

# Grundlagen der Theoretischen Informatik

## Turingmaschinen und rekursiv aufzählbare Sprachen (VI)

10.07.2019

Viorica Sofronie-Stokkermans

e-mail: [sofronie@uni-koblenz.de](mailto:sofronie@uni-koblenz.de)

# Übersicht

---

1. Motivation
2. Terminologie
3. Endliche Automaten und reguläre Sprachen
4. Kellerautomaten und kontextfreie Sprachen
5. Turingmaschinen und rekursiv aufzählbare Sprachen
6. Berechenbarkeit, (Un-)Entscheidbarkeit
7. Komplexitätsklassen P und NP

# Übersicht

---

- Determinierte Turing-Maschinen (DTMs)
- Varianten von Turing-Maschinen
- Indeterminierte Turing-Maschinen (NTMs)
- Universelle determinierte Turing-Maschinen
- Entscheidbar/Aufzählbar
- Determinierte Turing-Maschinen entsprechen Typ 0
- **Unentscheidbarkeit**

# Gödelisierung von DTMs

## Definition (Gödelnummern von DTMs)

DTMs werden als Gödelzahlen kodiert:

- DTMs können als Gödelwörter dargestellt werden.
- Die Buchstaben der Gödelwörter können in Ziffern kodiert werden, um Gödelnummern zu bekommen.
- Notation:  $\hat{g}(\mathcal{M})$  für die Gödelnummer der DTM  $\mathcal{M}$ .

**Definition** Jede natürliche Zahl  $n$  soll Gödelnummer einer DTM  $\mathcal{M}_n$  sein.

Wir definieren

$$\mathcal{M}_n := \begin{cases} \mathcal{M}, & \text{falls } \hat{g}(\mathcal{M}) = n \\ \mathcal{M}_{halt} & \text{falls } \nexists \mathcal{M} \hat{g}(\mathcal{M}) = n \end{cases}$$

$\mathcal{M}_{halt}$  ist eine TM, die sofort anhält und weiter nichts tut.

# Halteproblem

---

## Definition [Allgemeines Halteproblem]

Das **allgemeine Halteproblem** ist die Frage, ob die DTM mit Gödelnummer  $n$  bei Eingabe  $i$  hält.

Es entspricht der Sprache

$$\mathcal{H}_{allg} := \{ \langle n, i \rangle \mid \mathcal{M}_n \text{ hält bei Eingabe } i \}.$$

# Halteproblem

---

## Definition [Spezielles Halteproblem]

Das **spezielle Halteproblem** ist die Frage, ob die DTM mit Gödelnummer  $n$  bei Eingabe  $n$  hält.

Es entspricht der Sprache

$$\mathcal{H} := \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n\}.$$

# Halteproblem

---

## Definition [Null-Halteproblem]

Das **Null-Halteproblem** ist die Frage, ob die DTM mit Gödelnummer  $n$  bei leerer Eingabe hält.

Es entspricht der Sprache

$$\mathcal{H}_0 := \{n \mid \mathcal{M}_n \text{ hält bei leerer Eingabe}\}$$

Manchmal auch:

“bei Eingabe 0” anstatt “bei leerer Eingabe”.

# Leerheitsproblem

---

## Definition [Leerheitsproblem]

Das **Leerheitsproblem** ist die Frage, ob die DTM mit Gödelnummer  $n$  bei **keiner** Eingabe aus  $\Sigma^*$  hält.

Es entspricht der Sprache

$$\mathcal{E} := \{n \mid \mathcal{M}_n \text{ hält bei keiner Eingabe aus } \Sigma^*\}.$$



# Totalitätsproblem

---

## Definition [Totalitätsproblem]

Das **Totalitätsproblem** ist die Frage, ob die DTM mit Gödelnummer  $n$  bei **jeder** Eingabe aus  $\Sigma^*$  hält.

Es entspricht der Sprache

$$\mathcal{T} := \{n \mid \mathcal{M}_n \text{ hält bei jeder Eingabe aus } \Sigma^*\}.$$

# Gleichheitsproblem

---

## Definition [Gleichheitsproblem]

Das **Gleichheitsproblem** ist die Frage, ob die DTM mit Gödelnummer  $n$  die gleiche Sprache über  $\Sigma$  akzeptiert wie die DTM mit Gödelnummer  $m$ .

