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

Entscheidbarkeit und Aufzählbarkeit

Definition

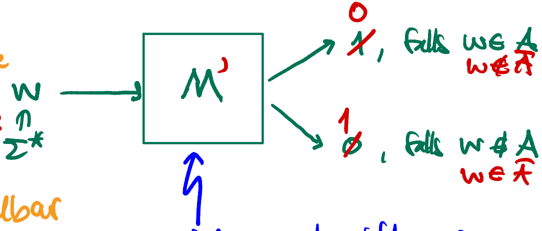
$\bar{A} = \Sigma^* \setminus A$

Eine Sprache $A \subseteq \Sigma^*$ heißt entscheidbar, wenn die Funktion $c_A : \Sigma^* \to \{0, 1\}$ mit

$$c_A(w) := \begin{cases} 1, & \text{falls } w \in A \\ 0, & \text{sonst} \end{cases}$$

berechenbar ist. c_A heißt charakteristische Funktion von A.

schon bewiesen:
Jede Typ-1-Sprache ist entscheidbar.
Typ-0-Sprachen?
- A entscheidbar $\Leftrightarrow$ $\bar{A}$ entscheidbar



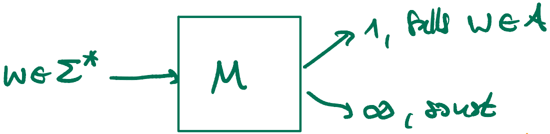
„Entscheidungsalgorithmus für A“

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt semi-entscheidbar, wenn die Funktion
$\chi_A \colon \Sigma^* \to \{0,1\}$ mit

$$\chi_A(w) := \begin{cases} 1, & \text{falls } w \in A \\ \text{undefiniert}, & \text{sonst} \end{cases}$$

berechenbar ist. χ_A heißt semi-charakteristische Fkt. von A.



- Falls A entscheidbar ist, dann ist A semi-entscheidbar.

  Gilt Umkehrung? Kann man Endlosschleifen erkennen?

Satz

Eine Sprache ist genau dann semi-entscheidbar, wenn sie vom Typ 0 ist.

Typ-0 $\equiv$ semi-entscheidbar

Typ-1 $\Longrightarrow$ entscheidbar

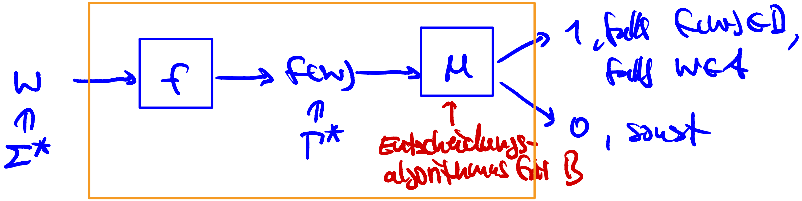
Exkurs: Wie zeigt man (Nicht-) Entscheidbarkeit?
(Nicht-) Semi-Entscheidbarkeit?

Definition

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

$$w \in A \Leftrightarrow f(w) \in B$$

„A kann auf B zurückgeführt werden."
„A ist nicht schwieriger als B."



Entscheidungsalgorithmus für A

Lemma

Ist $A \leq B$ und B entscheidbar, so ist A entscheidbar.

Ist $A \leq B$ und B semi-entscheidbar, so ist A semi-entscheidbar.

„Reduktionsmethode":

Ist $A \leq B$ und A nicht semi-entscheidbar, dann ist B nicht semi-entscheidbar.

Ende des Exkurses

Beobachtung

Sei $A \subseteq \Sigma^*$. Es gilt:

$\nLeftarrow$ (im Januar)

- A ist entscheidbar $\Longrightarrow$ A ist semi-entscheidbar.
- A ist entscheidbar $\Longleftrightarrow$ $\overline{A}$ ist entscheidbar.
- A ist entscheidbar $\Longrightarrow$ A und $\overline{A}$ sind semi-entscheidbar.

