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

Unentscheidbare Probleme

Erkennen von Endlosschleifen:

Das Halteproblem ist die Sprache

$$H = \{ \langle M, x \rangle \mid M \text{ hält bei Eingabe } x \}.$$

Sprache!

Wort

M, x muss über passenden Alphabet kodiert werden

Eingabe: TM M
Wort x auf Band von M

Frage: Hält M(x)?

Beispiel:

Eingabe: $n \in \mathbb{N}$

```
while n ≠ 1 do
    n := n - 2
end
```

modifizierte Subtraktion, um nicht-neg. Werte

Algorithmus hält, gdw. n ungerade

Eingabe: $n \in \mathbb{N}$

```
while n ≠ 1 do
    if n ist gerade then
        n := n div 2
    else
        n := 3*n+1
end
```

Beispiel:

$n = 3 \mid 10 \mid 5 \mid 16 \mid 8 \mid 4 \mid 2 \mid 1 \to$ hält

unbekannt, für welche n Algorithmus hält

Collatz-Vermutung: Alg. hält f. a. $n > 0$.

Kodierung von Turing-Maschinen durch Binärwörter:

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

$\Gamma = \{a_0, a_1, \ldots, a_k\}$, $a_0 = \square$,

$Z = \{z_0, z_1, \ldots, z_n\}$

Übergang in δ der Form

$$\delta(z_i, a_j) = (z_\ell, a_m, X)$$

wird kodiert durch

$\#\# \, bin(i) \# bin(j) \# bin(\ell) \# bin(m) \# r$, wobei

$$r = \begin{cases} 0, & \text{falls } X = L \\ 1, & \text{falls } X = R \\ 10, & \text{falls } X = N \end{cases}$$

M kodiert durch Konkatenation der Kodierungen aller Einträge in δ in beliebiger Reihenfolge $\rightarrow$ Wort über $\{0, 1, \#\}$

Blockkodierung:

$0 \rightarrow 00$

$1 \rightarrow 01$

$\# \rightarrow 11$

<u>Ergebnis</u>: M kodiert durch Wort über $\{0,1\}$,

äquivalent: M kodiert durch nat. Zahlen

TM $M \longrightarrow w \in \{0,1\}^*$

$n \in \mathbb{N}$

Nicht jedes $w \in \{0,1\}^*$ kommt als Kodierung einer TM vor.

Folgerung:

Menge der TM ist abzählbar (d.h. es gibt nicht mehr TM als nat. Zahlen)

Menge der Sprachen über bel. Alphabet / Menge der Fkt. $\mathbb{N} \to \mathbb{N}$ ist überabzählbar.

(d.h. es gibt mehr Sprachen als TM).

Folgerung:

Es gibt nicht entscheidbare Sprachen.

Es gibt nicht berechenbare Funktionen.

Gödelisierung

Kurt Gödel

Gödelisierung = Kodierung von Turing-Maschinen durch Binärwörter

Sei $w \in \{0,1\}^*$. Dann ist

$$M_w := \begin{cases} M, & \text{falls } w \text{ Gödelisierung von } M \\ \widehat{M}, & \text{sonst (d. h. } w \text{ ist keine gültige Gödelisierung),} \end{cases}$$

wobei $\widehat{M}$ eine festgehaltene Turingmaschine ist.

z.B. TM für w_0 aus der Vorlesung.

$M_0, M_1, M_2, M_3, \ldots$

$M \rightleftarrows w \in \{0,1\}^* \,/\, n \in \mathbb{N}$

Definition

Das spezielle Halteproblem ist die Sprache

$$K = \{ w \in \{0,1\}^* \mid M_w \text{ hält bei Eingabe } w \}.$$

Das (allgemeine) Halteproblem ist die Sprache

$$H = \{ w\#x \mid M_w \text{ hält bei Eingabe } x \}.$$

Sprache über $\{0,1,\#\}$ $\langle M,x \rangle \cong w\#x$

Beobachtung

K und H sind rekursiv-aufzählbar.

Beweis für H:

Semi-Entscheidungsalgorithmus für H:

Eingabe: $w\#x$

konstruiere TM $M_w =: M$;

simuliere M auf Eingabe x

Falls Simulation stoppt,

dann Ausgabe „1" + stopp.

(Ansonsten Endlosschleife).

Alg. gibt 1 aus, falls $w\#x \in H$

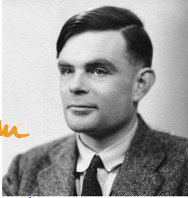
Alg. gerät in Endlosschleife sonst.

$\square$

Satz

K ist nicht entscheidbar.

Alan Turing, On computable numbers with an application to the Entscheidungsproblem (1936)



Beweis: ~~A~~ K wäre entscheidbar.

Sei M_K eine TM, die K entscheidet (d.h. c_K berechnet).

Definiere M wie folgt:

Start
↓
M_K
↓
Band = 0?
ja ↓ stopp
nein → Endlosschleife

Also: $M(w)$ stoppt
$\iff M_K(w) = 0$

M in Endlosschleife
$\iff M_K(w) = 1$

Sei w die Gödelisierung von M.

Nun gilt:

$w \in K \iff M(w)$ hält (Def. v. K)

$\iff M_K(w) = 0$ (Def. v. M)

$\iff w \notin K$ (Def. v. M_K)

Widerspruch!

Also kann M_K nicht existieren

K ist unentscheidbar.

□

Korollar

$\overline{\mathrm{K}}$ ist nicht rekursiv-aufzählbar.

<u>Beweis:</u> K ist r.a.

A $\overline{K}$ wäre r.a. $\Big\} \Longrightarrow$ K entscheidbar.

↯ zu Satz

□

Satz

H ist nicht entscheidbar.

Intuitiv: K ist „Spezialfall" von H.

Formal: Reduktion $K \leq H$.

Beweis: Wir zeigen $K \le H$.

$w \in K \iff f(w) \in H$

$\uparrow$ $\qquad\qquad\qquad \uparrow$

$M_w(w)$ hält $\qquad f(w) = w'\#x$, d.h. $M_{w'}(x)$ hält

Wähle $w' = w$, $x = w$

Def. $f(w) = w\#w$. Dann ist

$w \in K \iff f(w) = w\#w \in H$. $\square$

Definition

Seien $A \subseteq \Sigma^*$ und $B \subseteq \Gamma^*$ Sprachen.
A heißt auf B reduzierbar, in Zeichen: $A \le B$, falls es eine totale, berechenbare Funktion $f\colon \Sigma^* \to \Gamma^*$ gibt, sodass für alle $w \in \Sigma^*$ gilt:

$$\boxed{w \in A \Leftrightarrow f(w) \in B}$$

Dann: A nicht entscheidbar $\Rightarrow$ B nicht entscheidbar.

Methode zum Nachweis von Unentscheidbarkeit.

$H_{Java} = \{ \langle P, x \rangle \mid$ Programm P bei Eingabe x hält $\}$

„Haltproblem für Java"

<u>Satz</u>: H_{Java} ist nicht entscheidbar.

<u>Beweis</u>: $H \leq H_{Java}$

$f(w\#x) = \langle P, x \rangle$, wobei P ein Java-Programm ist, das M_w simuliert.

Dann: $w\#x \in H \Longleftrightarrow M_w(x)$ hält

$\Longleftrightarrow P(x)$ hält

$\Longleftrightarrow \langle P, x \rangle \in H_{Java}$. $\square$