Entscheidbarkeit, Unentscheidbarkeit und das Halteproblem

Prof. Dr. David Sabel

LFE Theoretische Informatik

Letzte Änderung der Folien: 2. Juli 2022

Entscheidbarkeit

Definition

Eine Sprache $L \subseteq \Sigma^*$ heißt **entscheidbar**, wenn die charakteristische Funktion von L, $\chi_L : \Sigma^* \to \{0, 1\}$ mit

$$\chi_L(w) = \begin{cases} 1, & \text{falls } w \in L \\ 0, & \text{falls } w \notin L \end{cases}$$

berechenbar ist.

algorithmisch:

Der χ_L-berechnende Algorithmus terminiert in jedem Fall und liefert ein Ergebnis.

Semi-Entscheidbarkeit

Definition

Eine Sprache heißt **semi-entscheidbar** falls $\chi'_L : \Sigma^* \to \{0, 1\}$ mit

$$\chi'_L(w) = \begin{cases} 1, & \text{falls } w \in L \\ \text{undefiniert}, & \text{falls } w \notin L \end{cases}$$

berechenbar ist.

algorithmisch:

Der χ'_L-berechnende Algorithmus terminiert nur, falls $w \in L$, und läuft anderenfalls endlos.

Zusammenhang: entscheidbar und semi-entscheidbar

Satz

Ein Sprache L ist genau dann entscheidbar, wenn L und $\overline{L}$ jeweils semi-entscheidbar sind.

Beweis:

„$\Rightarrow$“: Konstruiere aus TM, die χ_L berechnet, zwei TMs, die χ'_L und $\chi'_{\overline{L}}$ berechnen.

Zusammenhang: entscheidbar und semi-entscheidbar

Satz

Ein Sprache L ist genau dann entscheidbar, wenn L und $\overline{L}$ jeweils semi-entscheidbar sind.

Beweis:

„$\Rightarrow$“: Konstruiere aus TM, die χ_L berechnet, zwei TMs, die χ'_L und $\chi'_{\overline{L}}$ berechnen.

„$\Leftarrow$“: Gegeben TMs M_L und $M_{\overline{L}}$, die χ'_L und $\chi'_{\overline{L}}$ berechnen.
Konstruiere TM, die χ_L berechnet:

- Starte mit $i = 1$.
- Simuliere i-Schritte von M_L.
- Wenn diese akzeptiert, dann akzeptiere mit Ausgabe 1.
- Ansonsten simuliere i-Schritte von $M_{\overline{L}}$.
- Wenn diese akzeptiert, dann akzeptiere mit Ausgabe 0.
- Ansonsten erhöhe i um 1 und starte von neuem.

Komplement entscheidbar?

Korollar

Wenn L entscheidbar, dann ist auch $\overline{L}$ entscheidbar.

- da L entscheidbar, sind L und $\overline{L}$ semi-entscheidbar
- daher sind $\overline{\overline{L}} = L$ und $\overline{L}$ semi-entscheidbar
- daher ist $\overline{L}$ entscheidbar

Rekursive Aufzählbarkeit

Definition

Eine Sprache $L \subseteq \Sigma^*$ heißt **rekursiv aufzählbar**,

- falls $L = \emptyset$ oder
- falls es eine totale berechenbare Funktion $f : \mathbb{N} \to \Sigma^*$ gibt, sodass $L = \bigcup_{i \in \mathbb{N}} f(i)$. Man sagt dann „$f$ zählt L auf“.

Beispiel

Lemma

Die Sprache Σ^* ist rekursiv-aufzählbar.

Beweis: Sei $|\Sigma| = b$, $n \in \mathbb{N}$. Interpretiere $w \in \Sigma^*$ als $b+1$-äre Zahl.

- Konstruiere TM, die n in Binärdarstellung auf Eingabeband erhält.
- TM erzeugt auf anderem Band die $b+1$-äre Darstellung der 0
- Anschließend zählt die TM die Zahl auf dem Eingabeband um 1 herunter und die $b+1$-äre Zahl um 1 nach oben.
- Dies wird wiederholt bis auf dem Eingabeband die 0 steht
- Dann steht auf dem anderen Band $f(n)$.

