

Einführung in die Theoretische Informatik

Johannes Köbler



Institut für Informatik
Humboldt-Universität zu Berlin

WS 2022/23

Definition

- Eine NTM M **hält bei Eingabe x** (kurz: $M(x) = \downarrow$ oder $M(x) \downarrow$), falls alle Rechnungen von $M(x)$ nach endlich vielen Schritten halten
- Falls $M(x)$ nicht hält, schreiben wir auch kurz $M(x) = \uparrow$ oder $M(x) \uparrow$
- Eine NTM M **entscheidet** eine Eingabe x , falls $M(x)$ hält oder eine Konfiguration mit einem Endzustand erreicht
- Eine Sprache heißt **entscheidbar**, falls sie von einer DTM M akzeptiert wird, die alle Eingaben entscheidet. Die zugehörige Sprachklasse ist

$$\text{REC} = \{L(M) \mid M \text{ ist eine DTM, die alle Eingaben entscheidet}\}$$

- Jede von einer DTM akzeptierte Sprache heißt **semi-entscheidbar**

Bemerkung

- Eine DTM M entscheidet zwar immer alle Eingaben $x \in L(M)$, aber eventuell nicht alle $x \in \overline{L(M)}$. Daher heißt $L(M)$ semi-entscheidbar
- Später werden wir die Gleichheit $\text{RE} = \{L(M) \mid M \text{ ist eine DTM}\}$ zeigen

Definition

- Eine k -DTM $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ **berechnet** eine Funktion $f : \Sigma^* \rightarrow \Gamma^*$, falls M bei jeder Eingabe $x \in \Sigma^*$ in einer Konfiguration $K = (q, u_1, a_1, v_1, \dots, u_k, a_k, v_k)$ mit $u_k = f(x)$ hält (d.h. $K_x \vdash^* K$ und K hat keine Folgekonfiguration)
- Hierfür sagen wir auch, **M gibt bei Eingabe x das Wort $f(x)$ aus** und schreiben **$M(x) = f(x)$**
- f heißt **Turing-berechenbar** (oder einfach **berechenbar**), falls es eine k -DTM M mit $M(x) = f(x)$ für alle $x \in \Sigma^*$ gibt
- Aus historischen Gründen werden berechenbare Funktionen auch **rekursiv** (engl. **recursive**) genannt

Definition

Für eine Sprache $A \subseteq \Sigma^*$ ist die **charakteristische Funktion** $\chi_A : \Sigma^* \rightarrow \{0, 1\}$ wie folgt definiert:

$$\chi_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases}$$

Bemerkung

- In den Übungen wird gezeigt, dass eine Sprache A genau dann entscheidbar ist, wenn χ_A berechenbar (also rekursiv) ist
- Dies erklärt die Bezeichnung REC für die Klasse der entscheidbaren Sprachen
- Dort wird auch gezeigt, dass CSL echt in REC enthalten ist
- Beispiele für interessante semi-entscheidbare Sprachen, die nicht entscheidbar sind, werden wir später kennenlernen
- Somit gilt $\text{REG} \subsetneq \text{DCFL} \subsetneq \text{CFL} \subsetneq \text{DCSL} \subseteq \text{CSL} \subsetneq \text{REC} \subsetneq \text{RE}$

Definition

- Eine **partielle Funktion** hat die Form $f : A \rightarrow B \cup \{\uparrow\}$, wobei $\uparrow \notin B$ ist
- Für $f(x) = \uparrow$ sagen wir auch $f(x)$ ist **undefiniert**
- Der **Definitionsbereich** (engl. **domain**) von f ist

$$\text{dom}(f) = \{x \in A \mid f(x) \neq \uparrow\}$$

- Das **Bild** (engl. **image**) von f ist

$$\text{img}(f) = \{f(x) \mid x \in \text{dom}(f)\}$$

- Eine partielle Funktion $f : A \rightarrow B \cup \{\uparrow\}$ heißt **total**, falls $\text{dom}(f) = A$ ist
- Eine partielle Funktion $f : \Sigma^* \rightarrow \Gamma^* \cup \{\uparrow\}$ heißt **berechenbar**, falls es eine DTM $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ gibt, die f berechnet (d.h. $M(x)$ gibt für alle $x \in \text{dom}(f)$ das Wort $f(x)$ aus und hält im Fall $x \notin \text{dom}(f)$ nicht)
- In diesem Fall gilt also $M(x) = f(x)$ für alle $x \in \Sigma^*$ und wir bezeichnen die Menge $\text{dom}(f) = \{x \in \Sigma^* \mid M(x) \downarrow\}$ auch mit **dom(M)**

Definition (Fortsetzung)

Wir fassen die berechenbaren Funktionen und berechenbaren partiellen Funktionen in folgenden Klassen zusammen:

$$\text{FREC} = \{f \mid f \text{ ist eine berechenbare (totale) Funktion}\}$$

$$\text{FREC}_p = \{f \mid f \text{ ist eine berechenbare partielle Funktion}\}$$

Bemerkung

Offensichtlich gilt $\text{FREC} \not\subseteq \text{FREC}_p$

Beispiel

- Bezeichne x^+ den **lexikografischen Nachfolger** von $x \in \Sigma^*$
- Für $\Sigma = \{0, 1\}$ ergeben sich beispielsweise folgende Werte:

x	ε	0	1	00	01	10	11	000	...
x^+	0	1	00	01	10	11	000	001	...

