# Formale Sprachen und Komplexitätstheorie

WS 2019/20

**Robert Elsässer**

# 1. Einführung

## Definition

Eine *(deterministische 1-Band) Turingmaschine* (DTM) wird beschrieben durch ein 7-Tupel $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$.

Dabei sind $Q, \Sigma, \Gamma$ endliche, nichtleere Mengen und es gilt:

- $\Sigma$ ist Teilmenge von $\Gamma$
- $t$ in $\Gamma$ ist das *Blanksymbol* (auch ⊔)
- $Q$ ist die *Zustandsmenge*
- $\Sigma$ ist das *Eingabealphabet*
- $\Gamma$ ist das *Bandalphabet*
- $q_0$ in $Q$ ist der *Startzustand*
- $q_{accept}$ in $Q$ ist der akzeptierende Endzustand
- $q_{reject}$ in $Q$ ist der ablehnende Endzustand
- $\delta$: $Q \setminus \{q_{accept}, q_{reject}\} \times \Gamma \to Q \times \Gamma \times \{L, R\}$ ist die (partielle) *Übergangsfunktion*. Sie ist für kein Argument aus $\{q_{accept}, q_{reject}\} \times \Gamma$ definiert.

# 1. Einführung

## Momentaufnahme einer Turingmaschine:

- Bei Bandinschrift $uv$ (dabei beginnt $u$ am linken Ende des Bandes und hinter $v$ stehen nur Blanks)
- Zustand $q$
- Kopf auf erstem Zeichen von $v$

Konfiguration $C = uqv$

# 1. Einführung

**Definition**

- Eine Sprache $L$ heißt **rekursiv aufzählbar,** falls es eine Turingmaschine $M$ gibt, die $L$ *akzeptiert.*
- Eine Sprache L heißt **rekursiv** oder **entscheidbar,** falls es eine Turingmaschine $M$ gibt, die $L$ *entscheidet.*

# 1. Einführung

- Eine Mehrband- oder $k$-Band Turingmaschine ($k$-Band DTM) hat $k$ Bänder mit je einem Kopf.
- Die Übergangsfunktion ist dann von der Form $\delta: Q \times \Gamma^k \to Q \times \Gamma^k \times \{L, R, S\}^k$
- Zu Beginn steht die Eingabe auf Band $1$, sonst stehen überall Blanks. Die Arbeitsweise ist analog zu $1$-Band-DTMs definiert.

# 2. Berechenbarkeit

- **Universelle Turingmaschinen**
  - Bislang *special purpose Computer*:
    eine Sprache – eine Turing-Maschine
  - Allgemein programmierbare Turing-Maschinen:
    **universelle Turing-Maschinen**
  - Erhalten als Eingabe die Beschreibung einer Turingmaschine und simulieren diese Maschine
  - Benötigen dafür eine einheitliche Beschreibung von Turingmaschinen durch sog. *Gödel-Nummern*

# 2. Berechenbarkeit

**Definition Gödelnummern**

Sei $M$ eine 1-Band-Turingmaschine mit

$$Q = \{q_0, \dots, q_n\},$$
$$q_{accept} = q_{n-1},$$
$$q_{reject} = q_n.$$

Sei $X_1 = 0, X_2 = 1, X_3 = t, D_1 = L, D_2 = R$.

Wir kodieren $\delta\left(q_i, X_j\right) = (q_k, X_l, D_m)$ durch $0^{i+1}10^j10^{k+1}10^l10^m$.