$\Longleftarrow$

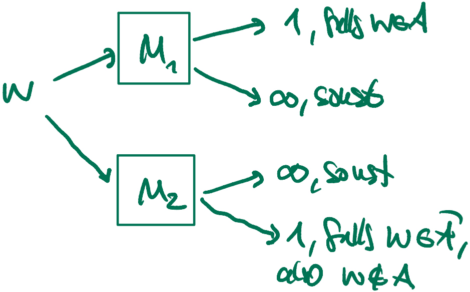
Satz

Sei $A \subseteq \Sigma^*$. Es gilt:

$$A \text{ ist entscheidbar gdw. } A \text{ und } \overline{A} \text{ sind semi-entscheidbar.}$$

Beweis: „$\Rightarrow$": ✓

„$\Leftarrow$": Sei $A \subseteq \Sigma^*$, sei M_1 ein Semi-Entscheidungsalgorithmus für A,
sei M_2 ein Semi-Entscheidungsalgorithmus für $\overline{A}$.



Entscheidungsalgorithmus für A:

Eingabe: $w \in \Sigma^*$

s := 1; {Schrittzähler}
t := 1; {Abbruchbedingung}
while t ≠ 0 do s := s+1;
Simuliere $M_1(w)$ für s Schritte
falls Sim. stoppt, dann Ausgabe „1"; t := 0;
Sim. $M_2(w)$ für s Schritte
falls Sim. stoppt, dann Ausgabe „0"; t := 0

Aus der Mengenlehre:

Menge A ist ~~abzählbar~~ rekursiv-aufzählbar, falls „man die Elemente in A effektiv durchnummerieren kann", formal:

es ex. Funktion $f: \mathbb{N} \to A$ mit
f ist total und berechenbar
und $A = \{ f(0), f(1), f(2), f(3), \ldots \}$

$$\begin{array}{ccccccc} A = \{ & f(0), & f(1), & f(2), & f(3), & \ldots & \} \\ & \updownarrow & \updownarrow & \updownarrow & \updownarrow & & \\ \mathbb{N} = \{ & 0, & 1, & 2, & 3, & \ldots & \} \end{array}$$

also: f ist surjektiv

„In A gibt es nicht mehr Elemente als in $\mathbb{N}$."

Hier: Jede Sprache $A \subseteq \Sigma^*$ ist abzählbar.

Geht das auch algorithmisch?

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt rekursiv-aufzählbar, falls $A = \emptyset$ oder falls es eine totale berechenbare Funktion $f\colon \mathbb{N} \to \Sigma^*$ gibt, sodass

$$A = \{f(0), f(1), f(2), \ldots\}.$$

Wir sagen: f zählt A auf.

Satz

Eine Sprache ist rekursiv-aufzählbar gdw. sie semi-entscheidbar ist.

Beweis: „$\Rightarrow$": Sei $A \subseteq \Sigma^*$, f zähle A auf. $A = \{f(0), f(1), \ldots\}$

Semi-Entscheidungsalgorithmus für A:

Eingabe: $w \in \Sigma^*$:

$t := 1;$

$s := 0;$

while $t \neq 0$ do

  if $f(s) = w$ then $t := 0;$

  $s := s+1$

end

Ausgabe „1" + stopp.

„$\Leftarrow$": Sei $A \subseteq \Sigma^*$. Sei M ein Semi-~~Ent~~scheidungsalgorithmus für A.
Folgender Algorithmus berechnet $f: \mathbb{N} \to \Sigma^*$.

Sei $A \neq \emptyset$, sei w_0 ein beliebiges Wort in A, fest gewählt.

Eingabe: $n \in \mathbb{N}$
Sei $n = \langle x, s \rangle$;
Sei w das x-te Wort in Σ^* (in quasi-lexikographischer Ordnung);
if $M(w)$ hält in $\leq s$ Schritten
then Ausgabe w
else Ausgabe w_0

Dann: $A = \{ f(0), f(1), f(2), \ldots \}$

$\supseteq$: klar

$\subseteq$: Sei $w \in A$.
Sei w das x-te Wort in Σ^*.
Sei s so dass $M(w)$ hält in s Schritten.
Definiere $n = \langle x, s \rangle$. Dann ist $f(n) = w$.

Also: f zählt A auf.

$\square$

<u>Benötigen</u>: Kodierung von <u>Paaren</u> von nat. Zahlen durch <u>eine</u> nat. Zahl,

$c : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ injektiv, surjektiv, berechenbar

z.B. $c(x,y) = \frac{1}{2}(x+y+1)(x+y) + x$

- ist berechenbar, Invertierung ist berechenbar
- ist bijektiv

Schreibweise: $\langle x,y \rangle$ statt $c(x,y)$

Korollar

Eine Sprache A ist entscheidbar gdw. A und $\overline{A}$ rekursiv-aufzählbar sind.