- Betrachte die auf Σ^* definierten partiellen Funktionen f_1, f_2, f_3, f_4 mit

$$\begin{aligned} f_1(x) &= 0 \\ f_2(x) &= x \\ f_3(x) &= x^+ \end{aligned} \quad \text{und} \quad f_4(x) = \begin{cases} \uparrow, & x = \varepsilon \\ y, & x = y^+ \end{cases}$$

- Da f_1, f_2, f_3, f_4 berechenbar sind, gehören die totalen Funktionen f_1, f_2, f_3 zu FREC und die partielle Funktion f_4 zu FREC_p
- Da f_4 keine totale Funktion ist, gehört f_4 nicht zu FREC



Für eine Sprache $A \subseteq \Sigma^*$ ist $\chi_A : \Sigma^* \rightarrow \{0, 1\}$ mit

$$\chi_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases}$$

die **charakteristische Funktion** von A

Definition

- Die **partielle charakteristische Funktion** von A ist $\hat{\chi}_A : \Sigma^* \rightarrow \{1\} \cup \{\uparrow\}$ mit

$$\hat{\chi}_A(x) = \begin{cases} 1, & x \in A \\ \uparrow, & x \notin A \end{cases}$$

- A heißt **rekursiv aufzählbar**, falls $A = \emptyset$ oder das Bild $\text{img}(f)$ einer (totalen) berechenbaren Funktion $f : \Gamma^* \rightarrow \Sigma^*$ ist

Satz

Folgende Eigenschaften sind für eine Sprache $A \subseteq \Sigma^*$ äquivalent:

- ➊ A ist semi-entscheidbar (d.h. A wird von einer DTM akzeptiert)
- ➋ A wird von einer 1-DTM akzeptiert
- ➌ A ist vom Typ 0
- ➍ A wird von einer NTM akzeptiert
- ➎ A ist rek. aufzählbar (d.h. $A = \emptyset$ oder $A = \text{img}(f)$ für eine Fkt. $f \in \text{FREC}$)
- ➏ $\hat{\chi}_A$ ist berechenbar (d.h. $\hat{\chi}_A \in \text{FREC}_p$)
- ➐ es gibt eine DTM M mit $A = \text{dom}(M)$

Beweis

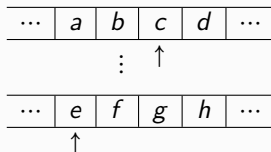
Die Implikationen ➋ $\Rightarrow$ ➌ $\Rightarrow$ ➍ werden in den Übungen gezeigt.

Hier zeigen wir ➊ $\Rightarrow$ ➋ und ➍ $\Rightarrow$ ➎ $\Rightarrow$ ➏ $\Rightarrow$ ➊

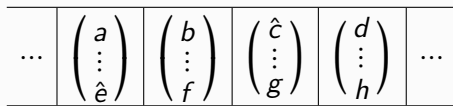
Simulation einer k -DTM durch eine 1-DTM

Beweis von ① $\Rightarrow$ ②: $\{L(M) \mid M \text{ ist eine DTM}\} \subseteq \{L(M) \mid M \text{ ist eine 1-DTM}\}$

- Sei $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ eine k -DTM mit $L(M) = A$
- Wir konstruieren eine 1-DTM $M' = (Z', \Sigma, \Gamma', \delta', z_0, E)$ für A
- M' simuliert M , indem sie jede Konfiguration K von M der Form



durch eine Konfiguration K' folgender Form nachbildet:



Beweis von ① $\Rightarrow$ ②: $\{L(M) \mid M \text{ ist eine DTM}\} \subseteq \{L(M) \mid M \text{ ist eine 1-DTM}\}$

- Hierzu arbeitet M' mit dem Alphabet

$$\Gamma' = \Gamma \cup (\Gamma \cup \hat{\Gamma})^k, \text{ wobei } \hat{\Gamma} = \{\hat{a} \mid a \in \Gamma\} \text{ ist}$$

- Bei Eingabe $x = x_1 \dots x_n$ erzeugt M' zuerst die der Startkonfiguration

$$K_x = (q_0, \varepsilon, x_1, x_2 \dots x_n, \varepsilon, \sqcup, \varepsilon, \dots, \varepsilon, \sqcup, \varepsilon)$$

von M bei Eingabe x entsprechende Konfiguration

$$K'_x = q'_0 \begin{pmatrix} \hat{x}_1 \\ \hat{\sqcup} \\ \vdots \\ \hat{\sqcup} \end{pmatrix} \begin{pmatrix} x_2 \\ \sqcup \\ \vdots \\ \sqcup \end{pmatrix} \dots \begin{pmatrix} x_n \\ \sqcup \\ \vdots \\ \sqcup \end{pmatrix}$$

