Berechenbarkeit

Script, Kapitel 2

- Intuitiver Berechenbarkeitsbegriff
- Turing-Berechenbarkeit
- WHILE-Berechenbarkeit
- Church'sche These
- Entscheidungsprobleme
- Unentscheidbarkeit des Halteproblems für Turingmaschinen

Intuitiver Algorithmusbegriff

Ein Algorithmus

- überführt Eingabedaten in Ausgabedaten (wobei die Art der Daten vom Problem, das durch den Algorithmus gelöst werden soll, abhängig ist),
- besteht aus einer endlichen Folge von Anweisungen mit folgenden Eigenschaften:
  - es gibt eine eindeutig festgelegte Anweisung, die als erste auszuführen ist,
  - nach Abarbeitung einer Anweisung gibt es eine eindeutig festgelegte Anweisung, die als nächste abzuarbeiten ist, oder die Abarbeitung des Algorithmus ist beendet und hat eindeutig bestimmte Ausgabedaten geliefert,
  - die Abarbeitung einer Anweisung erfordert keine Intelligenz (ist also prinzipiell durch eine Maschine realisierbar).

Beispiele für Algorithmen

- der Gaussche[1] Algorithmus zur Lösung von linearen Gleichungssystemen (über den rationalen Zahlen),
- Kochrezepte (mit Zutaten und Kochgeräten als Eingabe und dem fertigen Gericht als Ausgabe),
- Bedienungsanweisungen für Geräte,
- PASCAL-Programme.

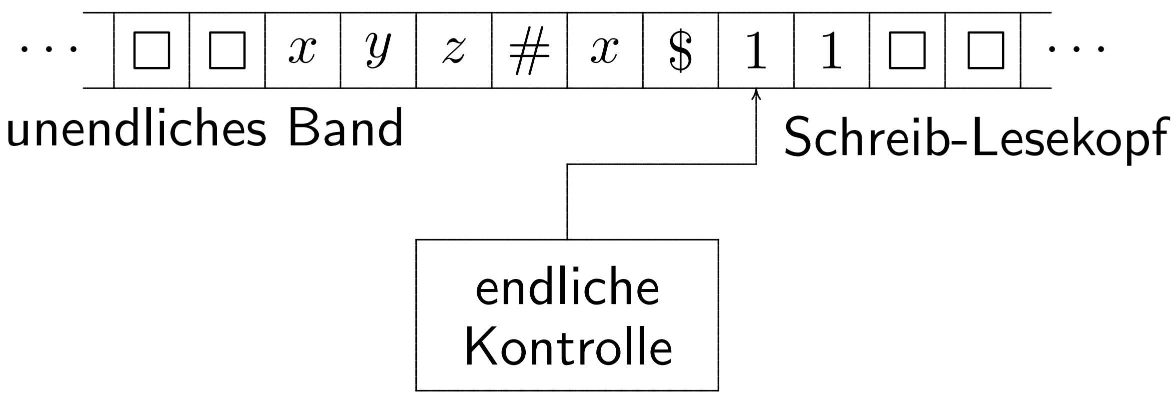
[1] Carl Friedrich Gauss, 1777–1855, deutscher Mathematiker.

Idee der Turingmaschine[2]

- Arbeitsband
  - beidseitig (potenziell) unendliches Band, in Felder eingeteilt;
  - jedes Feld enthält einen Buchstaben aus dem Bandalphabet;
  - zum Bandalphabet gehört das Leerzeichen oder Blank-Symbol $\Box$.
- Schreib-Lesekopf
  - über dem Band beweglich
  - bearbeitet ein aktuelles Feld
- endliche Steuerung oder Kontrolle
  - endliche Menge von internen Zuständen, die den Programmablauf regeln

[2] Alan Mathison Turing (1912–1954), britischer Mathematiker, Logiker, Kryptoanalytiker und Computerkonstrukteur.

Veranschaulichung einer Turingmaschine



Definition der Turingmaschine

Eine (deterministische) Turingmaschine (kurz: TM) ist gegeben durch ein 7-Tupel $M = (Z, \Sigma, \Gamma, \delta, z_0, \Box, E)$. Hierbei sind

- Z eine endliche Menge (Zustandsmenge),
- Σ ein Alphabet (Eingabealphabet),
- Γ ein Alphabet (Bandalphabet) mit $\Sigma \subseteq \Gamma$,
- $\delta\colon (Z \setminus E) \times \Gamma \to Z \times \Gamma \times \{L, R, N\}$ eine Funktion (Überführungsfunktion),
- $z_0 \in Z$ (Anfangszustand),
- $\Box \in \Gamma \setminus \Sigma$ (Leerzeichen, Blank),
- $E \subseteq Z$ (Menge der Endzustände).