$Code_r$: Kodierung des $r$-ten Eintrags für $\delta, 1 \leq r \leq 4(n-1)$

Gödelnummer $\langle M \rangle = 111Code_1 11Code_2 11 \dots 11Code_g 111$

# 2. Berechenbarkeit

$M = (Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$

$\Sigma = \{0,1\}$

$\Gamma = \Sigma \cup \{t\}$

$\delta(q_0, 0) = (q_{reject}, 0, R)$

$\delta(q_0, 1) = (q_0, 1, R)$

$\delta(q_0, t) = (q_{accept}, t, R)$

$L = \{1^n \mid n \geq 0\}$

## Gödel-Nummer:

$1110101000101001101001010010011010001001000100111$

# 2. Berechenbarkeit

**Definition Universelle Turingmaschine**

Eine Turingmaschine $M_0$ heißt **universell**, falls für jede $1$-Band-Turingmaschine $M$ und jedes $x$ aus $\{0,1\}^*$ gilt:

- $M_0$ gestartet mit $\langle M \rangle x$ hält genau dann, wenn $M$ gestartet mit $x$ hält.
- $M_0$ akzeptiert $\langle M \rangle x$ genau dann, wenn $M$ das Wort $x$ akzeptiert.

**Satz**

Es gibt eine universelle $2$-Band Turingmaschine.

# 2. Berechenbarkeit

## Die Sprache $\mathrm{Gödel}$:

Sprache $\mathrm{Gödel} \coloneqq \{w \text{ aus } \{0,1\}^* \mid w \text{ ist die Gödel}-\text{Nummer einer DTM}\}$

**Lemma**

Die Sprache $\mathrm{Gödel}$ ist entscheidbar.

## Die Sprache $\mathit{States}$:

Sprache $\mathrm{States} \coloneqq \{(\langle M \rangle, d) \mid M \text{ besitzt mindestens } d \text{ Zustände}\}$

**Lemma**

Die Sprache $\mathrm{States}$ ist entscheidbar.

# 2. Berechenbarkeit

## Das Halteproblem

$H \coloneqq \{(\langle M \rangle, x) \mid M \text{ ist DTM, die gestartet mit Eingabe } x \text{ hält}\}$

**Satz**

Das Halteproblem ist rekursiv aufzählbar.

# 2. Berechenbarkeit

## Die Sprache $\mathrm{Useful}$

$$\mathrm{Useful} \coloneqq \left\{ \begin{matrix} (\langle M \rangle, q) \mid M \text{ ist DTM mit Zustand } q, \text{ und es gibt eine Eingabe } w, \\ \text{so dass } M \text{ gestartet mit } w \text{ in den Zustand } q \text{ gerät} \end{matrix} \right\}$$

**Satz**

Die Sprache $\mathrm{Useful}$ ist rekursiv aufzählbar.

# 2. Berechenbarkeit

## Aufzählung von binären Eingabefolgen:

- für alle natürlichen Zahlen $i$ sei $w_i = w$, falls $bin(i) = 1w$
- damit werden alle möglichen $w$ aus $\{0,1\}^*$ aufgezählt

## Aufzählung von Turingmaschinen:

$M_i$ ist:

- $M_{reject}$, falls $i$ keine Gödelnummer ist
- $M$, falls $bin(i)$ die Gödelnummer der DTM $M$ ist, d.h. $\langle M \rangle = bin(i)$

## Die Sprache $\mathrm{Diag}$

$\mathrm{Diag} \coloneqq \{w \text{ in } \{0,1\}^* \mid w = w_i \text{ und die DTM } M_i \text{ akzeptiert } w \text{ nicht}\}$

**Satz**

Die Sprache $\mathrm{Diag}$ ist **nicht** rekursiv aufzählbar.

# 2. Berechenbarkeit

| | $M_1$ | $M_2$ | $M_3$ | … … | $M_7$ | … … | $M_i$ | … … |
|---|---|---|---|---|---|---|---|---|
| $w_1$ | $na$ | $na$ | $na$ | | $na$ | | $na$ | |
| $w_2$ | $na$ | $na$ | $na$ | | $na$ | | $na$ | |
| $w_3$ | $na$ | $na$ | $na$ | | $na$ | | $na$ | |
| $\vdots$ | | | | | | | | |
| $w_7$ | $na$ | $na$ | $na$ | | $na$ | | $a$ | |
| $\vdots$ | | | | | | | | |
| $w_i$ | $na$ | $na$ | $na$ | | $na$ | | $a$ | |

Diagonale

Tabelle für Akzeptanz/Nichtakzeptanz von DTMs

**Quelle:** Skript Johannes Blömer, Universität Paderborn

# 2. Berechenbarkeit

## Reduktionen

Formalisierung von

- Sprache $A$ ist nicht schwerer als Sprache $B$

## Idee

- Algorithmus/DTM für $B$ kann genutzt werden,
  um $A$ zu akzeptieren/entscheiden.