Beweis von ① $\Rightarrow$ ②: $\{L(M) \mid M \text{ ist eine DTM}\} \subseteq \{L(M) \mid M \text{ ist eine 1-DTM}\}$

- Dann simuliert M' jeweils einen Schritt von M durch folgende Sequenz von Rechenschritten:
 - Zuerst geht M' solange nach rechts, bis sie alle mit $\wedge$ markierten Zeichen (z.B. $\hat{a}_1, \dots, \hat{a}_k$) gefunden hat
 - Diese Zeichen speichert M' in ihrem Zustand
 - Anschließend geht M' wieder nach links und realisiert dabei die durch $\delta(q, a_1, \dots, a_k)$ vorgegebene Anweisung von M
 - Dabei speichert M' den aktuellen Zustand q von M ebenfalls in ihrem Zustand
- Sobald M in einen Endzustand übergeht, wechselt M' ebenfalls in einen Endzustand und hält
- Somit gilt $L(M') = L(M)$



Beweis von ④ $\Rightarrow$ ⑤: $\{L(M) \mid M \text{ ist eine NTM}\} \subseteq \{A \mid A \text{ ist rek. aufzählbar}\}$

- Sei $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ eine k -NTM und sei $A = L(M) \neq \emptyset$
- Sei $\tilde{\Gamma}$ das Alphabet $Z \cup \Gamma \cup \{\#\}$
- Wir kodieren eine Konfiguration $K = (q, u_1, a_1, v_1, \dots, u_k, a_k, v_k)$ durch das Wort

$$\text{code}(K) = \#q\#u_1\#a_1\#v_1\#\dots\#u_k\#a_k\#v_k\#$$

und eine Rechnung $K_0 \vdash \dots \vdash K_t$ durch $\text{code}(K_0) \dots \text{code}(K_t)$

- Dann lassen sich die Wörter von A durch folgende Funktion $f : \tilde{\Gamma}^* \rightarrow \Sigma^*$ aufzählen (dabei ist x_0 ein beliebiges Wort in A):

$$f(w) = \begin{cases} x, & w \text{ kodiert eine akz. Rechnung } K_0 \vdash \dots \vdash K_t \text{ von} \\ & M(x), \text{ d.h. } K_0 = K_x \text{ und } K_t \in E \times (\Gamma^* \times \Gamma \times \Gamma^*)^k \\ x_0, & \text{sonst} \end{cases}$$

- Da f total und berechenbar ist, ist $A = \text{img}(f)$ rekursiv aufzählbar $\square$

Beweis von ⑤ $\Rightarrow$ ⑥: $\{A \mid A \text{ ist rek. aufzählbar}\} \subseteq \{A \mid \hat{\chi}_A \in \text{FREC}_p\}$

- Sei M eine DTM, die eine Fkt. $f : \Gamma^* \rightarrow \Sigma^*$ mit $A = \text{img}(f)$ berechnet
- Dann wird $\hat{\chi}_A$ von der DTM M' berechnet, die bei Eingabe x
 - der Reihe nach für alle $w \in \Gamma^*$ das Wort $f(w)$ berechnet und
 - den Wert 1 ausgibt, sobald $f(w) = x$ ist

□

Beweis von ⑥ $\Rightarrow$ ⑦: $\{A \mid \hat{\chi}_A \in \text{FREC}_p\} \subseteq \{\text{dom}(M) \mid M \text{ ist eine DTM}\}$

- Sei M eine DTM, die $\hat{\chi}_A$ berechnet
- Da $\text{dom}(\hat{\chi}_A) = A$ ist, folgt $A = \text{dom}(M)$

□

Beweis von ⑦ $\Rightarrow$ ①: $\{\text{dom}(M) \mid M \text{ ist eine DTM}\} \subseteq \{L(M) \mid M \text{ ist eine DTM}\}$

- Sei $A = \text{dom}(M)$ für eine DTM M
- Dann gilt $A = L(M')$ für die DTM M' , die M simuliert und genau dann in einen Endzustand übergeht, wenn M hält

□

Satz

Folgende Eigenschaften sind äquivalent:

- ➊ A ist entscheidbar (d.h. A wird von einer DTM akzeptiert, die alle Eingaben entscheidet)
- ➋ die charakteristische Funktion χ_A von A ist berechenbar
- ➌ A wird von einer 1-DTM akzeptiert, die bei allen Eingaben hält
- ➍ A wird von einer NTM akzeptiert, die bei allen Eingaben hält
- ➎ A und $\bar{A}$ sind semi-entscheidbar

