Automaten und Formale Sprachen
5. Vorlesung

Prof. Dr. Dietrich Kuske

FG Automaten und Logik, TU Ilmenau

Wintersemester 2023/24

Reguläre Ausdrücke

reguläre Ausdrücke erlauben es, die regulären Sprachen kompakt in „Textform“ zu beschreiben

Definition

Die Menge $\mathrm{Reg}(\Sigma)$ der regulären Ausdrücke über dem Alphabet Σ ist die kleinste Menge mit folgenden Eigenschaften:

- $\varnothing \in \mathrm{Reg}(\Sigma)$, $\lambda \in \mathrm{Reg}(\Sigma)$, $\Sigma \subseteq \mathrm{Reg}(\Sigma)$.
- Wenn $\alpha, \beta \in \mathrm{Reg}(\Sigma)$, dann auch $(\alpha \cdot \beta), (\alpha + \beta), (\alpha^*) \in \mathrm{Reg}(\Sigma)$.

Bemerkungen:

- für $(\alpha \cdot \beta)$ schreibt man oft $(\alpha\,\beta)$
- für $(\alpha + \beta)$ schreibt man auch $(\alpha \mid \beta)$

Beispiel

$\Sigma = \{a, b, c, d\}$:

$$\lambda \quad ((ab)b) \quad ((a+d)a)$$
$$\Big(\big((ab)a\big) + \big(((ba)b)\big)^*\Big) \quad \Big(\big((ab)((a+b)^*)\big)(ba)\Big)$$

$\Sigma = \{0, 1\}$: $\Big(1\big((0+1)^*\big)\Big) \quad \Big(\big((0^*)(1^*)\big)(0^*)\Big)$

Beobachtung: Jeder reguläre Ausdruck über Σ ist ein Wort über dem Alphabet $\Sigma \cup \{\varnothing, \lambda, \cdot, +, ^*, (,)\}$.

Nach der Festlegung der Syntax regulärer Ausdrücke müssen wir auch deren Bedeutung (= Semantik) festlegen.
Dabei betrachten wir einen regulären Ausdruck α als ein „Wortschema“, $L(\alpha)$ soll die Menge der Wörter sein, die zum „Wortschema“ „passen“.

Definition

Für einen regulären Ausdruck $\alpha \in \mathrm{Reg}(\Sigma)$ ist die Sprache $L(\alpha) \subseteq \Sigma^*$ induktiv definiert:

$$L(\alpha) = \begin{cases} \emptyset & \text{falls } \alpha = \varnothing \\ \{\varepsilon\} & \text{falls } \alpha = \lambda \\ \{a\} & \text{falls } \alpha = a \in \Sigma \\ L(\beta) \cup L(\gamma) & \text{falls } \alpha = (\beta + \gamma) \\ L(\beta)L(\gamma) & \text{falls } \alpha = (\beta \cdot \gamma) \\ \big(L(\beta)\big)^* & \text{falls } \alpha = (\beta^*) \end{cases}$$

Beispiel (vgl. Folie 5.3)

$\Sigma = \{a, b, c, d\}$:

$$
\begin{aligned}
L(\lambda) &= \{\varepsilon\} \\
L\Big(\big((ab)b\big)\Big) &= \{abb\} \\
L\Big(\big((a+d)a\big)\Big) &= \{aa, da\} \\
L\Big(\Big(\big((ab)a\big) + \big(((ba)b)\big)^*\Big)\Big) &= \{aba\} \cup \{bab\}^* \\
L\Big(\Big(\big((ab)((a+b)^*)\big)(ba)\Big)\Big) &= \{ab\} \cdot \{a, b\}^* \cdot \{ba\}
\end{aligned}
$$

$\Sigma = \{0, 1\}$:

$$
\begin{aligned}
L\Big(\big(1((0+1)^*)\big)\Big) &= \{1\} \cdot \{0, 1\}^* \\
L\Big(\big(((0^*)(1^*))(0^*)\big)\Big) &= \{0^i 1^j 0^k \mid i, j, k \in \mathbb{N}\}
\end{aligned}
$$

Klammern sparen:

- Äußere Klammern weglassen: $(0+1)^*$ statt $\big((0+1)^*\big)$.
- Da $(L_1L_2)L_3 = L_1(L_2L_3)$ und $(L_1 \cup L_2) \cup L_3 = L_1 \cup (L_2 \cup L_3)$, läßt man Klammern bei Operatoren auf gleicher Ebene weg:
  $(0+1+2+3)$ statt $\big((0+1)+(2+3)\big)$ oder $\Big(0+\big(1+(2+3)\big)\Big)$
- **Präferenzregeln:**
  * bindet stärker als $\cdot$
  $\cdot$ bindet stärker als $+$.

