4 Turing-Maschinen

Frage: Was kann von einem rechnenden Menschen (ggf. mit Hilfsmitteln) berechnet werden?

In der Arbeit „*On computable numbers and an application to the Entscheidungsproblem*" von 1936 geht Alan Turing (1912-1954) davon aus, daß ein „Rechnender"

- sich Notizen auf beliebig viel Papier,
- sich endlich viel merken und
- nur einen begrenzten Teil der Notizen gleichzeitig überblicken und sich nur blätternd durch die Notizen bewegen kann.
- Außerdem arbeitet er mechanisch.

Aus diesen Überlegungen heraus definierte er die heute so genannte „Turing-Maschine".

Definition. 1. Eine *nichtdeterministische Turing-Maschine* (NTM) ist ein Tupel $M = (Q, \Sigma, \Gamma, \iota, \delta, \Box, F)$, wobei

- Q eine endliche Menge von „Zuständen“,
- Σ das „Eingabealphabet“,
- Γ das „Arbeitsalphabet“ mit $\Sigma \cup \{\Box\} \subseteq \Gamma$,
- $\iota \in Q$ der „Initialzustand“,
- $\delta\colon Q \times \Gamma \to \mathcal{P}(Q \times \Gamma \times \{-1, 0, 1\})$ die “Überführungsfunktion“,
- $\Box \in \Gamma \setminus \Sigma$ das „Leerzeichen“ und
- $F \subseteq Q$ die Menge der „Endzustände“ sind.

2. Eine *(deterministische) Turing-Maschine* (TM) ist eine NTM mit $|\delta(q, a)| \leq 1$ für alle $q \in Q$ und $a \in \Gamma$.

3. Eine *Konfiguration* der NTM M ist ein Tripel (q, p, B) mit $q \in Q$, $p \in \mathbb{Z}$ und $B \colon \mathbb{Z} \to \Gamma$ mit $B(i) = \square$ für fast alle $i \in \mathbb{Z}$.

4. Für zwei Konfigurationen (q, p, B) und (q', p', B') setzen wir

$$(q, p, B) \vdash (q', p', B')\,,$$

wenn es $(q', b, x) \in \delta(q, B(p))$ gibt mit $p' = p + x$ und

$$B'(i) = \begin{cases} b & \text{falls } i = p \\ B(i) & \text{sonst.} \end{cases}$$

5. Für Konfigurationen K und K' schreiben wir

$$K \vdash^* K'\,,$$

wenn es $n \in \mathbb{N}$ und Konfigurationen $K_0, K_1, \cdots, K_n$ gibt mit

$$K = K_0 \vdash K_1 \vdash \cdots \vdash K_n = K'\,.$$

6. Die NTM M *akzeptiert* das Wort $w = a_0 a_1 a_2 \cdots a_{n-1} \in \Sigma^*$, wenn es eine Konfiguration (f, p, B) mit $f \in F$ gibt, so daß

$$\delta(f, B(p)) = \emptyset \text{ und } (\iota, 0, B_w) \vdash^* (f, p, B)\,,$$

wobei

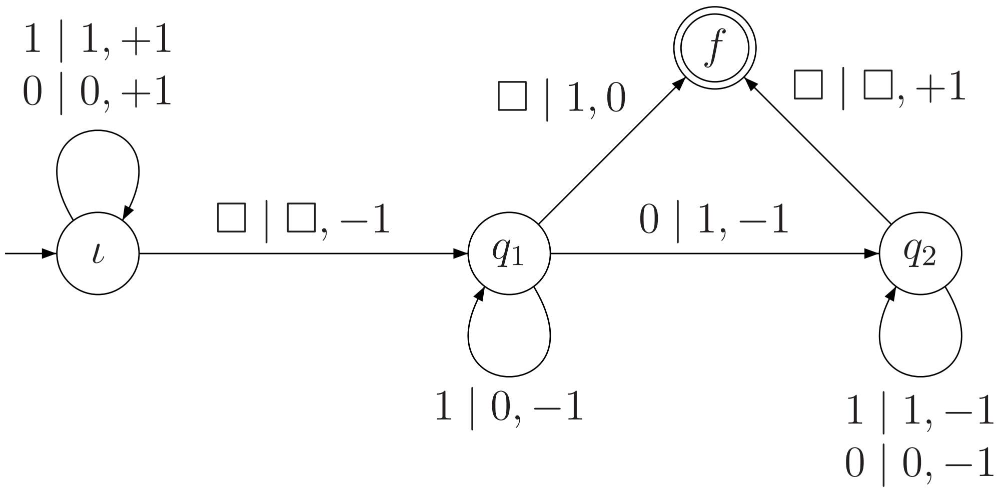
$$B_w(i) = \begin{cases} a_i & \text{falls } 0 \leq i < n \\ \square & \text{sonst} \end{cases}$$

für alle $i \in \mathbb{Z}$ gilt. Sei $L(M) \subseteq \Sigma^*$ die Menge der von M akzeptierten Wörter.

