Grundlagen der Theoretischen Informatik

Wintersemester 2024/25

Prof. Dr. Heribert Vollmer

Institut für Theoretische Informatik
Leibniz Universität Hannover

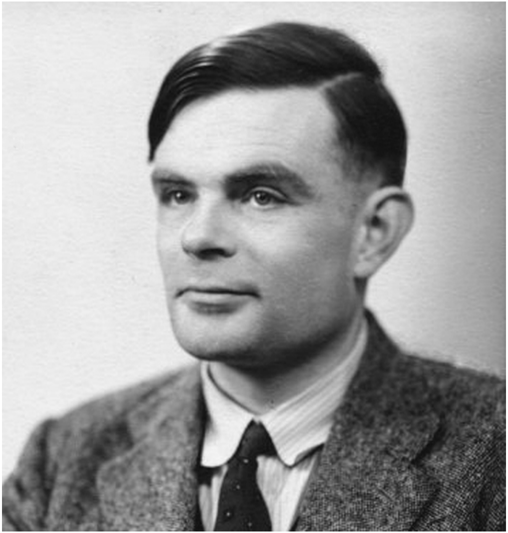
Link für Kurzfragen:

URL: https://pingo.coactum.de/events/739935

Inhalt

Typ-1- und Typ-0-Sprachen

Alan Turing

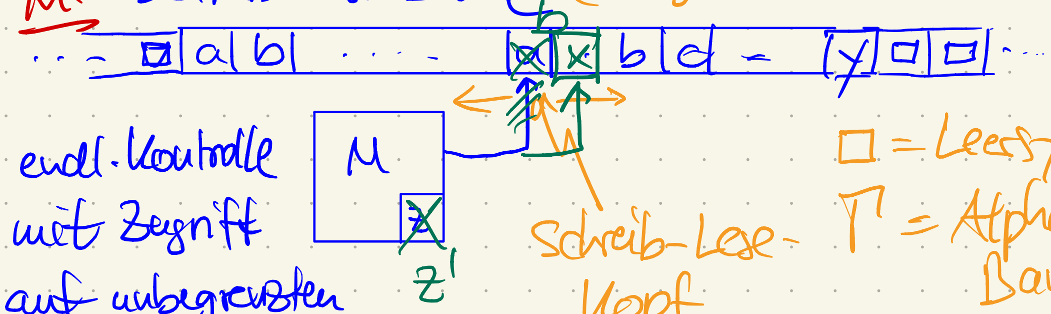


Geboren: 23. Juni 1912, Maida Vale

Gestorben: 7. Juni 1954, Wilmslow, Vereinigtes Königreich

Turing-Maschinen

M: Schreib-Lese-Band (Eingabe + Hilfsinformation)



endl. Kontrolle mit Zugriff auf unbegrenzten Hilfsspeicher (bel. Zugriff, Größe unbeschränkt)

Schreib-Lese-Kopf kann in beiden Richtungen bewegt werden

$\square$ = Leersymbol

Γ = Alphabet des Bandsymbole

Konfiguration:

$$k = \underbrace{ab \dots b}_{u}\,\underbrace{z'}_{q}\,\underbrace{xbc \dots y}_{v}$$

u: Band links vom Kopf

q: akt. Zustand

v: Band rechts vom Kopf

Abhängig von

- aktuellen Zustand sowie
- gelesenem Zeichen auf Band

führt die TM M folgende Aktionen durch:

- sie wechselt in neuen Zustand
- sie ersetzt das gelesene Zeichen durch ein neues
- sie bewegt den Kopf nach
  - L – links
  - N – keine Kopfbewegung
  - R – rechts

$$\delta : Z \times \Gamma \longrightarrow \mathcal{P}(Z \times \Gamma \times \{L, N, R\})$$

(D)TM / NTM

$$\delta(z, a) = (z', b, R)$$

Falls M in Zust. z ein a liest, so wechselt in Zust. z', schreibt b und bewegt Kopf nach rechts.

Definition

Eine Turingmaschine (TM) ist ein 7-Tupel

$$M = (Z, \Sigma, \Gamma, \delta, z_0, \square, E),$$

wobei für die einzelnen Komponenten gilt:

- Z ist die Menge der Zustände
- Σ ist das Eingabealphabet
- $\Gamma \supset \Sigma$ ist das Arbeitsalphabet
- $z_0 \in Z$ ist der Startzustand
- $\square \in \Gamma \setminus \Sigma$ ist das Leerzeichen bzw. Blank
- $E \subseteq Z$ ist die Menge der Endzustände
- δ ist die Übergangsfunktion

Definition (Fortsetzung)

Bei deterministischen Turingmaschinen (DTM, TM) gilt:

$$\delta\colon Z \times \Gamma \to Z \times \Gamma \times \{L, N, R\}$$

Bei nichtdeterministischen Turingmaschinen (NTM) gilt:

$$\delta\colon Z \times \Gamma \to \mathcal{P}(Z \times \Gamma \times \{L, N, R\})$$

Erläuterung der Arbeitsweise

Startkonfiguration:

M befindet sich am Anfang im Zustand z_0. Der Eingabekopf steht auf dem ersten Zeichen der Eingabe. Alle Bandzellen außerhalb der Eingabe enthalten das Leersymbol.

