Theoretische Informatik

Lösungsvorschlag zu Übungsblatt 5

Albert-Ludwigs-Universität Freiburg

Sommer-Semester 2022

Aufgabe 5.1 - Entscheidungsalgorithmus

Geben Sie einen Algorithmus zur Berechnung der folgender Funktion an. Reduzieren Sie das Problem dazu auf eines der in der Vorlesung vorgestellten Probleme.

Problem: Abgeschlossenheit der Sprache eines regulären Ausdrucks unter Spiegelung
Eingabe: ein regulärer Ausdruck x über einem Alphabet Σ
Ausgabe: *ja* wenn für alle $w \in L(x)$ auch $w^R \in L(x)$ und *nein* sonst

Idee: Wir reduzieren das Problem wie folgt auf das Inklusionsproblem für zwei NFAs:

1. Wir erzeugen einen regulären Ausdruck x^R, so dass $L(x^R) = \{w^R \mid w \in L(x)\}$.
2. Wir generieren zwei NFAs $\mathcal{M}$ aus x und $\mathcal{M}^R$ aus x^R mittels des Verfahrens aus dem Satz von Kleene (und ggf. der Eliminierung von ϵ-Übergängen).
3. Wir überprüfen, ob $L(\mathcal{M}^R) \subseteq L(\mathcal{M})$. Ist dies der Fall geben wir *ja* zurück, sonst *nein*.

Dies funkioniert, da $\{w^R \mid w \in L(x)\} \subseteq L(x)$ gdw. $w^R \in L(x)$ für alle $w \in L(x)$.

Aufgabe 5.1 - Entscheidungsalgorithmus

Wie sieht der reguläre Ausdruck x^R aus, der genau die gespiegelten Wörter matcht?

$$
\begin{aligned}
\emptyset^R &= \emptyset && \\
\epsilon^R &= \epsilon && \\
a^R &= a && a \in \Sigma \\
(uv)^R &= v^R u^R && u, v \text{ reguläre Ausdrücke über } \Sigma \\
(u|v)^R &= u^R | v^R && u, v \text{ reguläre Ausdrücke über } \Sigma \\
(u^*)^R &= (u^R)^* && u \text{ regulärer Ausdruck über } \Sigma
\end{aligned}
$$

Man kann $L(x^R) = \{w^R \mid w \in L(x)\}$ mittels einfacher struktureller Induktion zeigen.

Aufgabe 5.1 - Entscheidungsalgorithmus

Beweis, dass $L(x^R) = \{w^R \mid w \in L(x)\}$, mittels struktureller Induktion:

- Induktionsanfang:
  - $x = \emptyset$: $L(\emptyset^R) = L(\emptyset) = \emptyset = \{w^R \mid w \in L(\emptyset)\}$
  - $x = \epsilon$: $L(\epsilon^R) = L(\epsilon) = \{\epsilon\} = \{w^R \mid w \in \{\epsilon\}\}$
  - $x = a$ für ein $a \in \Sigma$: $L(a^R) = L(a) = \{a\} = \{w^R \mid w \in \{a\}\}$
- Induktionshypothese: $L(\alpha^R) = \{w^R \mid w \in L(\alpha)\}$ für alle kleineren Reg. Ausdrücke α, aus denen x besteht.
- Induktionsschritt:
  - $x = \alpha\beta$, die anderen Fälle gehen analog

$$\begin{aligned} L((\alpha\beta)^R) = L(\beta^R\alpha^R) = L(\beta^R)L(\alpha^R) &\overset{IV}{=} \{w^R \mid w \in L(\beta)\} \cdot \{w^R \mid w \in L(\alpha)\} \\ &= \{v^R w^R \mid v \in L(\beta), w \in L(\alpha)\} \\ &= \{(wv)^R \mid w \in L(\alpha), v \in L(\beta)\} \\ &= \{w^R \mid w \in L(\alpha\beta)\} \end{aligned}$$

Aufgabe 5.2 - Pumping-Lemma

Zeigen Sie mit Hilfe des Pumping-Lemmas, dass folgende Sprachen nicht regulär sind:

(a) $L_1 = \{a^{k!} \mid k \geq 0\}$

(b) $L_2 = \{a^k b^j \mid k \geq j \geq 0\}$

Idee: Wir nehmen an, das die Sprache regulär ist und führen dies zum Widerspruch. Dazu benutzen wir die Eigenschaft regulärer Sprachen, dass ein $n \geq 0$ existiert, so dass jedes Wort w der Sprache mit Länge $|w| \geq n$ eine Pumping-Zerlegung hat.

