1 Reguläre Sprachen

1.1 Reguläre Sprachen und endliche Automaten

Definition 1
Sei $\Sigma = \{a, b, \ldots\}$ ein endliches *Alphabet*. Ein *Wort* über Σ ist eine Folge $w = a_0 \ldots a_{n-1}$, wobei $a_i \in \Sigma$ für $i = 1, \ldots, n$. Wir schreiben $w(i)$ für das i-te Symbol a_i von w. Das *leere Wort* wird mit ϵ bezeichnet. Σ^* bezeichnet die Menge aller *Wörter* über Σ, Σ^+ die Menge aller nicht-leeren Wörter über Σ. Die *Konkatenation* zweier Wörter w und v ist das Wort wv.

Eine *Sprache* ist ein $L \subseteq \Sigma^*$. Für Sprachen L, L_1 und L_2 definiere $L_1 L_2 := \{w_1 w_2 \mid w_1 \in L_1, w_2 \in L_2\}$ und $L^* := \{w_1 \ldots w_n \mid n \in \mathbb{N}, w_i \in L \text{ für alle } i = 1, \ldots, n\}$.

Lemma 1
Für alle $L \subseteq \Sigma^*$ gilt: $L^* = \bigcup_{i \in \mathbb{N}} L^i$, wobei $L^0 := \{\epsilon\}$ und $L^{i+1} = LL^i$.

Definition 2
Die Klasse REG ist die kleinste Klasse von Sprachen, für die gilt:

- $\emptyset \in \text{REG}$, $\{\epsilon\} \in \text{REG}$ und $\{a\} \in \text{REG}$ für jedes $a \in \Sigma$,
- wenn $L_1, L_2 \in \text{REG}$, dann $L_1 \cup L_2 \in \text{REG}$, $L_1 L_2 \in \text{REG}$ und $L_1^* \in \text{REG}$.

Definition 3
Reguläre Ausdrücke über Σ sind gegeben durch die folgende Grammatik.

$$\alpha \ ::= \ \emptyset \mid \epsilon \mid a \mid \alpha\alpha \mid \alpha \cup \alpha \mid \alpha^*$$

wobei $a \in \Sigma$. Ein regulärer Ausdruck α definiert eine Sprache $L \subseteq \Sigma^*$ in natürlicher Weise.

$$\begin{aligned} L(\emptyset) &:= \emptyset \\ L(a) &:= \{a\} \\ L(\epsilon) &:= \{\epsilon\} \\ L(\alpha\beta) &:= L(\alpha)L(\beta) \\ L(\alpha \cup \beta) &:= L(\alpha) \cup L(\beta) \\ L(\alpha^*) &:= (L(\alpha))^* \end{aligned}$$

Wir benutzen auch die Abkürzung $\alpha^+ := \alpha\alpha^*$.

Satz 1
Für alle Sprachen L gilt: L is regulär gdw. es einen regulären Ausdruck α gibt mit $L = L(\alpha)$.

Die Richtung $\Leftarrow$ wird einfach durch Induktion über den Aufbau regulärer Ausdrücke gezeigt. Genauso kann man aber auch durch Induktion über den Aufbau von REG die Richtung $\Rightarrow$ zeigen.

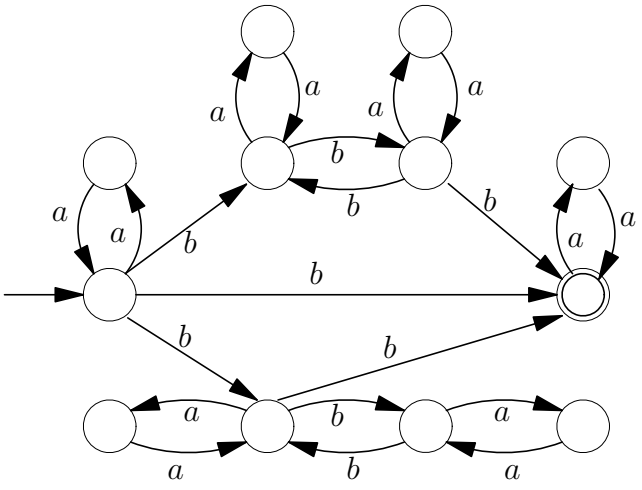
Definition 4
Ein *nicht-deterministischer Automat* (NFA) ist ein Tupel $\mathcal{A} = (Q, \Sigma, q_0, \delta, F)$ mit

- endlicher *Zustandsmenge* Q,
- *Eingabealphabet* Σ,
- *Startzustand* $q_0 \in Q$,
- *Transitionsfunktion* $\delta : Q \times \Sigma \to 2^Q$,
- Endzustandsmenge $F \subseteq Q$.

Ein *Lauf* eines NFA $\mathcal{A}$ auf einem Wort $w = a_0 \dots a_{n-1}$ ist eine Folge $q_0 \dots q_n$, so dass für alle $i = 0, \dots, n-1$ gilt: $q_{i+1} \in \delta(q_i, a_i)$. Der Lauf heißt *akzeptierend*, falls $q_n \in F$. Sei $L(\mathcal{A}) := \{w \in \Sigma^* \mid$ es gibt einen akzeptierenden Lauf von $\mathcal{A}$ auf $w\}$ die von $\mathcal{A}$ *erkannte Sprache*.