Zustandsübergang: (deterministischer Fall)

$\delta(z, a) = (z', b, X)$ bedeutet:
Ist M im Zustand z und liest das Eingabezeichen a, so geht M in den Zustand z' über, ersetzt das Eingabezeichen durch b und bewegt den Kopf gemäß X: $\mathrm{R} \triangleq$ rechts, $\mathrm{L} \triangleq$ links, $\mathrm{N} \triangleq$ neutral (keine Kopfbewegung).
$(z, z' \in Z,\ a, b \in \Gamma.)$

Nichtdeterministische Maschine: mehrere mögliche analoge Übergänge.

Erläuterung der Arbeitsweise

Ende der Rechnung:

M hält, sobald ein Zustand aus E angenommen wird.

Akzeptierte Sprache:

Ein Eingabewort x wird akzeptiert, falls in der Rechnung von M auf x irgendwann ein Zustand aus E angenommen wird.

$$L(M) = \{w \in \Sigma^* \mid M \text{ akzeptiert } w\}$$

Definition

Eine Konfiguration einer TM $M = (Z, \Sigma, \Gamma, \delta, z_0, \square, E)$ ist ein Wort $\boxed{k = uzv,}$ wobei $u, v \in \Gamma^*$ und $z \in Z$.

Startkonfiguration von M bei Eingabe w: $z_0 w$. $(u = \varepsilon, v = w)$

Konfiguration einer TM M (Momentaufnahme / vollst. Beschreibung):

- Zustand $z \in Z$
- Bandinhalt $w = uv \in \Gamma^*$
- Kopfposition: auf ersten Zeichen von v

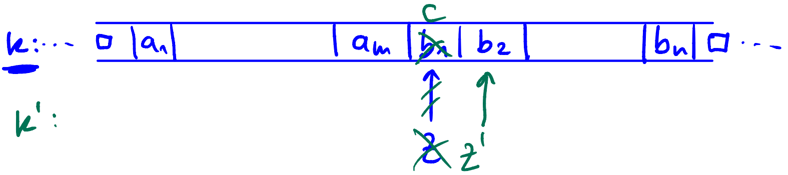
$k \vdash_M k'$

Definition

Sei $M = (Z, \Sigma, \Gamma, \delta, z_0, \square, E)$ eine TM. Wir definieren eine zweistellige Relation $\vdash_M$ auf der Menge der Konfigurationen wie folgt für $z \in Z \setminus E$:

$$\overbrace{a_1 \ldots a_m z b_1 \ldots b_n}^{k} \vdash$$

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



Sonderfälle

- $n = 1$, Maschine läuft nach rechts:

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

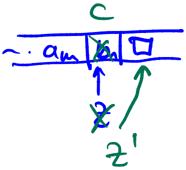
Konfiguration wird länger !

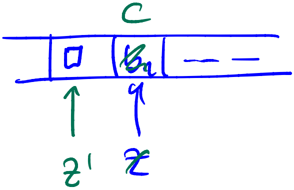
- $m = 0$, Maschine läuft nach links:

$$z b_1 \ldots b_n \vdash z' \square c b_2 \ldots b_n, \quad \text{falls } \delta(z, b_1) = (z', c, L),\ n \geq 1$$

- Für $z \in E$ gibt es keine Konfiguration k mit

$$a_1 \ldots a_m z b_1 \ldots b_n \vdash k.$$





Definition

Die von einer Turingmaschine $M = (Z, \Sigma, \Gamma, \delta, z_0, \square, E)$ akzeptierte Sprache ist

Startkonfiguration

$$L(M) = \{w \in \Sigma^* \mid z_0 w \vdash^* uzv \text{ für ein } z \in E \text{ und } u, v \in \Gamma^*\}.$$

Endkonfiguration

Dabei ist $k_a \vdash^* k_e$, falls $k_a = k_e$ oder es $k_1, \ldots, k_n$ gibt mit

Konfigurationenübergang in mehreren Schritten

$$k_a \vdash k_1 \vdash \cdots \vdash k_n \vdash k_e.$$

Also: Ein Wort wird akzeptiert, falls irgendwann ein Endzustand angenommen wird.