Es entspricht der Sprache

$$\mathcal{E}q := \{ \langle n, m \rangle \mid \mathcal{M}_n \text{ akzeptiert die gleiche Sprache über } \Sigma \text{ wie } \mathcal{M}_m \}.$$

# Entscheidbarkeitsproblem

---

## Definition [Entscheidbarkeitsproblem]

Das **Entscheidbarkeitsproblem** ist die Frage, ob die DTM mit Gödelnummer  $n$  eine entscheidbare Sprache über  $\Sigma$  akzeptiert.

Es entspricht der Sprache

$$\mathcal{E}nt := \{n \mid \mathcal{M}_n \text{ akzeptiert eine entscheidbare Sprache über } \Sigma\}.$$

# Unentscheidbarkeit des Halteproblems

---

## **Theorem (Halteproblem ist unentscheidbar)**

Das spezielle Halteproblem  $\mathcal{H} := \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n\}$  ist unentscheidbar.

# Unentscheidbarkeit des Halteproblems

---

## Theorem (Halteproblem ist unentscheidbar)

Das spezielle Halteproblem  $\mathcal{H} := \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n\}$  ist unentscheidbar.

Beweis (A. Turing)

Beweis durch Widerspruch mit einem **Diagonalisierungsargument**.

Angenommen, es gäbe eine DTM  $\mathcal{M}_{\mathcal{H}}$ , die das spezielle Halteproblem  $\mathcal{H}$  entscheidet.

Konstruiere eine neue Maschine  $\mathcal{M}'$  aus  $\mathcal{M}_{\mathcal{H}}$ :

- Wenn  $\mathcal{M}_{\mathcal{H}}$  „Y“ antwortet, geht  $\mathcal{M}'$  in eine Endlosschleife (terminiert nicht).
- Wenn  $\mathcal{M}_{\mathcal{H}}$  „N“ antwortet, terminiert  $\mathcal{M}'$ .

Die neue Maschine habe Gödelnummer  $n$  (also:  $\mathcal{M}_n = \mathcal{M}'$ )

# Unentscheidbarkeit des Halteproblems

---

Beweis (A. Turing), Forts.

Was macht  $\mathcal{M}_n$  bei Eingabe  $n$ ?

- Falls  $\mathcal{M}_n$  bei Eingabe  $n$  terminiert, dann antwortet  $\mathcal{M}_{\mathcal{H}}$  auf Eingabe  $n$  mit „Y“, dann terminiert  $\mathcal{M}_n$  auf Eingabe von  $n$  **nicht**  
Widerspruch!
- Falls  $\mathcal{M}_n$  bei Eingabe  $n$  nicht terminiert, dann antwortet  $\mathcal{M}_{\mathcal{H}}$  auf Eingabe  $n$  mit „N“, dann terminiert  $\mathcal{M}_n$  auf Eingabe von  $n$   
Widerspruch!

# Akzeptierbarkeit des Halteproblems

---

## Theorem (Akzeptierbarkeit von $\mathcal{H}$ )

Das spezielle Halteproblem  $\mathcal{H}$  ist aufzählbar. D. h. die Sprache

$$\mathcal{H} = \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n \}$$

ist akzeptierbar.

# Akzeptierbarkeit des Halteproblems

---

## Theorem (Akzeptierbarkeit von $\mathcal{H}$ )

Das spezielle Halteproblem  $\mathcal{H}$  ist aufzählbar. D. h. die Sprache

$$\mathcal{H} = \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n \}$$

ist akzeptierbar.

Beweis:

**Idee:**

Akzeptieren durch Simulation von  $\mathcal{M}_n$  mit Hilfe der universellen DTM.



# Akzeptierbarkeit des Halteproblems

---

## Theorem (Akzeptierbarkeit von $\mathcal{H}$ )

Das spezielle Halteproblem  $\mathcal{H}$  ist aufzählbar. D. h. die Sprache

$$\mathcal{H} = \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n \}$$

ist akzeptierbar.

Beweis:

Sei  $\mathcal{M}_{\text{prep}}$  die TM, die die Eingabe auf dem Arbeitsband in die Form bringt, die  $\mathcal{U}$  fordert.

Sei  $\mathcal{M}_{\mathcal{H}} := \mathcal{M}_{\text{prep}}\mathcal{U}$

$s, \#n\# \vdash_{\mathcal{M}_{\text{prep}}}^* q, \#n\#w_{\delta, \mathcal{M}_n}\#$   $\mathcal{U}$  simuliert dann die Arbeit von  $\mathcal{M}_n$  bei Input  $n$ .