Damit erhalten die Beispiel-Ausdrücke die lesbarere Form:

λ	abb	$(a+d)a$
$aba+(bab)^*$	$ab(a+b)^*ba$	
$1(0+1)^*$	$0^*1^*0^*$	

Beispiel

$\Sigma = \{a, b\}$.

- $L\big(a(a+b)^*bb\big)$ ist die Menge der Wörter, die ???
- $L\big((a+b)^*aba(a+b)^*\big)$ ist die Menge der Wörter, die ???
- $L\big((b^*ab^*a)^*b^*\big)$ ist die Menge der Wörter, die ???

Wo treten reguläre Ausdrücke in der Praxis auf?

- Suchen und Ersetzen in Editoren
  (z.B. `vi`, `emacs`, ...)
- Pattern-Matching und Verarbeitung großer Texte und Datenmengen, z.B. beim Data-Mining
  (z.B. `sed`, `awk`, ...)
- Übersetzung von Programmiersprachen:
  Lexikalische Analyse – Umwandlung einer Folge von Zeichen (das Programm) in eine Folge von Tokens, in der bereits die Schlüsselwörter, Bezeichner, Daten, etc. identifiziert sind.
  (z.B. `lex`, `flex`, ...)

Diese Anwendungen beruhen auf dem Zusammenhang zu endlichen Automaten, den wir jetzt untersuchen werden.

Proposition

Ist γ ein regulärer Ausdruck, so ist $L(\gamma)$ eine reguläre Sprache.

Beweis: per Induktion über den Aufbau von γ.

- IA Da die Sprachen $\emptyset$, $\{\varepsilon\}$ und $\{a\}$ für $a \in \Sigma$ regulär sind, gilt die Aussage für die regulären Ausdrücke $\gamma \in \{\varnothing, \lambda\} \cup \Sigma$.
- IS Die Klasse der regulären Sprachen ist abgeschlossen unter Vereinigung, Konkatenation und Kleene-Iteration (Sätze auf Folien 4.14, 4.16 und 4.21). □

Proposition

Zu jedem DFA M gibt es einen regulären Ausdruck γ mit $L(M) = L(\gamma)$.

Beweis:

Sei $M = \big(\{1, \ldots, n\}, \Sigma, 1, \delta, E\big)$ ein DFA.

Wir konstruieren einen regulären Ausdruck γ mit $L(M) = L(\gamma)$.

Für ein Wort $w \in \Sigma^*$ sei

$$\mathrm{Pref}(w) = \{u \in \Sigma^* \mid \exists v : w = uv, \varepsilon \neq u \neq w\}$$

die Menge aller nicht-leeren echten Präfixe von w.

Für $i, j \in \{1, \ldots, n\}$ und $k \in \{0, \ldots, n\}$ sei

$$L_{i,j}^{k} = \left\{ w \in \Sigma^* \;\middle|\; \widehat{\delta}(i, w) = j, \forall u \in \mathrm{Pref}(w) \colon 1 \leq \widehat{\delta}(i, u) \leq k \right\}.$$

Intuitiv: Ein Wort w gehört zu $L_{i,j}^{k}$ genau dann, wenn w den Zustand i in den Zustand j überführt und wenn dabei kein Zwischenzustand (außer vielleicht ganz am Anfang und ganz am Ende) $> k$ vorkommt.

Behauptung: Für alle $i, j \in \{1, \ldots, n\}$ und $k \in \{0, \ldots, n\}$ existieren reguläre Ausdrücke $\gamma_{i,j}^{k}$ mit $L(\gamma_{i,j}^{k}) = L_{i,j}^{k}$.

Bemerkung: Falls $E = \{i_1, i_2, \ldots, i_\ell\}$, ergibt sich dann

$$L(\gamma_{1,i_1}^{n} + \gamma_{1,i_2}^{n} + \cdots + \gamma_{1,i_\ell}^{n}) = L(M)\,,$$

womit der Beweis abgeschlossen sein wird.

Beweis der Behauptung, d.h. Konstruktion von $\gamma_{i,j}^k$ durch Induktion über $k \in \{0, \ldots, n\}$.

IA $k = 0$. Es gilt:

$$L_{i,j}^0 = \begin{cases} \{a \in \Sigma \mid \delta(i,a) = j\} & \text{falls } i \neq j \\ \{a \in \Sigma \mid \delta(i,a) = j\} \cup \{\varepsilon\} & \text{falls } i = j \end{cases}$$

Gilt $\{a_1, \ldots, a_m\} = \{a \in \Sigma \mid \delta(i,a) = j\}$, so setzen wir:

$$\gamma_{i,j}^0 = \begin{cases} a_1 + a_2 + \cdots + a_m + \varnothing & \text{falls } i \neq j \\ a_1 + a_2 + \cdots + a_m + \lambda & \text{falls } i = j \end{cases}$$

In jedem Fall gilt dann $L(\gamma_{i,j}^0) = L_{i,j}^0$.