# 2. Berechenbarkeit

## Zwei einfache Sprachen

$P := \{w \text{ in } \{0,1\}^* \mid w \text{ ist ein Palindrom}\}$

$$XOR := \left\{ \begin{array}{r} (a,b,c) \text{ in } \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^* \mid a,b,c \text{ haben die gleiche} \\ \text{Länge und } a \oplus b = c \end{array} \right\}$$

$f: \{0,1\}^* \to \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^*$

$w \to \left(w, w^R, 0^{|w|}\right)$

# 2. Berechenbarkeit

## Von $XOR$ und $f$ zu $P$

$M_P$ bei Eingabe $w$

1. Berechne mit $M_f$ das Tripel $f(w) = (w, w^R, 0^{|w|})$.
2. Simuliere $M_{XOR}$ mit Eingabe $f(w)$.
3. Falls $M_{XOR}$ die Eingabe $f(w)$ akzeptiert, akzeptiere $w$.
4. Falls $M_{XOR}$ die Eingabe $f(w)$ ablehnt, lehne $w$ ab.

$M_{XOR}$ entscheidet $XOR$, $M_f$ berechnet $f$.

# 2. Berechenbarkeit

**Definition Reduktionen**

$L'$ heißt reduzierbar auf $L$, falls es eine Funktion $f: \{0,1\}^* \to \{0,1\}^*$ gibt mit

1. Für alle $w$ aus $\{0,1\}^*$ gilt:
   $w$ ist in $L'$ genau dann, wenn $f(w)$ in $L$
2. Funktion $f$ ist berechenbar, d.h., es gibt eine DTM $M_f$, die die Funktion $f$ berechnet.

$f$ heißt Reduktion von $L'$ auf $L$, geschrieben $L' \leq L$.

# 2. Berechenbarkeit

**Definition**

Eine DTM $M$ berechnet die Funktion $f\colon \Sigma^* \to \Gamma$, falls für alle $w$ aus $\Sigma^*$ die Berechnung von $M$ mit Eingabe $w$ in einer akzeptierenden Konfiguration hält und dabei der Bandinhalt $f(w)$ ist.

Hierbei werden ▶ und alle $t$ ignoriert.

**Lemma**

Seien $L'$ und $L$ Sprachen mit $L' \leq L$. Dann gilt:

1. Ist $L$ entscheidbar, so ist auch $L'$ entscheidbar.
2. Ist $L$ rekursiv aufzählbar, so ist auch $L'$ rekursiv aufzählbar.

## Von $L$ und $f$ zu $L'$

$M'$ bei Eingabe $w$

1. Berechne mit $M_f$ die Folge $f(w)$.
2. Simuliere $M$ mit Eingabe $f(w)$.
3. Falls $M$ die Eingabe $f(w)$ akzeptiert, akzeptiere $w$.
4. Falls $M$ die Eingabe $f(w)$ ablehnt, lehne $w$ ab.

## Akzeptanz- und Halteproblem

$H \coloneqq \{\langle M \rangle x \mid M \text{ ist DTM, die gestartet mit Eingabe } x \text{ hält}\}$

$A \coloneqq \{\langle M \rangle x \mid M \text{ ist DTM, die die Eingabe } x \text{ akzeptiert}\}$

**Lemma**

Das Halteproblem kann auf das Akzeptanzproblem reduziert werden.

$$H \leq A$$