Ein NFA $\mathcal{A}$ heißt *deterministisch* (DFA), falls für alle $q \in Q$ und alle $a \in \Sigma$ gilt: $|\delta(q, a)| \leq 1$. Gilt $|\delta(q, a)| \geq 1$, so heißt der Automat *total*.

Beispiel 1
NFAs lassen sich am einfachsten grafisch darstellen. Sei $\mathcal{A}$ der folgende NFA.



Dann gilt

$$L(\mathcal{A}) \;=\; (aa)^*\Big(\; b \;\cup\; \big(b(aa)^*b(aa)^*\big)^+ b \;\cup\; b\big((aa)^*b(aa)^*b\big)^*b\Big)(aa)^*$$

Mit anderen Worten: L besteht aus der Menge aller Wörter, in denen a's nur paarweise vorkommen und die entweder 1, oder eine ungerade Anzahl ≥ 3 oder eine gerade Anzahl ≥ 2 an b's enthalten. Diese Sprache lässt sich auch einfacher beschreiben: $L(\mathcal{A}) = (aa)^*b\big((aa)^* \cup b\big)^*$. Außerdem gilt $L(\mathcal{A}) = L(\mathcal{A}')$ für den folgenden NFA $\mathcal{A}'$.

Satz 2 (Kleene)
Die folgenden Aussagen sind äquivalent für jedes $L \subseteq \Sigma^*$.

1. $L \in$ REG*,*
2. *es gibt einen NFA $\mathcal{A}$ mit $L = L(\mathcal{A})$,*
3. *es gibt einen DFA $\mathcal{A}$ mit $L = L(\mathcal{A})$.*

Die Implikation $(a) \Rightarrow (b)$ kann man durch eine lineare und induktive Übersetzung regulärer Ausdrücke in NFAs zeigen. Der Teil $(b) \Rightarrow (c)$ besteht aus der Potenzmengenkonstruktion. Teil $(c) \Rightarrow (a)$ oder auch $(b) \Rightarrow (a)$ kann schließlich gezeigt werden, indem man einen DFA oder NFA als Gleichungssystem über Sprachen auffasst und dieses mithilfe des Ardenschen Lemmas sukzessive löst und dabei reguläre Ausdrücke für die einzelnen Gleichungen findet.

1.2 Abschlusseigenschaften und Entscheidbarkeit

Satz 3
Die Klasse REG *ist abgeschlossen unter*

1. Vereinigung *($L_1, L_2 \in$ REG $\Rightarrow L_1 \cup L_2 \in$ REG$)$,*
2. Konkatenation *($L_1, L_2 \in$ REG $\Rightarrow L_1 L_2 \in$ REG$)$,*
3. Kleene-Stern */ endliche* Iteration *($L \in$ REG $\Rightarrow L^* \in$ REG$)$,*
4. Durchschnitt *($L_1, L_2 \in$ REG $\Rightarrow L_1 \cap L_2 \in$ REG$)$,*
5. Komplement *($L \in$ REG $\Rightarrow \overline{L} := \Sigma^* \setminus L \in$ REG$)$,*
6. Differenz *($L_1, L_2 \in$ REG $\Rightarrow L_1 \setminus L_2 \in$ REG$)$,*
7. Spiegelung *($L \in$ REG $\Rightarrow \{a_n \dots a_1 \mid a_1 \dots a_n \in L\} \in$ REG,*
8. Homomorphismen *($L \in$ REG $\Rightarrow \{h(w) \mid w \in L\} \in$ REG, wobei $h(\epsilon) = \epsilon$, $h(av) = g(a)h(v)$ für eine Abbildung $g : \Sigma \to \Delta^*$ und ein Alphabet Δ.*

Beweisskizze: (1)–(3) gehen trivialerweise aus der Definition von REG hervor. (4) zeigt man mittels einer Kreuzproduktkonstruktion auf NFAs; (5) durch Vertauschen von End- und Nicht-Endzuständen in einem DFA; (6) wird auf (4) und (5) zurückgeführt; (7) durch Umkehrung der Transitionen in einem NFA (da bei der Umkehrung i.A. nichtdeterministische Transitionen entstehen); (8) Übung.

Satz 4
Die folgenden Probleme sind für reguläre Sprachen L, L' und Wörter $w \in \Sigma^*$ entscheidbar.

1. *Das* Wortproblem*: Gegeben w und L, gilt $w \in L$?*
2. *Das* Leerheitsproblem*: Gegeben L, ist $L = \emptyset$?*
3. *Das* Universalitätsproblem*: Gegeben L, ist $L = \Sigma^*$?*
4. *Das* Schnittproblem*: Gegeben L, L', ist $L \cap L' = \emptyset$?*
5. *Das* Äquivalenzproblem*: Gegeben L, L', ist $L = L'$?*
6. *Das* Inklusionsproblem*: Gegeben L, L', ist $L \subseteq L'$?*

Beweis als Übung.