M deterministisch

- $k \vdash k'$ für eindeutiges k', falls Zustand in k aus $Z \setminus E$
- keine Folgekonf. für k, falls Endzustand

M nichtdeterministisch

- $k \vdash k'$, $k \vdash k''$ möglich für $k' \neq k''$
  (kann eindeutig sein)
- keine Folgekonf. für k, insbesondere für Endzustand
  ($\delta(z,a) = \emptyset$)

Beispiel: TM, die zu einer Eingabezahl in Binärdarstellung Eins addiert und Kopf auf erstes Zeichen bringt

$s: n \mapsto n+1$ Nachfolgerfunktion

$$\begin{array}{r} 1010 \\ +\quad 1 \\ \hline 1011 \end{array} \qquad \begin{array}{r} 1011 \\ +\ {}_1{}_1 1 \\ \hline 1100 \end{array} \qquad \begin{array}{r} 1111 \\ {}_1{}_1{}_1 1 \\ \hline 10000 \end{array}$$

$M = (\{z_0, z_1, z_2, z_e\}, \{0,1\}, \{0,1,\square\}, \delta, z_0, \square, \{z_e\})$, wobei δ wie folgt:

~~$\delta(z_0, 0) = (z_0, 0, R)$~~

~~$\delta(z_0, 1) = (z_0, 1, R)$~~

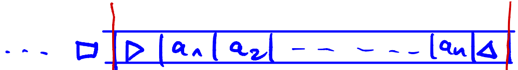
Kopf an rechtes Eingabeende	$z_0 0$	$\rightarrow$	$z_0 0 R$	(1)
	$z_0 1$	$\rightarrow$	$z_0 1 R$	(2)
	$z_0 \square$	$\rightarrow$	$z_1 \square L$	(3)
Addition 0→1 fertig 1→0 & weiter	$z_1 0$	$\rightarrow$	$z_2 1 L$	(4)
	$z_1 1$	$\rightarrow$	$z_1 0 L$	(5)
	$z_1 \square$	$\rightarrow$	$z_e 1 N$	(6)
Kopf an linkes Bandende	$z_2 0$	$\rightarrow$	$z_2 0 L$	(7)
	$z_2 1$	$\rightarrow$	$z_2 1 L$	(8)
	$z_2 \square$	$\rightarrow$	$z_e \square R$	(9)

Beispielrechnung:

$$\begin{aligned}
\underline{z_0 101} &\vdash^{(2)} 1 z_0 01 \\
&\vdash^{(1)} 10 z_0 1 \\
&\vdash^{(2)} 101 z_0 \square \\
&\vdash^{(3)} 10 z_1 1 \square \\
&\vdash^{(5)} 1 z_1 0 0 \square \\
&\vdash^{(4)} z_2 1 1 0 \square \\
&\vdash^{(8)} z_2 \square 1 1 0 \square \\
&\vdash^{(9)} \square z_e \underline{110} \square
\end{aligned}$$

$$\begin{array}{r} 101 \\ 1 \\ \hline 110 \end{array} \ \checkmark$$

Definition



Ein linear-beschränkter Automat (LBA) ist eine NTM

$$M = (Z, \Sigma, \Gamma, \delta, z_0, \square, E)$$

Kopf bewegt sich nur im Bereich der Eingabe

mit folgenden Eigenschaften:

- $\Gamma \setminus \Sigma$ enthält zwei spezielle Symbole $\triangleright$ und $\triangleleft$, die so genannte linke bzw. rechte Bandendemarkierung
- Falls M $\triangleright$ liest, ist keine Kopfbewegung nach links erlaubt
- Falls M $\triangleleft$ liest, ist keine Kopfbewegung nach rechts erlaubt
- Die Bandsymbole $\triangleright$ und $\triangleleft$ dürfen nicht durch andere Zeichen überschrieben werden

Die von M akzeptierte Sprache ist

$$L(M) = \{w \in \Sigma^* \mid z_0 \triangleright w \triangleleft \vdash^* uzv \text{ für ein } z \in E \text{ und } u, v \in \Gamma^*\}.$$

Satz

1. Eine Sprache L ist kontextsensitiv (Typ 1) gdw. es einen LBA gibt mit $\mathrm{L}(M) = \mathrm{L}$
2. Eine Sprache L ist vom Typ 0 gdw. es eine TM M gibt mit $\mathrm{L}(M) = \mathrm{L}$ gdw. es eine NTM M gibt mit $\mathrm{L}(M) = \mathrm{L}$

Bemerkung

Es ist unbekannt, ob deterministische LBAen nicht schon die Klasse der Typ-1-Sprachen akzeptieren.

LBA-Problem: Gibt es für jede Typ-1-Sprache einen deterministischen LBA, der sie akzeptiert?