

Abbildung 20: DFA  $\mathcal{M}$  aus Beispiel 2.30

**Beispiel 2.30 (Ersetzungsmethode).** Wir betrachten den DFA  $\mathcal{M} = (\{q_0, q_1, q_2\}, \{a, b, c\}, \delta, q_0, \{q_1\})$  aus Abbildung 20. Das Ziel ist es, jedem der drei Zustände  $q_i$  einen regulären Ausdruck  $\alpha_i$  für die Sprache  $L_i$  zuzuordnen, wobei  $L_i$  für die Sprache bestehend aus allen Wörtern  $w$  mit  $\delta(q_i, w) = q_1$  steht. Für die akzeptierte Sprache von  $\mathcal{M}$  gilt dann  $\mathcal{L}(\mathcal{M}) = \mathcal{L}(\alpha_0)$ .

Zunächst erstellen wir ein Gleichungssystem für die  $\alpha_i$ 's:

$$\begin{aligned}\alpha_0 &\equiv a\alpha_1 \\ \alpha_1 &\equiv b\alpha_0 + c\alpha_2 + \varepsilon \\ \alpha_2 &\equiv b\alpha_1\end{aligned}$$

Wir setzen nun die Gleichung für  $\alpha_0$  in die Gleichung für  $\alpha_1$  ein und erhalten:

$$\alpha_1 \equiv \underbrace{ba}_{\beta} \alpha_1 + \underbrace{c\alpha_2 + \varepsilon}_{\gamma}$$

Anwenden der oben genannten Regel ergibt:

$$\alpha_1 \equiv (ba)^*(c\alpha_2 + \varepsilon)$$

Einsetzen in die Gleichung für  $\alpha_2$  liefert:

$$\alpha_2 \equiv b(ba)^*(c\alpha_2 + \varepsilon) \equiv b(ba)^*c\alpha_2 + b(ba)^*$$

Durch Anwenden der genannten Regel mit  $\beta = b(ba)^*c$  und  $\gamma = b(ba)^*$  erhalten wir:

$$\alpha_2 \equiv (b(ba)^*c)^* b(ba)^*$$

Einsetzen der Lösung für  $\alpha_2$  in die Gleichung für  $\alpha_1$  ergibt.

$$\alpha_1 \equiv b\alpha_0 + c(b(ba)^*c)^* b(ba)^* + \varepsilon$$

Durch Einsetzen von  $\alpha_1$  in  $\alpha_0$  und Anwendung der Ersetzungsregel erhält man den gesuchten regulären Ausdruck für  $\alpha_0$

$$\alpha_0 \equiv (ab)^* (ac((b(ba)^*c)^* b(ba)^*) + a)$$

Selbstverständlich gibt es hier für das Ersetzen viele Möglichkeiten, die letztendlich zu verschiedenen (aber äquivalenten) Ausdrücken führen können. ■

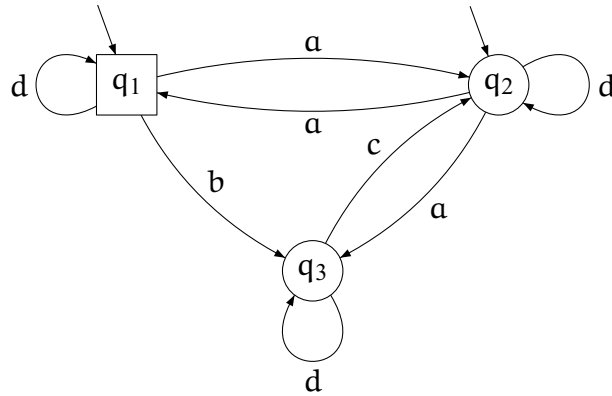


Abbildung 21: NFA  $\mathcal{M}$  aus Beispiel 2.31

**Beispiel 2.31 (Ersetzungsmethode).** Als weiteres Beispiel betrachten wir den NFA  $\mathcal{M}$  aus Abbildung 21 mit dem Alphabet  $\Sigma = \{a, b, c, d\}$ . Wie zuvor suchen wir reguläre Ausdrücke  $\alpha_i$  für die Sprache bestehend aus allen Wörtern  $w$ , so dass  $q_1 \in \delta(q_i, w)$ . Die akzeptierte Sprache von  $\mathcal{M}$  ist dann durch den regulären Ausdruck  $\alpha_1 + \alpha_2$  gegeben, da  $q_1$  und  $q_2$  Anfangszustände von  $\mathcal{M}$  sind.