Erstes Beispiel einer Turingmaschine

Gegeben sei die Turingmaschine

$$M = (\{z_0, z_1, z_2, z_e\}, \{0, 1\}, \{0, 1, \Box\}, \delta, z_0, \Box, \{z_e\}),$$

wobei δ wie folgt definiert ist. Wir geben dabei δ in einer Tabelle an, wobei im Kreuzungspunkt der Zeile mit der Bezeichnung a und der Spalte mit der Bezeichnung z der Funktionswert $\delta(z, a)$ steht.

δ	z_0	z_1	z_2
$\Box$	$(z_1, \Box, L)$	$(z_e, 1, N)$	$(z_e, \Box, R)$
0	$(z_0, 0, R)$	$(z_2, 1, L)$	$(z_2, 0, L)$
1	$(z_0, 1, R)$	$(z_1, 0, L)$	$(z_2, 1, L)$

Konfiguration einer Turingmaschine

Eine Konfiguration k einer Turingmaschine $M = (Z, \Sigma, \Gamma, \delta, z_0, \Box, E)$ ist ein Wort $k \in \Gamma^* Z \Gamma^*$. Dabei soll $k = \alpha z \beta$ folgendermaßen interpretiert werden:

- $\alpha\beta$ steht auf dem Eingabeband, des weiteren stehen nur noch Blankzeichen $\Box$ auf dem Band,
- die Turingmaschine befindet sich im Zustand z und
- der Kopf der Turingmaschine steht über dem ersten Symbol von β.

Eine Startkonfiguration ist $k_0 = z_0 w$ mit $w \in \Sigma^*$.

Eine Endkonfiguration ist $k_e = \Box^m z_e w' \Box^n$ mit $z_e \in E$, $w' \in \Sigma^*$, $m, n \in \mathbb{N}$.

Die binäre Relation $\vdash$ in der Menge der Konfigurationen

(Änderung der Konfiguration in einem Schritt der TM)

$$
\begin{aligned}
& a_1 \dots a_m z b_1 \dots b_n \\
\vdash & \begin{cases}
a_1 \dots a_m z' c b_2 \dots b_n & \text{falls } \delta(z, b_1) = (z', c, N),\ m \geq 0,\ n \geq 1, \\
a_1 \dots a_m c z' b_2 \dots b_n & \text{falls } \delta(z, b_1) = (z', c, R),\ m \geq 0,\ n \geq 2, \\
a_1 \dots a_{m-1} z' a_m c b_2 \dots b_n & \text{falls } \delta(z, b_1) = (z', c, L),\ m \geq 1,\ n \geq 1,
\end{cases}
\end{aligned}
$$

$$a_1 \dots a_m z b_1 \vdash a_1 \dots a_m c z' \Box \text{ falls } \delta(z, b_1) = (z', c, R),\ m \geq 0,$$

$$z b_1 \dots b_n \vdash z' \Box c b_2 \dots b_n \text{ falls } \delta(z, b_1) = (z', c, L),\ n \geq 1.$$

Die binäre Relation $\vdash^*$

(Änderung der Konfiguration in endlich vielen Schritten der TM)

Mit $\vdash^*$ bezeichnen wir den reflexiven und transitiven Abschluss der binären Relation $\vdash$, also es gilt $k_0 \vdash^* k_e$ genau dann, wenn

(i) $k_0 = k_e$ ist, oder

(ii) eine Zahl $n \geq 0$ und Konfigurationen $k_1, k_2, \ldots, k_n$ existieren, so dass

$$k_0 \vdash k_1 \vdash k_2 \vdash \cdots \vdash k_n \vdash k_e.$$

Turingberechenbarkeit

Eine Funktion $f : \Sigma^* \to \Sigma^*$ heißt Turingberechenbar, falls es eine Turingmaschine $M = (Z, \Sigma, \Gamma, \delta, z_0, \Box, E)$ gibt, so dass für alle $x, y \in \Sigma^*$ gilt:

$$f(x) = y \quad \text{genau dann, wenn} \quad z_0 x \vdash^* \Box \ldots \Box z_e y \Box \ldots \Box \text{ für ein } z_e \in E.$$