Rekursiv aufzählbar = semi-entscheidbar

Satz

Eine Sprache ist genau dann rekursiv aufzählbar, wenn sie semi-entscheidbar ist.

Beweis:

„$\Rightarrow$“: Sei f die totale, berechenbare Funktion, die L aufzählt.
Dann berechnet der folgende Algorithmus $\chi'_L(w)$:

Für $i = 0, 1, 2, 3, \ldots$ **tue**
wenn $f(i) = w$ **dann**
stoppe und gebe 1 aus

„$\Leftarrow$“: ...

Rekursiv aufzählbar = semi-entscheidbar

„$\Leftarrow$“: Sei M eine TM, die χ'_L berechnet.

- Wenn $L = \emptyset$, dann ist L rekursiv-aufzählbar.
- Anderenfalls sei $u \in L$ ein Wort.
  Wir konstruieren TM M', die die L aufzählende Funktion berechnet.
- Sei n eine Eingabe. Wir interpretieren n als $c(x, y)$.
- M' simuliert y Schritte von M bei Eingabe $g(x)$, wobei g die Σ^* aufzählende Funktion.
- Wenn M nach y Schritten $g(x)$ akzeptiert, dann akzeptiert M'mit Ausgabe $g(x)$.

Anderenfalls, akzeptiert M' mit Ausgabe u.

- Die von M' berechnete Funktion:

$$f(n) = \begin{cases} w \in L, \text{falls } w = g(\mathit{left}(n)) \text{ und } M \text{ akzeptiert } w \text{ in } \mathit{right}(n) \text{ Schritten} \\ u \in L, \text{ sonst} \end{cases}$$

- f zählt L auf, da für jedes Wort w ein x existiert mit $g(x) = w$ und
  ein y existiert, sodass M mit Eingabe w nach y Schritten akzeptiert.

Zusammenfassung: Äquivalente Eigenschaften

Die folgenden Eigenschaften sind äquivalent:

- L ist vom Typ 0.
- L ist semi-entscheidbar.
- L ist rekursiv-aufzählbar.
- Es gibt eine Turingmaschine M, die L akzeptiert (d.h. $L(M) = L$).
- χ'_L ist Turing-, WHILE-, GOTO-berechenbar.
- Es gibt berechenbare Funktionen, die L als Wertebereich
  (nämlich die L aufzählende Funktion)
  bzw. als Definitionsbereich (nämlich χ'_L) haben.

Rekursiv aufzählbar $\neq$ abzählbar

- Sprache L ist abzählbar, wenn es eine totale Funktion $f : \mathbb{N} \to L$ gibt, sodass $\bigcup_{i \in \mathbb{N}} f(i) = L$.
- Beachte: Abzählbarkeit fordert nicht, dass f berechenbar ist!

Gödelisierung von Turingmaschinen

Ziel:

Stelle Turingmaschinenbeschreibung als natürliche Zahl in Binärdarstellung dar:

Grund:

Andere Turingmaschinen können die Beschreibung als Eingabe erhalten, erzeugen, usw.

Gödelisierung von Turingmaschinen (2)

Sei $(Z, \Sigma, \Gamma, \delta, z_0, \Box, E)$ eine DTM mit $\Sigma = \{0, 1\}$ und

- $\Gamma = \{a_0, \dots, a_k\}$ wobei $a_0 = \Box$, $a_1 = \#$, $a_2 = 0$, $a_3 = 1$
- $Z = \{z_0, \dots, z_n\}$
- $E = \{z_n\}$

Für $\delta(z_p, a_i) = (z_q, a_j, D)$ erzeuge Wort über Alphabet $\{0, 1, \#\}$:

$$w_{p,i,q,j,D} = \#\#bin(p)\#bin(i)\#bin(q)\#bin(j)\#bin(D_m)$$

mit $D_m = 0,$ falls $D = L, D_m = 1,$ falls $D = R, D_m = 2,$ falls $D = N$

Kodierung w_δ: Schreibe alle δ-Worte hintereinander.

