root of $z \bmod p$, likewise coded as an element of Σ^*. Then $f \in \mathbf{P}$ by 5.3.

2. Let $L = \mathbb{N}_2$ be the set of integers ≥ 2 (binary coded). Let $f(x) =$ be the smallest prime factor of x. Then $f \in$ **EXPTIME** since we can try all the integers $\leq \sqrt{x} \leq 2^{n/2}$.

Presumably $f \notin \mathbf{P}$.

3. The **knapsack problem**. Here

$$L = \{(m, a_1, \ldots, a_m, N) \mid m, a_1, \ldots, a_m, N \in \mathbb{N}\}$$

with suitable binary encoding,

$$f(m, a_1, \ldots, a_m, N) = \begin{cases} 1, & \text{if there is } S \subseteq \{1, \ldots, m\} \\ & \text{with } \sum_{i \in S} a_i = N, \\ 0 & \text{otherwise.} \end{cases}$$

Then $f \in$ **EXPTIME** since we can try all of the 2^m subsets $S \subseteq \{1, \ldots, m\}$.

Presumably $f \notin \mathbf{P}$.