IV Sei $0 \leq k < n$ und seien die regulären Ausdrücke $\gamma^k_{p,q}$ für alle $p, q \in \{1, \ldots, n\}$ bereits konstruiert.

IS Sei $i, j \in \{1, \ldots, n\}$.

Dann gilt:

$$L^{k+1}_{i,j} = L^k_{i,j} \cup L^k_{i,k+1}\big(L^k_{k+1,k+1}\big)^* L^k_{k+1,j}$$

Begründung:

„$\subseteq$“: Sei $w \in L^{k+1}_{i,j}$ und sei $\ell \geq 0$ so, daß der Zustand $k+1$ auf dem eindeutigen mit w beschrifteten Pfad von i nach j genau ℓ mal als echter Zwischenzustand auftaucht.

1.Fall: $\ell = 0$, d.h. $k+1$ kommt nicht als echter Zwischenzustand vor.

$\rightsquigarrow \;\; w \in L^k_{i,j}$

2.Fall: $\ell > 0$.

$\rightsquigarrow w = w_0 w_1 \cdots w_{\ell-1} w_\ell$, wobei:

$$\begin{aligned} \widehat{\delta}(i, w_0) &= k+1 \\ \widehat{\delta}(k+1, w_p) &= k+1 \text{ für } 1 \leq p \leq \ell - 1 \\ \widehat{\delta}(k+1, w_\ell) &= j \end{aligned}$$

$\rightsquigarrow \;\; w_0 \in L^k_{i,k+1}$, $w_p \in L^k_{k+1,k+1}$ $(1 \leq p \leq \ell - 1)$, $w_\ell \in L^k_{k+1,j}$

$\rightsquigarrow \;\; w = w_0(w_1 \cdots w_{\ell-1}) w_\ell \in L^k_{i,k+1} \big(L^k_{k+1,k+1}\big)^* L^k_{k+1,j}$

Damit ist die Inklusion „$\subseteq$“ der Behauptung von Folie 5.13 gezeigt.

„$\supseteq$“:
Für alle $p, q \in \{1, \ldots, n\}$ gilt $L^k_{p,q} \subseteq L^{k+1}_{p,q}$.

Damit erhalten wir

$$\begin{aligned} & L^k_{i,j} \cup L^k_{i,k+1}\big(L^k_{k+1,k+1}\big)^* L^k_{k+1,j} \\ \subseteq \quad & L^{k+1}_{i,j} \cup L^{k+1}_{i,k+1}\big(L^{k+1}_{k+1,k+1}\big)^* L^{k+1}_{k+1,j} \\ \subseteq \quad & L^{k+1}_{i,j} \end{aligned}$$

Damit ist auch die Inklusion „$\supseteq$“ der Behauptung von Folie 5.13 gezeigt.

Es gilt also wirklich

$$L_{i,j}^{k+1} = L_{i,j}^{k} \cup L_{i,k+1}^{k} \big(L_{k+1,k+1}^{k}\big)^{*} L_{k+1,j}^{k} \,.$$

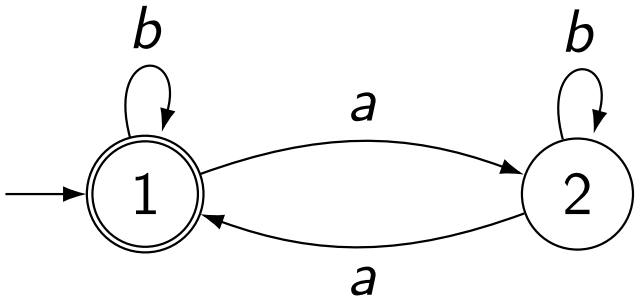
Außerdem haben wir bereits reguläre Ausdrücke $\gamma_{p,q}^{k}$ mit $L(\gamma_{p,q}^{k}) = L_{p,q}^{k}$ konstruiert.