Beispiel. TM, die alle Wörter über $\{0, 1\}$ akzeptiert und dabei die Eingabe (als Binärzahl) um 1 erhöht:

- $Q = \{\iota, q_0, q_1, f\}$
- $\Gamma = \{0, 1, \Box\}$
- $\Sigma = \{0, 1\}$
- $F = \{f\}$

Die Überführungsfunktion δ ist durch das folgenden Bild gegeben:

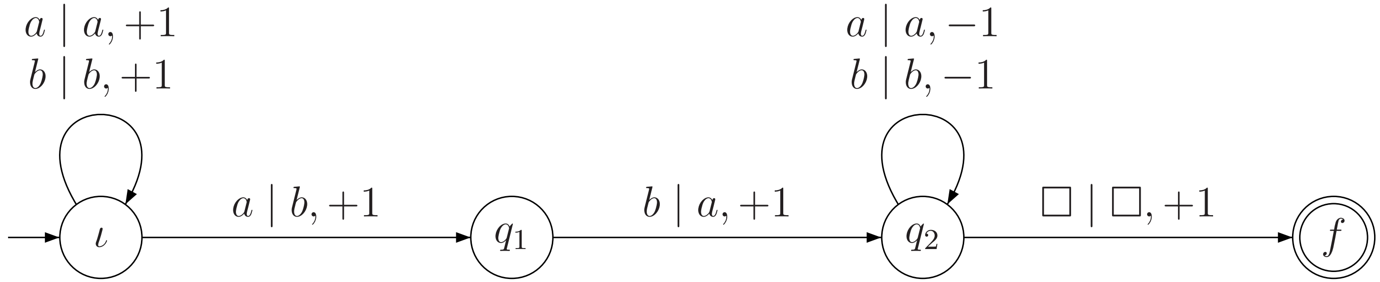


z.B. $\delta(\iota, 1) = \{(\iota, 1, +1)\}$, d.h. bei Lesen von 1 im Zustand ι bleibt die TM im Zustand ι, läßt die 1 auf dem Band stehen und bewegt den Kopf eine Stelle nach rechts.