Das induzierte Gleichungssystem für die  $\alpha_i$ 's lautet:

$$\begin{aligned}\alpha_1 &\equiv d\alpha_1 + a\alpha_2 + b\alpha_3 + \varepsilon \\ \alpha_2 &\equiv a\alpha_1 + d\alpha_2 + a\alpha_3 \\ \alpha_3 &\equiv c\alpha_2 + d\alpha_3\end{aligned}$$

Aufgrund der rekursiven Gestalt des Gleichungssystems führt das Einsetzen der Gleichungen für  $\alpha_i$  in die Gleichungen der beiden anderen  $\alpha_j$ 's hier nicht zur Elimination von  $\alpha_i$ . Dennoch können wir sukzessive die Regel "aus  $\alpha \equiv \beta\alpha + \gamma$ ,  $\varepsilon \notin \mathcal{L}(\beta)$  folgt  $\alpha \equiv \beta^*\gamma$ " anwenden. Z.B. kann man aus  $\alpha_3 \equiv c\alpha_2 + d\alpha_3$  auf

$$\alpha_3 \equiv d^*c\alpha_2$$

schliessen. Einsetzen in die Gleichung für  $\alpha_1$  liefert:

$$\begin{aligned}\alpha_1 &\equiv d\alpha_1 + a\alpha_2 + b\alpha_3 + \varepsilon \\ &\equiv d\alpha_1 + a\alpha_2 + bd^*c\alpha_2 + \varepsilon \\ &\equiv d\alpha_1 + (a + bd^*c)\alpha_2 + \varepsilon\end{aligned}$$

und somit

$$\begin{aligned}\alpha_1 &\equiv d^*((a + bd^*c)\alpha_2 + \varepsilon) \\ &\equiv d^*(a + bd^*c)\alpha_2 + d^*\end{aligned}$$

Diese Darstellungen von  $\alpha_1$  und  $\alpha_3$  in Abhängigkeit von  $\alpha_2$  können nun in die Gleichung für  $\alpha_2$  eingesetzt werden. Wir erhalten:

$$\begin{aligned}\alpha_2 &\equiv a\alpha_1 + d\alpha_2 + a\alpha_3 \\ &\equiv a(d^*(a + bd^*c)\alpha_2 + d^*) + d\alpha_2 + ad^*c\alpha_2 \\ &\equiv ad^*(a + bd^*c)\alpha_2 + d\alpha_2 + ad^*c\alpha_2 + ad^* \\ &\equiv (ad^*(a + bd^*c) + d + ad^*c)\alpha_2 + ad^*\end{aligned}$$

und somit eine Lösung für  $\alpha_2$ :

$$\alpha_2 \equiv (\text{ad}^*(a + \text{bd}^*c) + d + \text{ad}^*c)^* \text{ad}^*$$

Lösungen für  $\alpha_1$  und  $\alpha_3$  ergeben sich nun durch Einsetzen dieser Lösung für  $\alpha_2$  in die oben genannten Darstellungen von  $\alpha_1$  und  $\alpha_3$  in Abhängigkeit von  $\alpha_2$ . ■

**2. Verfahren “NFA  $\rightsquigarrow$  regulärer Ausdruck”:** dynamisches Programmieren. Wie zuvor sei  $\mathcal{M} = (Q, \Sigma, \delta, Q_0, F)$  ein NFA. Für  $q, p \in Q$  definieren wir

$$L_{q,p} \stackrel{\text{def}}{=} \{ w \in \Sigma^* : p \in \delta(q, w) \} = \{ w \in \Sigma^* : q \xrightarrow{w} p \}$$

Offenbar gilt  $L_{q,p} = \mathcal{L}(\mathcal{M}_{q,p})$ , wobei  $\mathcal{M}_{q,p}$  der NFA ist, der aus  $\mathcal{M}$  entsteht, indem  $q$  als einziger Anfangszustand und  $p$  als einziger Endzustand deklariert wird, d.h.,  $\mathcal{M}_{q,p} = (Q, \Sigma, \delta, \{q\}, \{p\})$ . Daher ist die Sprache  $L_{q,p}$  regulär. Das Ziel ist es nun, reguläre Ausdrücke  $\alpha[q,p]$  für alle Zustände  $q, p \in Q$  anzugeben, so dass  $\mathcal{L}(\alpha[q,p]) = L_{q,p}$ . Liegen diese regulären Ausdrücke  $\alpha_{q,p}$  vor, dann erhält man einen regulären Ausdruck  $\alpha$  für die Sprache  $\mathcal{L}(\mathcal{M})$  wie folgt:

$$\alpha \stackrel{\text{def}}{=} \sum_{q \in Q_0} \sum_{p \in F} \alpha[q,p]$$

Tatsächlich ist  $\alpha$  dann ein regulärer Ausdruck für die von  $\mathcal{M}$  akzeptierte Sprache, da nämlich:

$$\mathcal{L}(\alpha) = \bigcup_{q \in Q_0} \bigcup_{p \in F} \mathcal{L}(\alpha[q,p]) = \bigcup_{q \in Q_0} \bigcup_{p \in F} L_{q,p} = \mathcal{L}(\mathcal{M})$$