$\mathcal{M}_{\mathcal{H}} := \mathcal{M}_{\text{prep}}\mathcal{U}$  hält bei Input  $n$  genau dann, wenn  $\mathcal{M}_n$  bei Input  $n$  hält  
genau dann, wenn  $n \in \mathcal{H}$ .

# Halteproblem

---

## Theorem (Halteproblem ist unentscheidbar)

Das spezielle Halteproblem  $\mathcal{H} := \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n\}$  ist unentscheidbar.

## Theorem (Akzeptierbarkeit von $\mathcal{H}$ )

Das spezielle Halteproblem  $\mathcal{H}$  ist aufzählbar. D. h. die Sprache

$$\mathcal{H} = \{n \mid \mathcal{M}_n \text{ hält bei Eingabe } n \}$$

ist akzeptierbar.

**Korollar** Das Komplement von  $\mathcal{H}$  ist nicht aufzählbar.

# Reduktion von Problemen

---

Wie zeigt man, dass ein Problem unentscheidbar ist?

## Reduktion (informell)

Wir geben eine **totale, berechenbare Funktion**  $f$  an, die

- eine Instanz  $p_1$  von  $P_1$
- in eine Instanz  $p_2$  von  $P_2$  umwandelt,
- und zwar so, dass die Antwort zu  $p_1$  „ja“ ist gdw die Antwort zu  $p_2$  „ja“ ist.

Wenn  $P_1$  unentscheidbar ist, dann ist auch  $P_2$  unentscheidbar.

# Reduktion von Problemen

---

## Definition

Seien  $L_1, L_2$  Sprachen über  $\mathbb{N}$ .

$L_1$  wird auf  $L_2$  reduziert,

$$L_1 \preceq L_2$$

gdw

es gibt eine TM-berechenbare Funktion  $f : \mathbb{N} \rightarrow \mathbb{N}$ , so dass gilt:

$$\forall n \in \mathbb{N} \quad (n \in L_1 \text{ gdw } f(n) \in L_2).$$

# Reduktion von Problemen

---

## Lemma

Ist  $L_1 \preceq L_2$ , und ist  $L_1$  **unentscheidbar**, so ist auch  $L_2$  **unentscheidbar**.

# Reduktion von Problemen

---

## Lemma

Ist  $L_1 \preceq L_2$ , und ist  $L_1$  **unentscheidbar**, so ist auch  $L_2$  **unentscheidbar**.

## Beweis

- Angenommen,  $L_2$  ist entscheidbar.
- Sei  $\mathcal{M}_2$  eine Turing-Maschine, die  $L_2$  entscheidet.
- Wegen  $L_1 \preceq L_2$  gibt es eine Funktion  $f: \mathbb{N} \rightarrow \mathbb{N} \in TM$  mit  $n \in L_1$  gdw  $f(n) \in L_2$ .
- Sei  $\mathcal{M}_f$  eine DTM, die  $f$  berechnet.
- Dann kann man daraus die Maschine  $\mathcal{M}_1 := \mathcal{M}_f \mathcal{M}_2$  konstruieren, für die gilt:
  - $\mathcal{M}_1$ , gestartet mit Input  $n$ , hält mit  $h, \#Y\#$ , falls  $f(n) \in L_2$ , d.h. wenn  $n \in L_1$  ist.
  - $\mathcal{M}_1$ , gestartet mit Input  $n$ , hält mit  $h, \#N\#$ , falls  $f(n) \notin L_2$ , d.h. wenn  $n \notin L_1$  ist.
- Die Maschine  $\mathcal{M}_1$  entscheidet also  $L_1$ , ein Widerspruch.

# Unentscheidbarkeit

---

**Theorem [Unentscheidbarkeit von  $\mathcal{H}_0$ ].**

Das Null-Halteproblem

$$\mathcal{H}_0 = \{n \mid \mathcal{M}_n \text{ h\u00e4lt bei Eingabe } 0\}$$

ist unentscheidbar.

# Unentscheidbarkeit

---

**Beweis:** Gegeben eine TM  $\mathcal{M}_n$ .

Kombiniere diese mit einer DTM, die  $n$  aufs Band schreibt.

$f(n)$  sei definiert als die Gödelnummer dieser neuen Maschine.

$\mathcal{M}_{f(n)}$  terminiert auf Eingabe von 0

gdw

$\mathcal{M}_n$  terminiert auf Eingabe von  $n$

Damit:

Reduktion des speziellen Halteproblems auf das Null-Halteproblem.

( $f$  ist total und berechenbar!)