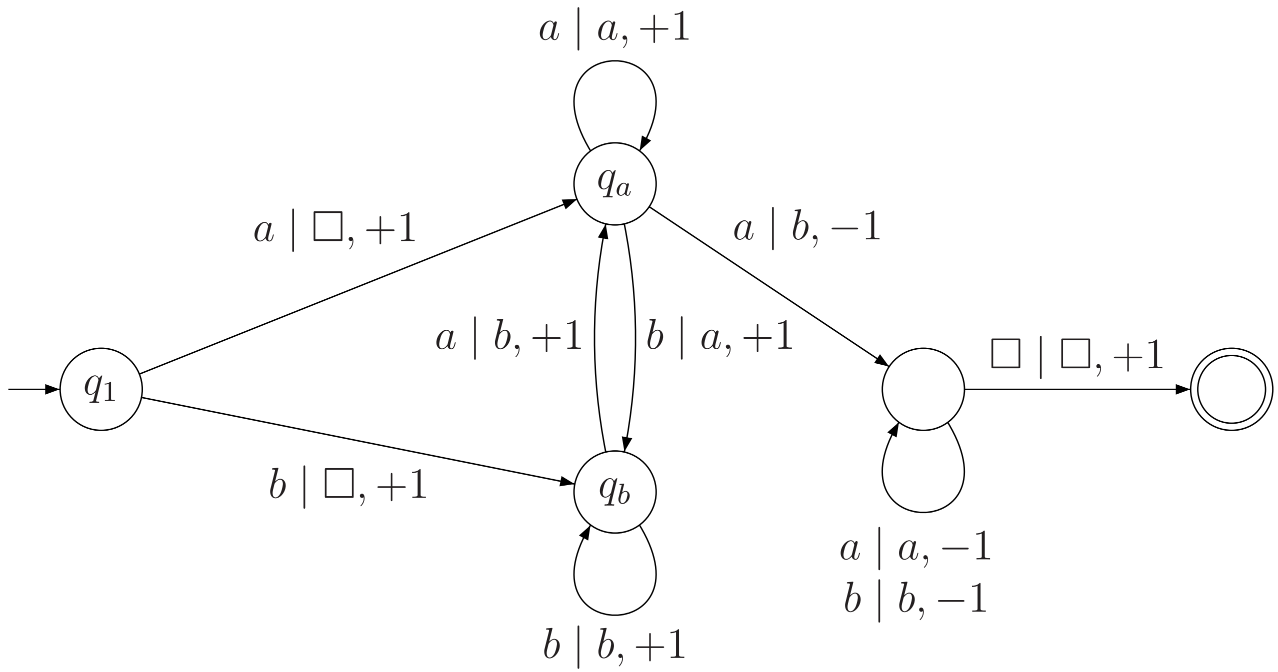
<u>Beispielrechnung:</u>

$$
\begin{array}{llllll}
 & \cdots\Box\Box\,\overset{\iota}{1}\,0011\Box\Box\cdots & \vdash & \cdots\Box\Box 1\,\overset{\iota}{0}\,011\Box\Box\cdots & \vdash^* & \cdots\Box\Box 10011\,\overset{\iota}{\Box}\,\Box\cdots \\
\vdash & \cdots\Box\Box 1001\,\overset{q_1}{1}\,\Box\Box\cdots & \vdash & \cdots\Box\Box 100\,\overset{q_1}{1}\,0\Box\Box\cdots & \vdash & \cdots\Box\Box 10\,\overset{q_1}{0}\,00\Box\Box\cdots \\
\vdash & \cdots\Box\Box 1\,\overset{q_2}{0}\,100\Box\Box\cdots & \vdash & \cdots\Box\Box\,\overset{q_2}{1}\,0100\Box\Box\cdots & \vdash & \cdots\Box\,\overset{q_2}{\Box}\,10100\Box\Box\cdots \\
\vdash & \cdots\Box\Box\,\overset{f}{1}\,0100\Box\Box\cdots & & & &
\end{array}
$$

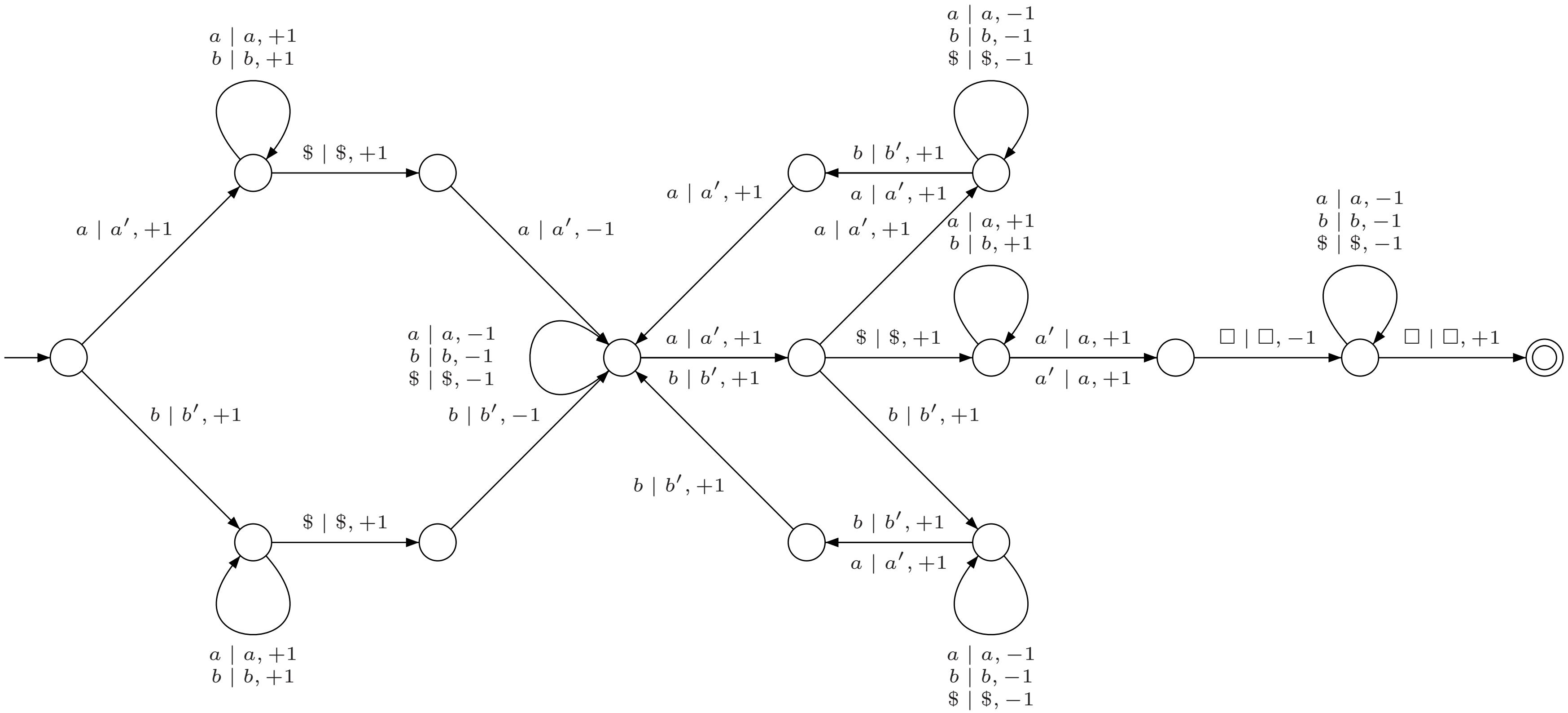
Beispiel. NTM mit $L(M) = \{a, b\}^*\{ab\}\{a, b\}^*$, die beim Akzeptieren ein Vorkommen von ab durch ba ersetzt:



Beispiel. NTM mit $L(M) = \{a, b\}^*\{aa\}\{a, b\}^*$, die beim Akzeptieren ein Vorkommen von aa durch b ersetzt:



Beispiel. Die folgende TM akzeptiert die Menge der Wörter $w\$w$ mit $w \in \{a, b\}^*$. Wir haben gesehen, daß diese Sprache nicht kontextfrei ist.



Definition. Eine Sprache $L \subseteq \Sigma^*$ heißt *semi-entscheidbar*, wenn es eine det. TM M gibt mit $L = L(M)$. Sie heißt *entscheidbar*, wenn M zusätzlich so gewählt werden kann, daß sie bei jeder Eingabe irgendwann hält (d.h. für jedes Wort $w \in \Sigma^*$ existiert eine Konfiguration (q, p, B) mit $(\iota, 0, B_w) \vdash^* (q, p, B)$ und $\delta(q, B(p)) = \emptyset$).

Bemerkung. Es ist leicht zu sehen, daß jede reguläre Sprache entscheidbar ist (betrachte den DFA als TM, die niemals den Inhalt des Bandes ändert).

Etwas weniger einfach ist zu sehen, daß jede kontextfreie Sprache entscheidbar ist (verwende einen zusätzlichen Abschnitt des Bandes, um den CYK-Algorithmus auszuführen).

Man kann auch zeigen, daß jede NTM M in eine äquivalente TM umgewandelt werden kann (die eine Breitensuche in der Menge der Konfigurationen von M durchführt und daher „sehr viel langsamer“ als M ist).

Church-Turing-These *Eine Sprache $L \subseteq \Sigma^*$ kann genau dann durch einen Algorithmus akzeptiert werden, wenn es eine Turing-Maschine M gibt mit $L(M) = L$.*

mögliches Killerargument gegen die These:

- ein Algorithmus, der einen Sprache akzeptiert, die von keiner Turing-Maschine akzeptiert wird.

allg. akzeptierte Argumente für die These:

- Turings Analyse des Rechnens eines Menschen.
- Man kann von vielen Sprachen zeigen, daß sie von einer TM akzeptiert werden.
- Verschiedenste alternative Berechnungsmodelle (z.B. Registermaschinen, Markov-Algorithmen und -Tabellen, Gödels rekursive Funktionen, Churchs λ-Kalkül, rudimentäre Programmiersprachen, ...) haben sich als nicht stärker als Turing-Maschinen erwiesen.
- Niemand hat in den vergangenen 80 Jahren eine algorithmisch erkennbare Sprache angegeben, die von keiner Turingmaschine akzeptiert wird.