Im Folgenden gehen wir von einer beliebigen, aber festen Nummerierung der Zustände von  $\mathcal{M}$  aus, etwa

$$Q = \{q_1, \dots, q_n\},$$

wobei  $q_1, \dots, q_n$  paarweise verschieden sind. Zur Vereinfachung identifizieren wir im Folgenden die Zustände mit ihren Nummern und schreiben  $L_{i,j}$  für  $L_{q_i, q_j}$  und  $\alpha[i,j]$  statt  $\alpha[q_i, q_j]$ .

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

$$L_{i,j}^k \stackrel{\text{def}}{=} \left\{ w \in \Sigma^* : \begin{array}{l} \text{es gibt einen Lauf für } w \text{ der Form } q_i p_1 \dots p_m q_j \\ \text{mit } m \geq 0 \text{ und } \{p_1, p_2, \dots, p_m\} \subseteq \{q_1, \dots, q_k\} \end{array} \right\}$$

Der Sonderfall  $m = 0$  bedarf einer Erklärung. Ist  $i = j$ , so sind für  $m = 0$  alle Läufe der Länge 0 und 1 zugelassen. Hierfür gibt es nur zwei Kandidaten, nämlich den Lauf  $q_i$  für das leere Wort und den Lauf  $q_i q_i$  für alle Wörter  $a$ , für die es eine  $a$ -Schleife an Zustand  $q_i$  gibt. Im Sonderfall  $m = 0$  und  $i \neq j$  ist nur der Lauf  $q_i q_j$  der Länge 1 zugelassen. Dieser ist ein Lauf für alle Wörter  $a$ , für die es eine  $a$ -Transition von  $q_i$  nach  $q_j$  gibt.<sup>5</sup>

Die Sprache  $L_{i,j}^k$  besteht also aus allen Wörtern  $w$ , für die es einen Lauf gibt, der das Wort  $w$  vollständig liest und in Zustand  $q_i$  beginnt, in Zustand  $q_j$  endet und der – eventuell abgesehen

<sup>5</sup>Eine technische Bemerkung zu dem Begriff “Lauf”. Läufe für NFA wurden als Zustandsfolgen definiert, die in einem Anfangszustand beginnen. Der Begriff eines Laufs bezieht sich hier also streng genommen nicht auf  $\mathcal{M}$ , sondern auf den NFA  $\mathcal{M}' = (Q, \Sigma, \delta, \{q_i\}, F)$  mit eindeutigem Anfangszustand  $q_i$ .

vom ersten Zustand  $q_i$  und letzten Zustand  $q_j$  – höchstens die ersten  $k$  Zustände  $q_1, \dots, q_k$  besucht. Ist nun  $q = q_i$  und  $p = q_j$ , so ist

$$L_{q_i, q_j} = L_{i, j} = L_{i, j}^n,$$

da in der Definition von  $L_{i, j}^k$  für  $k = n$  alle Läufe von  $q = q_i$  nach  $p = q_j$  betrachtet werden.

Die Idee ist nun, durch Induktion nach  $k$  reguläre Ausdrücke für die Sprachen  $L_{i, j}^k$  anzugeben. Wir betrachten zuerst den Fall  $k = 0$  und  $i \neq j$ . Die Sprache  $L_{i, j}^0$  besteht nur aus Wörtern der Länge 1, nämlich

$$L_{i, j}^0 = \{a \in \Sigma : q_i \xrightarrow{a} q_j\}.$$

Ist also  $\{a \in \Sigma : q_j \in \delta(q_i, a)\} = \{a_1, a_2, \dots, a_r\}$ , so ist

$$\alpha^0[i, j] \stackrel{\text{def}}{=} a_1 + a_2 + \dots + a_r$$

ein regulärer Ausdruck mit  $\mathcal{L}(\alpha^0[i, j]) = L_{i, j}^0$ , wobei  $i \neq j$ . Falls  $r = 0$ , d.h., falls  $\{a \in \Sigma : q_j \in \delta(q_i, a)\} = \emptyset$ , so steht  $a_1 + \dots + a_r$  für den regulären Ausdruck  $\emptyset$ . Für den Sonderfall  $i = j$  ist das leere Wort zu berücksichtigen. Dieses liegt stets in  $L_{i, i}^0$ . Ist also  $\{a \in \Sigma : q_i \in \delta(q_i, a)\} = \{a_1, a_2, \dots, a_r\}$ , so ist

$$\alpha^0[i, i] \stackrel{\text{def}}{=} a_1 + a_2 + \dots + a_r + \varepsilon$$

ein regulärer Ausdruck mit  $\mathcal{L}(\alpha^0[i, i]) = L_{i, i}^0$ .

