Einführung in die Theoretische Informatik

Johannes Köbler

Institut für Informatik
Humboldt-Universität zu Berlin

WS 2018/19

Entscheidbare und semi-entscheidbare Sprachen

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 DTM 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

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

- 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 sehen, dass $\mathsf{RE} = \{L(M) \mid M \text{ ist eine DTM}\}$ ist

Berechnung von Funktionen

Definition

- Eine k-DTM $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ berechnet eine Funktion $f : \Sigma^* \to \Gamma^*$, falls M bei jeder Eingabe $x \in \Sigma^*$ in einer Konfiguration

  $$K = (q, u_1, a_1, v_1, \ldots, u_k, a_k, v_k) \text{ mit } u_k = f(x)$$

  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

Berechenbarkeit von charakteristischen Funktionen

Definition

Für eine Sprache $A \subseteq \Sigma^*$ ist die charakteristische Funktion $\chi_A : \Sigma^* \to \{0,1\}$ 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 noch kennenlernen
- Somit gilt REG $\subsetneq$ DCFL $\subsetneq$ CFL $\subsetneq$ DCSL $\subseteq$ CSL $\subsetneq$ REC $\subsetneq$ RE

Berechenbarkeit von partiellen Funktionen

Definition

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

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

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

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

- Eine partielle Funktion $f : A \to B \cup \{\uparrow\}$ heißt total, falls $dom(f) = A$ ist
- Eine partielle Funktion $f : \Sigma^* \to \Gamma^* \cup \{\uparrow\}$ heißt berechenbar, falls es eine DTM M mit $M(x) = f(x)$ für alle $x \in \Sigma^*$ gibt (d.h. $M(x)$ gibt für alle $x \in dom(f)$ das Wort $f(x)$ aus und hält im Fall $x \notin dom(f)$ nicht)

- Jede DTM $M = (Z, \Sigma, \Gamma, \delta, q_0, E)$ ber. eine part. Fkt. $f : \Sigma^* \to \Gamma^* \cup \{\uparrow\}$
- Die Menge $dom(f) = \{x \in \Sigma^* \mid M(x)\downarrow\}$ bezeichnen wir mit $dom(M)$

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

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

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

Dann gilt $\text{FREC} \subsetneq \text{FREC}_p$.

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	$\cdots$
x^+	0	1	00	01	10	11	000	001	$\cdots$

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

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

- 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 ◁

Partielle charakteristische Funktionen

Definition

Sei $A \subseteq \Sigma^*$ eine Sprache.

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

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

- A heißt rekursiv aufzählbar, falls $A = \varnothing$ oder das Bild $img(f)$ einer (totalen) berechenbaren Funktion $f : \Gamma^* \to \Sigma^*$ ist

Charakterisierung der rekursiv aufzählbaren Sprachen

Satz

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

1. A ist semi-entscheidbar (d.h. A wird von einer DTM akzeptiert)
2. A wird von einer 1-DTM akzeptiert
3. A ist vom Typ 0
4. A wird von einer NTM akzeptiert
5. A ist rek. aufzählbar (d.h. $A = \varnothing$ oder $A = img(f)$ für eine Fkt. $f \in \mathsf{FREC}$)
6. $\hat{\chi}_A$ ist berechenbar (d.h. $\hat{\chi}_A \in \mathsf{FREC}_p$)
7. es gibt eine DTM M mit $A = dom(M)$

Beweis

Die Implikationen ❷ $\Rightarrow$ ❸ $\Rightarrow$ ❹ werden in den Übungen gezeigt.
Hier zeigen wir ❶ $\Rightarrow$ ❷ und ❹ $\Rightarrow$ ❺ $\Rightarrow$ ❻ $\Rightarrow$ ❼ $\Rightarrow$ ❶.

Simulation einer k-DTM durch eine 1-DTM

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

- 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

$$\begin{array}{c|c|c|c|c|c}
\hline
\cdots & a & b & c & d & \cdots \\
\hline
\multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{\vdots} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} \\
\hline
\cdots & e & f & g & h & \cdots \\
\hline
\multicolumn{1}{c}{} & \multicolumn{1}{c}{\uparrow} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{}
\end{array}$$

durch eine Konfiguration K' folgender Form nachbildet:

$$\begin{array}{c|c|c|c|c|c}
\hline
\cdots & \begin{pmatrix} a \\ \vdots \\ \hat{e} \end{pmatrix} & \begin{pmatrix} b \\ \vdots \\ f \end{pmatrix} & \begin{pmatrix} \hat{c} \\ \vdots \\ g \end{pmatrix} & \begin{pmatrix} d \\ \vdots \\ h \end{pmatrix} & \cdots \\
\hline
\end{array}$$

Simulation einer k-DTM durch eine 1-DTM

Beweis von ❶ ⇒ ❷: $\{L(M) \mid M \text{ ist eine DTM}\} \subseteq \{L(M) \mid M \text{ ist eine 1-DTM}\}$

- 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

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

Simulation einer k-DTM durch eine 1-DTM

Beweis von ❶ $\Rightarrow$ ❷: $\{L(M) | M \text{ ist eine DTM}\} \subseteq \{L(M) | M \text{ ist eine 1-DTM}\}$

- Dann simuliert M' jeweils einen Schritt von M durch folgende Sequenz von Rechenschritten:
  - Zuerst geht M' solange nach rechts, bis sie alle mit ˆ 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 $L(M') = L(M)$ $\square$

Charakterisierung der rekursiv aufzählbaren Sprachen

Beweis von ❹ ⇒ ❺: $\{L(M) \mid M \text{ ist eine NTM}\} \subseteq \{A \mid A \text{ ist rek. aufzählbar}\}$

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

  $$code(K) = \#q\#u_1\#a_1\#v_1\#\ldots\#u_k\#a_k\#v_k\#$$

  und eine Rechnung $K_0 \vdash \cdots \vdash K_t$ durch $code(K_0)\ldots code(K_t)$
- Dann lassen sich die Wörter von A durch folgende Funktion $f : \tilde{\Gamma}^* \to \Sigma^*$ aufzählen (dabei ist x_0 ein beliebiges Wort in A):

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

- Da f berechenbar ist, ist $A = img(f)$ rekursiv aufzählbar □

Charakterisierung der rekursiv aufzählbaren Sprachen

Beweis von ❺ ⇒ ❻: $\{A \mid A \text{ ist rek. aufzählbar}\} \subseteq \{A \mid \hat{\chi}_A \in \mathrm{FREC}_p\}$

- Sei M eine DTM, die eine Fkt. $f : \Gamma^* \to \Sigma^*$ mit $A = img(f)$ 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 ❻ ⇒ ❼: $\{A \mid \hat{\chi}_A \in \mathrm{FREC}_p\} \subseteq \{dom(M) \mid M \text{ ist eine DTM}\}$

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

Beweis von ❼ ⇒ ❶: $\{dom(M) \mid M \text{ ist eine DTM}\} \subseteq \{L(M) \mid M \text{ ist eine DTM}\}$

- Sei $A = dom(M)$ 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 □

Charakterisierung der entscheidbaren Sprachen

Satz

Folgende Eigenschaften sind äquivalent:

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

Beweis

Die Äquivalenz der Bedingungen 1 bis 4 wird in den Übungen gezeigt. Hier zeigen wir nur die Äquivalenz dieser vier Bedingungen zu 5.