Mit

$$\gamma_{i,j}^{k+1} = \gamma_{i,j}^{k} + \gamma_{i,k+1}^{k} \big(\gamma_{k+1,k+1}^{k}\big)^{*} \gamma_{k+1,j}^{k}$$

erhalten wir $L(\gamma_{i,j}^{k+1}) = L_{i,j}^{k+1}$. □

Beispiel: Betrachte den folgenden DFA:



Damit ergibt sich (bei Durchführung offensichtlicher Vereinfachungen):

$$\gamma^0_{1,1} = \lambda + b \qquad \gamma^0_{1,2} = a \qquad \gamma^0_{2,1} = a \qquad \gamma^0_{2,2} = \lambda + b$$

$$\begin{aligned}
\gamma^1_{1,1} &= \gamma^0_{1,1} + \gamma^0_{1,1}(\gamma^0_{1,1})^*\gamma^0_{1,1} = \lambda + b + (\lambda + b)(\lambda + b)^*(\lambda + b) \mathrel{\widehat{=}} b^* \\
\gamma^1_{1,2} &= \gamma^0_{1,2} + \gamma^0_{1,1}(\gamma^0_{1,1})^*\gamma^0_{1,2} = a + (\lambda + b)(\lambda + b)^* a \mathrel{\widehat{=}} b^* a \\
\gamma^1_{2,1} &= \gamma^0_{2,1} + \gamma^0_{2,1}(\gamma^0_{1,1})^*\gamma^0_{1,1} = a + a(\lambda + b)^*(\lambda + b) \mathrel{\widehat{=}} ab^* \\
\gamma^1_{2,2} &= \gamma^0_{2,2} + \gamma^0_{2,1}(\gamma^0_{1,1})^*\gamma^0_{1,2} = \lambda + b + a(\lambda + b)^* a \mathrel{\widehat{=}} \lambda + b + ab^* a \\
\gamma^2_{1,1} &= \gamma^1_{1,1} + \gamma^1_{1,2}(\gamma^1_{2,2})^*\gamma^1_{2,1} = b^* + b^* a(\lambda + b + ab^* a)^* ab^*
\end{aligned}$$

Zusammenfassender Satz

Sei $L \subseteq \Sigma^*$ eine Sprache. Dann sind äquivalent

1. L ist regulär, d.h. es gibt einen DFA M mit $L(M) = L$.
2. Es gibt einen NFA M mit $L(M) = L$.
3. L ist rechtslinear, d.h. es gibt eine rechtslineare Grammatik G mit $L(G) = L$.
4. Es gibt einen regulären Ausdruck γ mit $L(\gamma) = L$.

5. L ist erste Komponente einer Lösung eines linearen Gleichungssystems.
6. L ist in monadischer Logik 2. Stufe definierbar.
7. L ist abgeschlossen unter einer Kongruenz endlichen Indexes.
8. Es gibt Homomorphismus $\eta\colon \Sigma^* \to S$ in ein endliches Monoid S mit $L = \eta^{-1}\eta(L)$.
9. L wird von einem alternierenden Automaten akzeptiert.
10. L wird von einem 2-Weg-Automaten akzeptiert.

Weitere Forschungsfragen

1. DFAs mit speziellen Eigenschaften (z.B. „zählerfrei“): Welche Sprachen akzeptieren diese? Kann ich feststellen, ob ein allgemeiner DFA äquivalent zu einem mit spezieller Eigenschaft ist?
2. Größenvergleich von äquivalenten Mechanismen (z.B. „DFAs sind notwendigerweise exponentiell größer als NFAs“ oder „Übersetzung von regulären Ausdrücken in NFA ist in polynomieller Zeit möglich“)

Zusammenfassung

verschiedene Modelle zur Beschreibung aller regulären Sprachen:

- Rechtslineare Grammatiken:
  - Verbindung zur Chomsky-Hierarchie
  - erzeugen Sprachen
  - nicht geeignet, um zu entscheiden, ob ein geg. Wort zur Sprache gehört
- NFAs:
  - erlauben kleine, kompakte Darstellung
  - intuitive graphische Notation
  - nicht geeignet, um zu entscheiden, ob ein geg. Wort zur Sprache gehört
- DFAs:
  - für effiziente Beantwortung der Frage, ob ein Wort zur Sprache gehört
  - sind u.U. exponentiell größer als NFA
- Reguläre Ausdrücke:
  - erlauben kompakte Darstellung in Textform

Zusammenfassung 5. Vorlesung

in dieser Vorlesung neu

- Abschluß der Klasse der rechtslinearen Sprachen unter positiver Iteration und unter Kleene-Iteration
- reguläre Ausdrücke beschreiben genau die rechtslinearen Sprachen, sind also gleich ausdrucksstark wie Typ-3-Grammatiken, DFAs und NFAs

kommende Vorlesung

- Gibt es Sprachen, die nicht rechtslinear sind?
- Wie kannn ich von einer geg. Sprachen zeigen, daß sie nicht rechtslinear ist?