Beweis

Die Äquivalenz der Bedingungen ➊ bis ➍ wird in den Übungen gezeigt. Hier zeigen wir nur die Äquivalenz dieser vier Bedingungen zu ➎

Beweis von ① $\Rightarrow$ ⑤: $\text{REC} \subseteq \text{RE} \cap \text{co-RE}$

- Falls A entscheidbar ist, ist χ_A wegen ① $\Rightarrow$ ② berechenbar
- Mit χ_A ist auch $\chi_{\bar{A}}$ berechenbar, d.h. A und $\bar{A}$ sind wegen ② $\Rightarrow$ ① entscheidbar und damit auch semi-entscheidbar

Beweis von ⑤ $\Rightarrow$ ①: $\text{RE} \cap \text{co-RE} \subseteq \text{REC}$

- Sei $A \in \text{RE} \cap \text{co-RE}$ und seien M_A und $M_{\bar{A}}$ DTMs, die $\hat{\chi}_A$ und $\hat{\chi}_{\bar{A}}$ berechnen
- Betrachte die DTM M , die abwechselnd M_A und $M_{\bar{A}}$ für jeweils einen weiteren Rechenschritt simuliert und
 - in einem Endzustand hält, sobald $M_A(x)$ hält, sowie
 - in einem Nichtendzustand hält, sobald $M_{\bar{A}}(x)$ hält
- Da jede Eingabe x entweder in $\text{dom}(\hat{\chi}_A) = A$ oder in $\text{dom}(\hat{\chi}_{\bar{A}}) = \bar{A}$ enthalten ist, hält M bei allen Eingaben
- Da zudem $L(M) = A$ ist, folgt $A \in \text{REC}$

- Um Eigenschaften von TMs algorithmisch untersuchen zu können, müssen wir TMs als Teil der Eingabe kodieren
- Sei $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ eine 1-DTM mit
 - Zustandsmenge $Z = \{q_0, \dots, q_m\}$ (o.B.d.A. sei $E = \{q_m\}$),
 - Eingabealphabet $\Sigma = \{0, 1\}$ und
 - Arbeitsalphabet $\Gamma = \{a_0, \dots, a_l\}$,
wobei wir o.B.d.A. $a_0 = 0$, $a_1 = 1$ und $a_2 = \sqcup$ annehmen
- Dann kodieren wir eine Anweisung $q_i a_j \rightarrow q_{i'} a_{j'} D$ durch das Wort
$$\#bin(i)\#bin(j)\#bin(i')\#bin(j')\#b_D\#$$
- Dabei ist $bin(n)$ die Binärdarstellung von n und

$$b_D = \begin{cases} 0, & D = N \\ 1 & D = L \\ 10, & D = R \end{cases}$$

Kodierung von Turingmaschinen

- M lässt sich nun als ein Wort über dem Alphabet $\{0, 1, \#\}$ kodieren, indem wir die Anweisungen von M in kodierter Form auflisten
- Kodieren wir die Zeichen $0, 1, \#$ binär (z.B. $0 \mapsto 00$, $1 \mapsto 01$, $\# \mapsto 10$), so gelangen wir zu einer Binärkodierung w_M von M
- Die durch die Binärzahl $w_M = b_n \dots b_0$ repräsentierte natürliche Zahl $(w_M)_2 = \sum_{i=0}^n b_i 2^i$ wird auch die **Gödel-Nummer** von M genannt
- M_w ist durch Angabe von w_M bzw. $(w_M)_2$ bis auf die Benennung ihrer Zustände und der Arbeitszeichen in $\Gamma \setminus \{\sqcup, 0, 1\}$ eindeutig bestimmt
- Ganz analog lassen sich auch k -DTMs mit $k > 1$ sowie NTMs binär kodieren
- Umgekehrt können wir jedem Binärstring $w \in \{0, 1\}^*$ eine DTM M_w wie folgt zuordnen (dabei ist M_0 eine beliebige, aber fest gewählte DTM):

$$M_w = \begin{cases} M, & \text{falls eine DTM } M \text{ mit } w_M = w \text{ existiert} \\ M_0, & \text{sonst} \end{cases}$$

Definition

Das **Halteproblem** ist die Sprache

$$H = \left\{ w \# w' \mid \begin{array}{l} w, w' \in \{0, 1\}^* \text{ und} \\ \text{die DTM } M_w \text{ h\"alt} \\ \text{bei Eingabe } w' \end{array} \right\}$$

und das **spezielle Halteproblem** ist

$$K = \left\{ w \in \{0, 1\}^* \mid \begin{array}{l} \text{die DTM } M_w \\ \text{h\"alt bei Eingabe } w \end{array} \right\}$$

χ_H	w_1	w_2	w_3	$\dots$
w_1	0	1	0	$\dots$
w_2	0	1	1	$\dots$
w_3	1	1	0	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