Zur Erinnerung: Eine Pumping-Zerlegung $w = uvw$ erfüllt die Eigenschaft, dass für alle $m \geq 0$ das aufgepumpte Wort $uv^m w$ ebenfalls Teil der Sprache ist. Dabei muss $|v| \geq 1$ und $|uv| \leq n$ sein.

L regulär

$\Rightarrow \exists n \in \mathbb{N} : \forall z \in L, |z| \geq n \Rightarrow \exists u, v, w \in \Sigma^* : (z = uvw \land |v| \geq 1 \land |uv| \leq n \land (\forall m \geq 0 : uv^m w \in L))$

Aufgabe 5.2 (a) - $L_1 = \{a^{k!} \mid k \geq 0\}$

- Wir nehmen an, dass $L_1 = \{a^{k!} \mid k \geq 0\}$ regulär ist. Wegen des Pumping-Lemmas gibt es dann ein $n \geq 0$, so dass jedes $w \in L_1$ mit $|w| \geq n$ eine Pumping-Zerlegung hat.
- Wir betrachten das Wort $a^{k!}$, wobei $k = n+1 \geq 1$ ist. Nach Annahme hat dieses Wort eine Zerlegung $a^{k!} = uvw$ mit $|v| \geq 1$, $|uv| \leq n$, und $uv^m w \in L_1$ für jedes $m \in \mathbb{N}$.
- Dann gilt $u = a^i$, $v = a^j$ und $w = a^{k!-i-j}$ für ein $i \in \mathbb{N}_0$ und $j \in \mathbb{N}_0$.
- Wegen $|uv| \leq n$ gilt dann $i + j \leq n < k$.
- Wegen $|v| \geq 1$ gilt zudem $1 \leq j < k$.
- Wir betrachten nun das Wort $uv^2w = a^{i+2j+(k!-i-j)} = a^{k!+j}$.
- Wegen $1 \leq j < k$ gilt $k! < k! + j$ und außerdem
  $k! + j < k! + k = k \cdot ((k-1)! + 1) \leq k \cdot (2 \cdot (k-1)!) \leq (k+1) \cdot k \cdot (k-1)! = (k+1)!$.
  (Für diese Abschätzung benötigen wir $(k-1)! \geq 1$ und $k + 1 \geq 2$.)
- Da $m!$ streng monoton steigt (für $m \geq 1$), existiert kein $m \in \mathbb{N}$ mit $k! < m! < (k+1)!$.
- Also existiert auch kein $m \in \mathbb{N}$ mit $m! = k! + j$, daher gilt $a^{k!+j} \notin L_1$
  $\rightarrow$ Widerspruch zu $uv^2w \in L_1$.

Aufgabe 5.2 (b) - $L_2 = \{a^k b^j \mid k \geq j \geq 0\}$

- Wir nehmen an, dass $L_2 = \{a^k b^j \mid k \geq j \geq 0\}$ regulär ist. Wegen des Pumping-Lemmas gibt es dann ein $n \geq 0$, so dass jedes $w \in L_2$ mit $|w| \geq n$ eine Pumping-Zerlegung hat.
- Wir betrachten das Wort $a^n b^n$. Nach Annahme hat dieses Wort eine Zerlegung $a^n b^n = uvw$ mit $|v| \geq 1$, $|uv| \leq n$, und $uv^m w \in L_2$ für jedes $m \in \mathbb{N}$.
- Wegen $|uv| \leq n$ gilt dann $uv = a^i$ sowie $w = a^{n-i} b^n$ für ein i mit $1 \leq i \leq n$.
- Wegen $|v| \geq 1$ gilt zudem $v = a^j$ für ein j mit $1 \leq j \leq i \leq n$.
- Wir betrachten nun das Wort $uv^0 w = a^{(i-j)+(n-i)} b^n = a^{n-j} b^n$.
- Wegen $1 \leq j \leq n$ gilt $n - j < n$, woraus gilt $a^{n-j} b^n \notin L_2$
  $\rightarrow$ Widerspruch zu $uv^0 w \in L_2$.