Eine Funktion $f : \mathbb{N}^k \to \mathbb{N}$ für ein $k \in \mathbb{N}$ heißt Turingberechenbar, falls es eine Turingmaschine $M = (Z, \{0, 1, \#\}, \Gamma, \delta, z_0, \Box, E)$ gibt, so dass für alle $n_1, n_2, \ldots, n_k, m \in \mathbb{N}$ gilt:

$$f(n_1, n_2, \ldots, n_k) = m \quad \text{genau dann, wenn für ein } z_e \in E:$$
$$z_0 \mathrm{bin}(n_1) \# \mathrm{bin}(n_2) \# \ldots \# \mathrm{bin}(n_k) \vdash^* \Box \ldots \Box z_e \mathrm{bin}(m) \Box \ldots \Box$$

Beispiele für Turingberechenbare Funktionen

Die Nachfolgerfunktion

$$f: \mathbb{N} \to \mathbb{N} \text{ vermöge } n \mapsto f(n) = n + 1$$

ist Turingberechenbar, da die Turingmaschine aus unserem ersten Beispiel die Eingabe $\mathrm{bin}(n)$ in die Ausgabe $\mathrm{bin}(n+1)$ transformiert.

Die für alle Wörter aus $\{a, b\}^*$ nicht-definierte Funktion

$$\Omega: \{a, b\}^* \to \{a, b\}^* \text{ vermöge } w \mapsto \Omega(w) = \textit{ nicht definiert}$$

ist Turing-berechenbar, da sie von der Turingmaschine

$$M = (\{z_0, z_e\}, \{a, b\}, \{a, b, \Box\}, \delta, z_0, \Box, \{z_e\})$$

mit $\delta(z_0, x) = (z_0, x, N)$ für alle $x \in \{a, b, \Box\}$ berechnet wird.

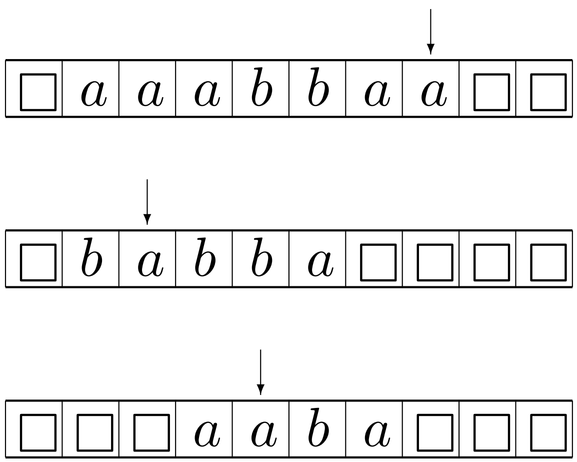
Mehrband-Turingmaschinen

- $k \geq 1$ Bänder (Eingabe/Ausgabe auf Band 1)
- Jedes Band hat einen eigenen Schreib-Lesekopf, der separat bewegt wird.
- Formal $\delta \colon Z \times \Gamma^k \to Z \times \Gamma^k \times \{L, R, N\}^k$

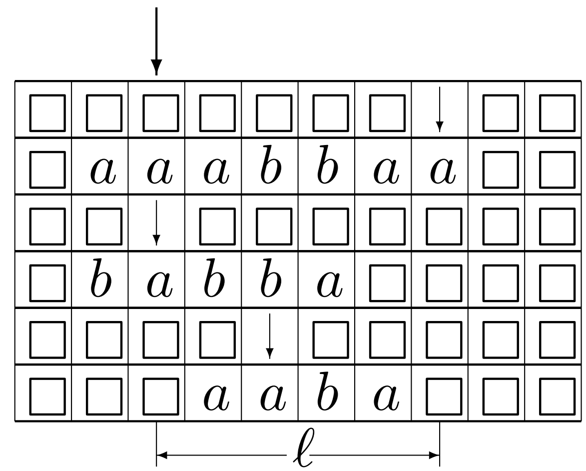
Äquivalenz von Mehrband- und (Einband)-TM

Zu jeder Mehrband-TM M gibt es eine (Einband-)TM M', die dieselbe Funktion berechnet wie M.