Also:

Unentscheidbarkeit des Null-Halteproblems folgt aus

Unentscheidbarkeit des speziellen Halteproblems



# Unentscheidbarkeit

---

**Theorem [Unentscheidbarkeit von  $\mathcal{K}_0$ ].**

Das allgemeine Halteproblem

$$\mathcal{H}_{\text{allg}} = \{ \langle n, i \rangle \mid \mathcal{M}_n \text{ hält bei Eingabe } i \}$$

ist unentscheidbar.

# Unentscheidbarkeit

---

**Theorem [Unentscheidbarkeit von  $\mathcal{K}_0$ ].**

Das allgemeine Halteproblem

$$\mathcal{H}_{\text{allg}} = \{ \langle n, i \rangle \mid \mathcal{M}_n \text{ h\u00e4lt bei Eingabe } i \}$$

ist unentscheidbar.

Beweis 1:

Wir zeigen, dass  $\mathcal{H}_0 \preceq \mathcal{H}_{\text{allg}}$ :

Sei  $f : \mathbb{N} \rightarrow \mathbb{N}$  mit  $f(n) = \langle n, 0 \rangle$  (TM berechenbar).

Dann  $n \in \mathcal{H}_0$  gdw.  $\mathcal{M}_n$  h\u00e4lt bei Eingabe 0 gdw.  $f(n) \in \mathcal{H}_{\text{allg}}$ .

# Unentscheidbarkeit

---

**Theorem [Unentscheidbarkeit von  $\mathcal{K}_0$ ].**

Das allgemeine Halteproblem

$$\mathcal{H}_{\text{allg}} = \{ \langle n, i \rangle \mid \mathcal{M}_n \text{ h\u00e4lt bei Eingabe } i \}$$

ist unentscheidbar.

Beweis 2:

Wir zeigen, dass  $\mathcal{H} \preceq \mathcal{H}_{\text{allg}}$ :

Sei  $g : \mathbb{N} \rightarrow \mathbb{N}$  mit  $g(n) = \langle n, n \rangle$  (TM berechenbar).

Dann  $n \in \mathcal{H}$  gdw.  $\mathcal{M}_n$  h\u00e4lt bei Eingabe  $n$  gdw.  $g(n) \in \mathcal{H}_{\text{allg}}$ .

# Unentscheidbarkeit

---

Ähnlich kann man per Reduktion die Unentscheidbarkeit der folgenden Probleme zeigen:

- $\mathcal{E}$ , das Leerheitsproblem.

$$\mathcal{E} := \{n \mid \mathcal{M}_n \text{ hält bei keiner Eingabe aus } \Sigma^*\}$$

- $\mathcal{T}$ , das Totalitätsproblem.

$$\mathcal{T} := \{n \mid \mathcal{M}_n \text{ hält bei jeder Eingabe aus } \Sigma^*\}.$$

- $\mathcal{E}q$ , das Gleichheitsproblem.

$$\mathcal{E}q := \{\langle n, m \rangle \mid \mathcal{M}_n \text{ akzeptiert die gleiche Sprache über } \Sigma \text{ wie } \mathcal{M}_m \}.$$

- $\mathcal{E}nt$ , das Entscheidbarkeitsproblem.

$$\mathcal{E}nt := \{n \mid \mathcal{M}_n \text{ akzeptiert eine entscheidbare Sprache über } \Sigma\}.$$

# Übersicht

---

- Determinierte Turing-Maschinen (DTMs)
- Varianten von Turing-Maschinen
- Indeterminierte Turing-Maschinen (NTMs)
- Universelle determinierte Turing-Maschinen
- Entscheidbar/Aufzählbar
- Determinierte Turing-Maschinen entsprechen Typ 0
- **Unentscheidbarkeit**

# Übersicht

---

1. Motivation
2. Terminologie
3. Endliche Automaten und reguläre Sprachen
4. Kellerautomaten und kontextfreie Sprachen
5. Turingmaschinen und rekursiv aufzählbare Sprachen
6. Berechenbarkeit, (Un-)Entscheidbarkeit
7. Komplexitätsklassen P und NP

# Übersicht

---

1. Motivation
2. Terminologie
3. Endliche Automaten und reguläre Sprachen
4. Kellerautomaten und kontextfreie Sprachen
5. Turingmaschinen und rekursiv aufzählbare Sprachen
6. Berechenbarkeit, (Un-)Entscheidbarkeit
7. Komplexitätsklassen P und NP