χ_K				
w_1	0			
w_2		1		
w_3			0	
$\vdots$				$\ddots$

Satz

$K, H \in \text{RE}$

Beweis von $K, H \in \text{RE}$

- Sei w_h die Kodierung einer DTM, die sofort hält, und betrachte die Funktionen $f_K : \{0,1\}^* \rightarrow \{0,1\}^*$ und $g_H : \{0,1,\#\}^* \rightarrow \{0,1,\#\}^*$ mit

$$f_K(x) = \begin{cases} w, & x \text{ ist die Binärkodierung einer haltenden Rechnung} \\ & \text{einer DTM } M_w \text{ bei Eingabe } w, \\ w_h, & \text{sonst} \end{cases}$$

und

$$g_H(x) = \begin{cases} w\#w', & x = w\#x' \text{ und } x' \text{ ist die Binärkodierung einer hal-} \\ & \text{tenden Rechnung der DTM } M_w \text{ bei Eingabe } w', \\ w_h\#\varepsilon, & \text{sonst} \end{cases}$$

- Da f_K und g_H in FREC sind und $\text{img}(f_K) = K$ sowie $\text{img}(g_H) = H$ ist, folgt $K, H \in \text{RE}$

□

Satz

$\bar{K} \notin \text{RE}$ und somit $\text{REC} \not\subseteq \text{RE} \neq \text{co-RE}$

Beweisidee

- Sei $B = (b_{ij})$ die durch $b_{ij} = \chi_H(w_i \# w_j) \in \{0, 1\}$ definierte Binärmatrix
- Dann kann keine Zeile $b_{i1} b_{i2} \dots$ von B mit der invertierten Diagonalen $\bar{b}_{11} \bar{b}_{22} \dots$ von B übereinstimmen, da sonst $b_{ii} = \bar{b}_{ii}$ sein müsste

- Da die i -te Zeile von B wegen

$$b_{ij} = \chi_H(w_i \# w_j) = \chi_{\text{dom}(M_{w_i})}(w_j)$$

die Binärsprache $\text{dom}(M_{w_i})$ und die invertierte Diagonale wegen

$$\bar{b}_{ii} = \chi_{\bar{H}(w_i \# w_i)} = \chi_{\bar{K}}(w_i)$$

die Sprache $\bar{K}$ repräsentiert, folgt $\bar{K} \notin \{\text{dom}(M_{w_i}) \mid i \geq 1\}$

- Dies impliziert $\bar{K} \notin \text{RE}$, da die Zeilen von B wegen

$$\{\text{dom}(M_{w_i}) \mid i \geq 1\} = \{A \subseteq \{0, 1\}^* \mid A \in \text{RE}\}$$

genau die Binärsprachen in RE repräsentieren

Beweis von $\bar{K} \notin \text{RE}$

- Angenommen, die Sprache

$$\bar{K} = \{w \in \{0,1\}^* \mid M_w(w) \uparrow\} \quad (*)$$

wäre semi-entscheidbar

- Dann existiert eine DTM M_{w_i} mit

$$\text{dom}(M_{w_i}) = \bar{K} \quad (**)$$

- Dies führt jedoch auf einen Widerspruch:

$$w_i \in \bar{K} \stackrel{(*)}{\Leftrightarrow} M_{w_i}(w_i) \uparrow \Leftrightarrow w_i \notin \text{dom}(M_{w_i}) \stackrel{(**)}{\Leftrightarrow} w_i \notin \bar{K} \quad \textcolor{red}{\text{⚡}}$$

□

χ_H	w_1	w_2	w_3	$\dots$
w_1	0	1	0	$\dots$
w_2	0	1	1	$\dots$
w_3	1	1	0	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	
w_i	1	0	1	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	

Bemerkung

- Die Methode in obigem Beweis wird als **Diagonalisierung** bezeichnet
- Mit dieser Beweistechnik lässt sich auch eine Sprache in $\text{REC} \setminus \text{CSL}$ definieren (siehe Übungen)

Definition

Eine Sprache $A \subseteq \Sigma^*$ heit auf $B \subseteq \Gamma^*$ **reduzierbar** (kurz: $A \leq B$), falls eine berechenbare Funktion $f : \Sigma^* \rightarrow \Gamma^*$ ex., so dass gilt:

$$\forall x \in \Sigma^* : x \in A \Leftrightarrow f(x) \in B$$

Beispiel

- Es gilt $K \leq H$ mittels $f : w \mapsto w\#w$, da fr alle $w \in \{0,1\}^*$ gilt:

$$w \in K \Leftrightarrow M_w(w) \downarrow \Leftrightarrow w\#w \in H$$

- Es gilt sogar $A \leq H$ fr jede Binrsprache $A \in \text{RE}$ mittels $f : w' \mapsto w\#w'$, wobei w die Kodierung einer DTM M_w mit $\text{dom}(M_w) = A$ ist:

$$w' \in A \Leftrightarrow M_w(w') \downarrow \Leftrightarrow w\#w' \in H$$



Definition

- Eine Sprache B heißt **hart** für eine Sprachklasse $\mathcal{C}$ (kurz: **$\mathcal{C}$ -hart** oder **$\mathcal{C}$ -schwer**), falls jede Sprache $A \in \mathcal{C}$ auf B reduzierbar ist:

$$\forall A \in \mathcal{C} : A \leq B$$

- Eine $\mathcal{C}$ -harte Sprache B , die zu $\mathcal{C}$ gehört, heißt **$\mathcal{C}$ -vollständig**