Mehrband-TM



Einband-TM



- k Bänder
- 1 Schritt

- 1 Band mit $2k$ Spuren
- $\approx 2(\ell + k)$ Schritte

Notationen

" $\dot{-}$ ": modifizierte Subtraktion

$\dot{-} : \mathbb{N}^2 \to \mathbb{N}$ vermöge $(n_1, n_2) \mapsto n_1 \dot{-} n_2 = \begin{cases} n_1 - n_2 & \text{falls } n_1 \geq n_2, \\ 0 & \text{sonst.} \end{cases}$

"Band $:=$ Band $+ 1$": TM, die zu einer Zahl 1 dazuaddiert

"Band(i) $:=$ Band$(i) + 1$": k-Band-TM, die auf i-tem Band 1 addiert und alle anderen Bänder unverändert läßt

"Band(i) $:=$ Band$(i) \dot{-} 1$": k-Band-TM, die auf i-tem Band 1 modifiziert subtrahiert und alle anderen Bänder unverändert läßt

"Band(i) $:=$ Band(j)": k-Band-TM, die Inhalt des j-ten Bandes auf i-tes Band kopiert

Nacheinanderausführung von Turingmaschinen

Seien M_1 und M_2 zwei Turingmaschinen, so wollen wir durch

$$\text{start} \longrightarrow M_1 \longrightarrow M_2 \longrightarrow \text{stop}$$

oder auch durch

$$M_1; M_2$$

diejenige Turingmaschine verstehen, die zuerst wie die Turingmaschine M_1 arbeitet und wenn M_1 einen Stopzustand erreichen würde in den Anfangszustand von M_2 übergeht und jetzt wie die Turingmaschine M_2 arbeitet. Sie stoppt dann, wenn M_2 einen Stopzustand erreichen würde.

Beispiel für Nacheinanderausführung von TM

Betrachten wir das Diagramm in folgender Abbildung.

$$
\begin{array}{ccccc}
\text{start} & \longrightarrow & \text{“Band} := \text{Band} + 1\text{”} & & \\
 & & \downarrow & & \\
 & & \text{“Band} := \text{Band} + 1\text{”} & & \\
 & & \downarrow & & \\
 & & \text{“Band} := \text{Band} + 1\text{”} & \longrightarrow & \text{stop}
\end{array}
$$

So erkennen wir, dass dort das schematische Flussbild einer Turingmaschine steht, welche dreimal nacheinander zur Zahl auf dem Band 1 addiert, also es sich um die Turingmaschine “Band $:=$ Band $+ 3$” handelt.

Beispiel für Verzweigung von TM

Folgende Abbildung stellt eine sich verzweigende TM dar.

$$\begin{tikzcd} \text{start} \arrow[r] & M \arrow[r, "z_{e1}"] \arrow[d, "z_{e2}"] & M_1 \arrow[r] & \text{stop} \\ & M_2 \arrow[r] & \text{stop} & \end{tikzcd}$$

Sie soll nach Simulation der Turingmaschine M die Turingmaschine M_1 simulieren, falls sie bei der Simulation von M im Zustand z_{e1} stoppt. Analog soll sie die Turingmaschine M_2 abarbeiten, falls sie bei der Simulation von M im Zustand z_{e2} stoppt.

Beispiel für “Band (i) = 0 ?”

Es sei $M = (\{z_0, z_1, \textit{ja}, \textit{nein}\}, \Sigma, \Gamma, \delta, z_0, \Box, \{\textit{ja}, \textit{nein}\})$ mit $0 \in \Sigma$ sowie mit der Überführungsfunktion δ, gegeben durch

$$\begin{aligned}
\delta(z_0, a) &= (\textit{nein}, a, N) \quad \text{für } a \neq 0, \\
\delta(z_0, 0) &= (z_1, 0, R), \\
\delta(z_1, a) &= (\textit{nein}, a, L) \quad \text{für } a \neq \Box, \\
\delta(z_1, \Box) &= (\textit{ja}, \Box, L).
\end{aligned}$$

Diese Turingmaschine testet, ob die Eingabe genau das Wort 0 ist. Falls ja, stoppt sie im Zustand *ja*, falls nein, stoppt sie im Zustand *nein*. Wir wollen diese Turingmaschine mit “Band $= 0$?” bezeichnen.

Die k-Band-TM, die das i-te Band auf 0 testet, nennen wir “Band$(i) = 0$?”

Beispiel für WHILE-Schleife

Sei M eine beliebige Turingmaschine.

$$
\begin{tikzcd}
\text{start} \arrow[r] & \text{“Band } (i) = 0\,?\text{”} \arrow[r, "ja"] \arrow[d, shift left, "nein"] & \text{stop} \\
& M \arrow[u, shift left, bend left]
\end{tikzcd}
$$

Dann bezeichnen wir die Turingmaschine, die durch das Diagramm gegeben ist, mit “WHILE Band$(i) \neq 0$ DO M”. Die Arbeitsweise ist einfach zu erkennen.