Schließlich: Kodiere Alphabet $\{0, 1, \#\}$ durch $\{0 \mapsto 00, 1 \mapsto 01, \# \mapsto 11\}$.

Wende dies auf w_δ an.

Wir bezeichnen mit w_M die so kodierte TM M.

Gödelisierung von Turingmaschinen (3)

- Nicht jedes Wort über $\{0,1\}$ entspricht der Kodierung einer Turingmaschine.
- Sei $\widehat{M}$ eine beliebige aber feste Turingmaschine.
- Definiere für jedes $w \in \{0,1\}^*$ die zugehörige TM M_w:

$$M_w := \begin{cases} M, & \text{wenn } w = w_M \\ \widehat{M}, & \text{sonst} \end{cases}$$

Spezielles Halteproblem

Definition (Spezielles Halteproblem)

Das spezielle Halteproblem ist die Sprache

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

Unentscheidbarkeit von K

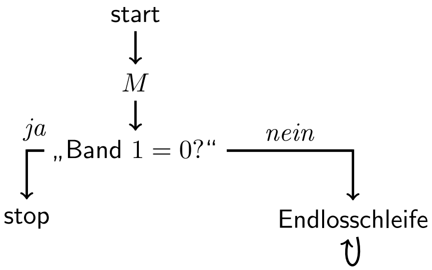
Satz

Das spezielle Halteproblem ist nicht entscheidbar (und damit *unentscheidbar*).

Beweis:

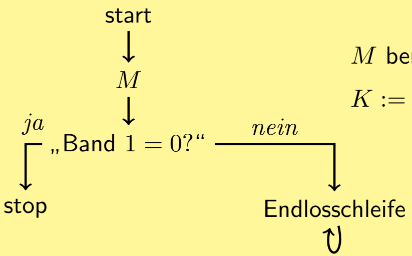
- Annahme: K ist entscheidbar.
- Dann ist χ_K berechenbar, und es gibt TM M, die χ_K berechnet.
- Konstruiere M':

1. M' lässt M ablaufen
2. Wenn M mit 0 auf dem Band endet, dann akzeptiert M'
3. Wenn M mit 1 auf dem Band endet, dann läuft M' in eine Endlosschleife.



Unentscheidbarkeit von K (2)

Zur Erinnerung:



Turingmaschine M'

M berechnet χ_K wobei K:

$K := \{w \in \{0,1\}^* \mid M_w$ hält für Eingabe $w\}$

Betrachte nun M' auf der Eingabe $w_{M'}$:

Es gilt: M' hält für Eingabe $w_{M'}$

g.d.w. M angesetzt auf $w_{M'}$ gibt 0 aus

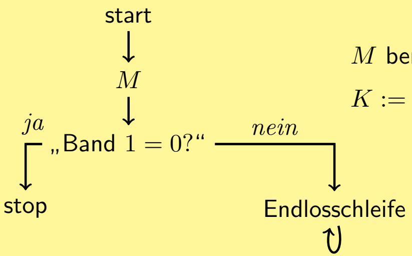
g.d.w. $\chi_K(w_{M'}) = 0$

g.d.w. $w_{M'} \notin K$

g.d.w. M' hält nicht für Eingabe $w_{M'}$

Unentscheidbarkeit von K (2)

Zur Erinnerung:



Turingmaschine M'

M berechnet χ_K wobei K:

$K := \{w \in \{0,1\}^* \mid M_w$ hält für Eingabe $w\}$

Betrachte nun M' auf der Eingabe $w_{M'}$:

Es gilt: M' hält für Eingabe $w_{M'}$

g.d.w. M angesetzt auf $w_{M'}$ gibt 0 aus

g.d.w. $\chi_K(w_{M'}) = 0$

g.d.w. $w_{M'} \notin K$

g.d.w. M' hält nicht für Eingabe $w_{M'}$

Unentscheidbarkeit von K (3)

- M' hält für Eingabe $w_{M'}$ $\iff$ M' hält nicht für Eingabe $w_{M'}$
- Widerspruch!
- Annahme war falsch!
- K ist nicht entscheidbar, sondern unentscheidbar.