Beispiel: H ist RE-vollständig

- Wir wissen bereits, dass alle Binärsprachen in RE auf H reduzierbar sind
- Ist A eine Sprache über dem Alphabet $\Sigma = \{a_0, \dots, a_{m-1}\}$, so können wir die Zeichen $a_i \in \Sigma$ durch Binärstrings $\text{bin}(a_i)$ der Länge $\lceil \log_2(m) \rceil$ und die Wörter $x = x_1 \dots x_n \in \Sigma^*$ durch $\text{bin}(x) = \text{bin}(x_1) \dots \text{bin}(x_n)$ kodieren
- Dann ist A auf die Binärsprache $\text{bin}(A) = \{\text{bin}(x) \mid x \in A\}$ mittels der Funktion $x \mapsto \text{bin}(x)$ reduzierbar und mit A ist auch $\text{bin}(A)$ in RE
- Somit folgt $A \leq \text{bin}(A) \leq H$, und da $\leq$ transitiv ist (dies wird in den Übungen bewiesen), folgt $A \leq H$ für jede Sprache $A \in \text{RE}$

Absgeschlossenheit von REC unter $\leq$

Definition

Eine Sprachklasse $\mathcal{C}$ heißt **unter $\leq$ abgeschlossen**, wenn für beliebige Sprachen A, B gilt:

$$A \leq B \wedge B \in \mathcal{C} \Rightarrow A \in \mathcal{C}$$

Satz

Die Klasse REC ist unter $\leq$ abgeschlossen

Beweis

- Gelte $A \leq B$ mittels f und sei $B \in \text{REC}$
- Wegen $B \in \text{REC}$ ex. eine DTM M_B , die χ_B berechnet
- Betrachte folgende DTM M_A :
 - M_A berechnet bei Eingabe x zuerst den Wert $f(x)$ und
 - simuliert dann M_B bei Eingabe $f(x)$

Abgeschlossenheit von REC und RE unter $\leq$

Satz

Die Klasse REC ist unter $\leq$ abgeschlossen

Beweis.

- Gelte $A \leq B$ mittels f und sei $B \in \text{REC}$
- Dann ex. eine DTM M_B , die χ_B berechnet
- Betrachte folgende DTM M_A :
 - M_A berechnet bei Eingabe x zuerst den Wert $f(x)$ und
 - simuliert dann M_B bei Eingabe $f(x)$
- Wegen $x \in A \Leftrightarrow f(x) \in B$ ist $\chi_A(x) = \chi_B(f(x))$ und daher folgt
$$M_A(x) = M_B(f(x)) = \chi_B(f(x)) = \chi_A(x)$$
- Also berechnet M_A die Funktion χ_A , d.h. $A \in \text{REC}$ □

Bemerkung

Die Abgeschlossenheit von RE unter $\leq$ folgt analog (siehe Übungen)

Korollar

- $A \leq B \wedge A \notin \text{REC} \Rightarrow B \notin \text{REC}$
- $A \leq B \wedge A \notin \text{RE} \Rightarrow B \notin \text{RE}$

Beweis

Aus der Annahme, dass B entscheidbar (bzw. semi-entscheidbar) ist, folgt wegen $A \leq B$, dass dies auch auf A zutrifft (Widerspruch). $\square$

Bemerkung

Wegen $K \leq H$ überträgt sich somit die Unentscheidbarkeit von K auf H

Korollar

$H \notin \text{REC}$, d.h. H ist unentscheidbar

Das Halteproblem bei leerem Band

Definition

Das **Halteproblem bei leerem Band** ist die Sprache

$$H_0 = \left\{ w \in \{0,1\}^* \mid \begin{array}{l} \text{die DTM } M_w \\ \text{hält bei Eingabe } \varepsilon \end{array} \right\}$$

Satz

H_0 ist RE-vollständig und somit unentscheidbar

Beweis

- $H_0 \in \text{RE}$ folgt wegen $H_0 \leq H \in \text{RE}$ mittels der Reduktionsfunktion $w \mapsto w\#\varepsilon$

χ_H	w_1	w_2	w_3	$\dots$
w_1	0	1	0	$\dots$
w_2	0	1	1	$\dots$
w_3	1	1	0	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