Aufgabe 5.3 - Erzeugung CFG in CNF

Wir betrachten die folgende Grammatik G mit Variablen $V = \{S, X, Y\}$, Alphabet $\Sigma = \{a, b\}$, Startsymbol S und den folgenden Produktionsregeln P:

$$S \rightarrow XbX$$
$$X \rightarrow XX \mid Y \mid \epsilon$$
$$Y \rightarrow aXb \mid bXa \mid b$$

(a) Beschreiben Sie die Sprache, die von G erzeugt wird in Worten.

- Die Sprache ist die Menge aller Wörter über Σ, in denen b echt öfter als a vorkommt.
- X erzeugt dabei alle Wörter über Σ, in denen b mindestens so oft wie a vorkommt.

(b) Transformieren Sie die Grammatik mit dem in der Vorlesung angegebenen Verfahren in Chomsky-Normalform.

Aufgabe 5.3 (b) - Transformation in CNF (Schritt 1)

Zur Erinnerung: Die Produktionsregeln von G sind gegeben als

$$S \rightarrow XbX$$
$$X \rightarrow XX \mid Y \mid \epsilon$$
$$Y \rightarrow aXb \mid bXa \mid b$$

Schritt 1: Umwandlung in ϵ-freie CFG

Wir haben $V_\epsilon = \{X\}$. Da $S \notin V_\epsilon$ brauchen wir kein neues Startsymbol.

Wir erhalten die Grammatik $G' = (V, \Sigma, P', S)$ mit den folgenden Produktionsregeln P':

$$S \rightarrow XbX \mid b \mid bX \mid Xb$$
$$X \rightarrow XX \mid Y \mid X$$
$$Y \rightarrow aXb \mid bXa \mid b \mid ab \mid ba$$

Aufgabe 5.3 (b) - Transformation in CNF (Schritt 2)

Zur Erinnerung: Die Produktionsregeln von G' sind gegeben als

$$S \to XbX \mid b \mid bX \mid Xb$$
$$X \to XX \mid Y \mid X$$
$$Y \to aXb \mid bXa \mid b \mid ab \mid ba$$

Schritt 2: Eliminierung von Kettenregeln

Wir haben $E(S) = \{S\}$, $E(X) = \{X, Y\}$ und $E(Y) = \{Y\}$.

Wir erhalten $G'' = (V, \Sigma, P'', S)$ mit den folgenden Produktionsregeln P'':

$$S \to XbX \mid b \mid bX \mid Xb$$
$$X \to XX \mid aXb \mid bXa \mid b \mid ab \mid ba$$
$$Y \to aXb \mid bXa \mid b \mid ab \mid ba$$

Aufgabe 5.3 (b) - Transformation in CNF (Schritt 3)

Zur Erinnerung: Die Produktionsregeln von G'' sind gegeben als

$$S \rightarrow XbX \mid b \mid bX \mid Xb$$
$$X \rightarrow XX \mid aXb \mid bXa \mid b \mid ab \mid ba$$
$$Y \rightarrow aXb \mid bXa \mid b \mid ab \mid ba$$

Schritt 3: Extraktion von Regeln der Form $A \rightarrow c$

Wir führen die neuen Variablen V_a und V_b ein mit $V_a \rightarrow a$ und $V_b \rightarrow b$.

Wir erhalten $G''' = (V', \Sigma, P''', S)$ mit $V' = \{S, X, Y, V_a, V_b\}$ und den Produktionsregeln P''':

$$S \rightarrow XV_bX \mid b \mid V_bX \mid XV_b$$
$$X \rightarrow XX \mid V_aXV_b \mid V_bXV_a \mid b \mid V_aV_b \mid V_bV_a$$
$$Y \rightarrow V_aXV_b \mid V_bXV_a \mid b \mid V_aV_b \mid V_bV_a$$
$$V_a \rightarrow a \quad V_b \rightarrow b$$

Aufgabe 5.3 (b) - Transformation in CNF (Schritt 4)

Zur Erinnerung: Die Produktionsregeln von G''' sind gegeben als

$$
\begin{aligned}
S &\rightarrow XV_bX \mid b \mid V_bX \mid XV_b \qquad V_a \rightarrow a \quad V_b \rightarrow b \\
X &\rightarrow XX \mid V_aXV_b \mid V_bXV_a \mid b \mid V_aV_b \mid V_bV_a \\
Y &\rightarrow V_aXV_b \mid V_bXV_a \mid b \mid V_aV_b \mid V_bV_a
\end{aligned}
$$

Schritt 4: Zerlegung von Regeln der Form $A \rightarrow B_1 \ldots B_n$

Wir erhalten $G'''' = (V'', \Sigma, P'''', S)$ mit $V'' = \{S, X, Y, C_1, D_1, E_1, F_1, G_1, V_a, V_b\}$ und P'''':

$$
\begin{aligned}
S &\rightarrow XC_1 \mid b \mid V_bX \mid XV_b \qquad V_a \rightarrow a \quad V_b \rightarrow b \\
X &\rightarrow XX \mid V_aD_1 \mid V_bE_1 \mid b \mid V_aV_b \mid V_bV_a \\
Y &\rightarrow V_aF_1 \mid V_bG_1 \mid b \mid V_aV_b \mid V_bV_a \\
C_1 &\rightarrow V_bX \quad D_1 \rightarrow XV_b \quad E_1 \rightarrow XV_a \quad F_1 \rightarrow XV_b \quad G_1 \rightarrow XV_a
\end{aligned}
$$

Die Grammatik G'''' ist äquivalent zu G und in Chomsky-Normalform.

Aufgabe 5.3 (b) - Eine Anmerkung

Anmerkung: Bei der händischen Umwandlung in CNF ist es auch erlaubt, in den Zwischenschritten Vereinfachungen durchzuführen, sofern diese klar sind.

Beispiel: Nach Schritt 2 hätte man bereits sehen können, dass man vom Startzustand das Symbol Y (auch als Teilwort) nicht erzeugen kann. Damit kann man das Symbol und die dazugehörigen Produktionsregeln weglassen.

Beispiel: In Schritt 4 hätte zur Aufteilung der Regeln $S \to XV_bX$ und $X \to V_bXV_a$ die Einführung einer einzigen Regel $C_1 \to V_bX$ gereicht.

Mit beiden Vereinfachungen erhalten wir folgende Grammatik:

$$S \to XC_1 \mid b \mid V_bX \mid XV_b \qquad V_a \to a \quad V_b \to b$$
$$X \to XX \mid V_aD_1 \mid C_1V_a \mid b \mid V_aV_b \mid V_bV_a$$
$$C_1 \to V_bX \quad D_1 \to XV_b$$

Aufgabe 5.4 - Ableitungen

Überlegen Sie sich eine Typ-1-Grammatik G und ein Wort $w \in L(G)$, so dass eine Linksableitung, jedoch keine Rechtsableitung vom Startsymbol zu w existiert.

Betrachten wir die Grammatik mit der Variablenmenge $\{S, A, B, X, Y\}$, dem Startsymbol S, dem Alphabet $\{y, d, o, n, e\}$ und den folgenden Produktionsregeln:

$$S \to XY \qquad X \to AB \qquad Y \to y \qquad ABY \to done$$

Die Grammatik ist vom Typ 1, da $|w| \leq |v|$ für alle Produktionsregeln $w \to v$.

Einzige Linksableitung: $S \Rightarrow XY \Rightarrow ABY \Rightarrow done$.

Will man Rechtsableitung bilden, kommt man nach $S \Rightarrow XY \Rightarrow Xy \Rightarrow ABy$ nicht weiter.