χ_{H_0}	$w_1 (= \varepsilon)$
w_1	0
w_2	0
w_3	1
$\vdots$	$\vdots$

H_0 ist RE-vollständig

Beweis

- $H_0 \in \text{RE}$ folgt wegen $H_0 \leq H \in \text{RE}$ mittels der Funktion $w \mapsto w\#\varepsilon$
- Sei $A \in \text{RE}$ und sei M_A eine DTM mit $\text{dom}(M_A) = A$
- Da jede Sprache $A \in \text{RE}$ auf eine Binärsprache in RE reduzierbar ist, können wir o.B.d.A. $A \subseteq \{0,1\}^*$ annehmen
- Um A auf H_0 zu reduzieren, transformieren wir x in die Kodierung w_x einer DTM M_{w_x} , die ihre Eingabe ignoriert und $M_A(x)$ simuliert
- Dann gilt

$$x \in A \Leftrightarrow M_A(x) \downarrow \Leftrightarrow M_{w_x}(\varepsilon) \downarrow \Leftrightarrow w_x \in H_0$$

- Da die Funktion $x \mapsto w_x$ total und berechenbar ist, folgt $A \leq H_0$ □

Bemerkung. Ebenso folgt, dass folgende Sprachen RE-vollständig sind:

- Für jede Spalte $w' \in \{0,1\}^*$ die Sprache $H_{w'} = \{w \in \{0,1\}^* \mid M_w(w') \downarrow\}$
- $K = \{w \in \{0,1\}^* \mid w\#w \in H\}$

Der Satz von Rice

- Oft möchte man wissen, ob die von einer DTM akz. Sprache (bzw. die von ihr ber. partielle Funktion) eine gewisse Eigenschaft hat oder nicht
- Solche Eigenschaften werden **semantisch** genannt, da sie nur vom Akzeptanz- bzw. Ein/Ausgabeverhalten der geg. DTM abhängen
- Der Satz von Rice besagt, dass so gut wie alle semantischen Eigenschaften von Turingmaschinen unentscheidbar sind
- Zum Beispiel ist das Problem unentscheidbar, ob eine gegebene DTM bei allen Eingaben hält (also eine totale Funktion berechnet)
- Formal lassen sich semantische Eigenschaften durch eine Menge $\mathcal{F}$ von partiellen Funktionen bzw. durch eine Sprachklasse $\mathcal{S}$ beschreiben
- Eine DTM M_w hat dann die Eigenschaft $\mathcal{F}$ (bzw. $\mathcal{S}$), wenn sie eine Funktion in $\mathcal{F}$ berechnet (bzw. eine Sprache in $\mathcal{S}$ akzeptiert)
- Eine semantische Eigenschaft $\mathcal{F}$ (bzw. $\mathcal{S}$) heißt **trivial**, falls entweder alle DTMs oder keine diese Eigenschaft haben

Definition

- Zu einer Klasse $\mathcal{F}$ von partiellen Funktionen definieren wir die Sprache

$$L_{\mathcal{F}} = \{w \in \{0,1\}^* \mid \text{die DTM } M_w \text{ ber. eine partielle Fkt. in } \mathcal{F}\}$$

- Die Eigenschaft $\mathcal{F}$ heißt **trivial**, wenn $L_{\mathcal{F}} = \emptyset$ oder $L_{\mathcal{F}} = \{0,1\}^*$ ist

Der Satz von Rice besagt, dass jede semantische Eigenschaft von DTMs entweder trivial oder unentscheidbar ist

Satz (Satz von Rice)

Für jede nicht triviale Eigenschaft $\mathcal{F}$ ist die Sprache $L_{\mathcal{F}}$ unentscheidbar

Beispiel

- Betrachte die beiden Sprachen

$$L_1 = \{w \in \{0, 1\}^* \mid M_w(0^n) = 0^{n+1} \text{ für alle } n \geq 0\} \text{ und}$$

$$L_2 = \{w \in \{0, 1\}^* \mid M_w(x) = \hat{\chi}_K(x) \text{ für alle } x \in \{0, 1\}^*\}$$

- Dann gilt $L_i = L_{\mathcal{F}_i}$ für die Eigenschaften

$$\mathcal{F}_1 = \{f \in \text{FREC}_p \mid f(0^n) = 0^{n+1} \text{ für alle } n \geq 0\} \text{ und}$$

$$\mathcal{F}_2 = \{f \in \text{FREC}_p \mid f(x) = \hat{\chi}_K(x) \text{ für alle } x \in \{0, 1\}^*\}$$

- $\mathcal{F}_1$ ist nicht trivial, da die partiellen Fkten $f, u : \{0, 1\}^* \rightarrow \{0, 1\}^* \cup \{\uparrow\}$ mit $f(x) = x0$ und $u(x) = \uparrow$ berechenbar sind und $f \in \mathcal{F}_1$ sowie $u \notin \mathcal{F}_1$ ist
- Auch $\mathcal{F}_2$ ist nicht trivial, da $\hat{\chi}_K$ wegen $K \in \text{RE}$ eine berechenbare partielle Funktion in $\mathcal{F}_2$ ist, während $f, u \notin \mathcal{F}_2$ sind
- Also sind die beiden Sprachen L_1 und L_2 nach dem Satz von Rice unentscheidbar

Beispiel (Fortsetzung)

- Dagegen liefert der Satz von Rice nicht die Unentscheidbarkeit folgender Sprachen:

$$L_4 = \{w \in \{0, 1\}^* \mid M_w(x) = \hat{\chi}_{\bar{K}}(x) \text{ für alle } x \in \{0, 1\}^*\} \text{ und}$$
$$L_5 = \{w \in \{0, 1\}^* \mid M_w(0^n) \text{ hält für alle } n \geq 0 \text{ nach } n \text{ Schritten}\}$$

- Zwar ist $L_4 = L_{\mathcal{F}_4}$ für die Eigenschaft $\mathcal{F}_4 = \{\hat{\chi}_{\bar{K}}\}$, d.h. L_4 beschreibt eine semantische Eigenschaft von DTMs, da sie nur von deren Ein-/Ausgabeverhalten abhängt
- Da aber $\bar{K} \notin \text{RE}$ und somit $\hat{\chi}_{\bar{K}}$ nicht berechenbar, also $L_4 = L_{\mathcal{F}_4} = \emptyset$ ist, ist $\mathcal{F}_4$ eine triviale Eigenschaft
- Dagegen hängt die Frage, ob eine DTM M_w die durch L_5 beschriebene Eigenschaft hat, nicht nur vom Ein-/Ausgabe-verhalten von M_w ab, sondern auch von ihrer Laufzeit
- Daher gilt für keine Funktionenklasse $\mathcal{F}$ die Gleichheit $L_5 = L_{\mathcal{F}}$



Satz (Satz von Rice)

Für jede nicht triviale Eigenschaft $\mathcal{F}$ ist die Sprache $L_{\mathcal{F}}$ unentscheidbar

Beweisidee

- Die Idee besteht darin, H_0 auf $L_{\mathcal{F}}$ (oder auf $\bar{L}_{\mathcal{F}}$) zu reduzieren, indem wir für eine gegebene DTM M_w eine DTM $M_{w'}$ konstruieren mit
$$w \in H_0 \Leftrightarrow M_{w'} \text{ berechnet (k)eine partielle Funktion in } \mathcal{F}$$
- Hierzu lassen wir $M_{w'}$ bei Eingabe x zunächst einmal die DTM M_w bei Eingabe ε simulieren
- Falls $w \notin H_0$ ist, berechnet $M_{w'}$ also die überall undefinierte Funktion $u(x) = \uparrow$ für alle $x \in \{0, 1\}^*$
- Damit die Reduktion gelingt, müssen wir nur noch dafür sorgen, dass $M_{w'}$ im Fall $w \in H_0$ eine partielle Funktion f berechnet, die sich bzgl. der Eigenschaft $\mathcal{F}$ von u unterscheidet, d.h. $f \in \mathcal{F} \Leftrightarrow u \notin \mathcal{F}$
- Da $\mathcal{F}$ nicht trivial ist, ex. eine DTM M , die ein solches f berechnet

Satz (Satz von Rice)

Für jede nicht triviale Eigenschaft $\mathcal{F}$ ist die Sprache $L_{\mathcal{F}}$ unentscheidbar

Beweis

- Sei M eine DTM, die eine Funktion f mit $f \in \mathcal{F} \Leftrightarrow u \notin \mathcal{F}$ berechnet
- Betrachte die Reduktionsfunktion
$$h(w) = w', \text{ wobei } w' \text{ die Kodierung einer DTM ist, die bei Eingabe } x \text{ zunächst die DTM } M_w(\varepsilon) \text{ simuliert und im Fall, dass } M_w(\varepsilon) \text{ hält, mit der Simulation von } M(x) \text{ fortfährt}$$
- Dann ist $h : w \mapsto w'$ eine totale berechenbare Funktion und es gilt
$$\begin{aligned} w \in H_0 &\Rightarrow M_{w'} \text{ berechnet } f \\ w \notin H_0 &\Rightarrow M_{w'} \text{ berechnet } u \end{aligned}$$
- Dies zeigt, dass h die Sprache H_0 auf $L_{\mathcal{F}}$ oder auf $\bar{L}_{\mathcal{F}}$ reduziert
- Daher überträgt sich die Unentscheidbarkeit von H_0 auf $L_{\mathcal{F}}$ □

Der Satz von Rice gilt auch für Eigenschaften, die das Akzeptanzverhalten einer gegebenen Turingmaschine betreffen

Satz (Satz von Rice für Spracheigenschaften)

Für eine beliebige Sprachklasse $\mathcal{S}$ sei

$$L_{\mathcal{S}} = \{w \in \{0, 1\}^* \mid L(M_w) \in \mathcal{S}\}$$

Dann ist $L_{\mathcal{S}}$ unentscheidbar, außer wenn $L_{\mathcal{S}} = \emptyset$ oder $L_{\mathcal{S}} = \{0, 1\}^*$ ist

Beweis

Siehe Übungen