Wir nehmen nun an, dass  $k \in \{0, 1, \dots, n-1\}$  und dass reguläre Ausdrücke  $\alpha^k[i, j]$  mit  $\mathcal{L}(\alpha^k[i, j]) = L_{i, j}^k$  für alle  $i, j$  bereits vorliegen. Wir leiten nun aus den Ausdrücken  $\alpha^k[\dots]$  reguläre Ausdrücke  $\alpha^{k+1}[i, j]$  für alle Zustandsnummern  $i, j$  her. Gedanklich teilen wir die in der Definition von  $L_{i, j}^{k+1}$  vorkommenden Läufe  $q_i p_1 \dots p_m q_j$  in zwei Gruppen ein:

- diejenigen Läufe, die nicht durch den  $(k+1)$ -ten Zustand  $q_{k+1}$  laufen (eventuell abgesehen von dem ersten bzw. letzten Zustand, sofern  $k+1 = i$  oder  $k+1 = j$ ), d.h.,

$$\{p_1, \dots, p_m\} \subseteq \{q_1, \dots, q_k\}.$$

Diese Läufe  $q_i p_1 \dots p_m q_j$  sind bereits in der Definition von  $L_{i, j}^k$  erfasst.

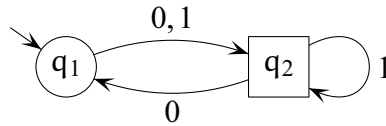
- diejenigen Läufe, die mindestens einmal durch den Zustand  $q_{k+1}$  führen. In diesem Fall kann  $q_i p_1 \dots p_m q_j$  in Fragmente zerlegt werden:

$$\begin{array}{ccccccc} q_i & \dots & q_{k+1} & \dots & q_{k+1} & \dots & q_{k+1} & \dots & q_j \\ & & \uparrow & & \uparrow & & \uparrow & & \\ & & \text{Zustände in} & & \text{Zustände in} & & \text{Zustände in} & & \\ & & \{q_1, \dots, q_k\} & & \{q_1, \dots, q_k\} & & \{q_1, \dots, q_k\} & & \end{array}$$

Daher gilt  $L_{i, j}^{k+1} = L_{i, j}^k \cup L_{i, k+1}^k (L_{k+1, k+1}^k)^* L_{k+1, j}^k$  und wir können  $\alpha^{k+1}[i, j]$  wie folgt definieren:

$$\alpha^{k+1}[i, j] \stackrel{\text{def}}{=} \alpha^k[i, j] + \alpha^k[i, k+1] (\alpha^k[k+1, k+1])^* \alpha^k[k+1, j].$$

**Beispiel 2.32 (NFA  $\rightsquigarrow$  regulärer Ausdruck).** Wir veranschaulichen die angegebene Methode exemplarisch an folgendem Automaten  $\mathcal{M}$ .



Da  $\mathcal{M}$  nur einen Anfangszustand (nämlich  $q_1$ ) und einen Endzustand (nämlich  $q_2$ ) hat, ist  $\mathcal{L}(\mathcal{M}) = \mathcal{L}(\alpha^2[1,2])$ . Wir behandeln zunächst den Fall  $k = 0$ :

$$\begin{aligned} \alpha^0[1,1] &= \varepsilon & \alpha^0[1,2] &= 0+1 \\ \alpha^0[2,1] &= 0 & \alpha^0[2,2] &= 1+\varepsilon \end{aligned}$$

Für  $k = 1$  erhalten wir z.B.

$$\alpha^1[1,1] = \alpha^0[1,1] + \alpha^0[1,1] (\alpha^0[1,1])^* \alpha^0[1,1] = \varepsilon + \varepsilon \varepsilon^* \varepsilon$$

und somit  $\alpha^1[1,1] \equiv \varepsilon$ . Insgesamt erhalten wir für  $k = 1$  folgende Ausdrücke, die zu den Ausdrücken  $\alpha^1[i,j]$  äquivalent sind:

$$\begin{aligned} \alpha^1[1,1] &\equiv \varepsilon & \alpha^1[1,2] &\equiv 0+1 \\ \alpha^1[2,1] &\equiv 0 & \alpha^1[2,2] &\equiv 1+\varepsilon+00+01 \end{aligned}$$

Hieraus ergeben sich die regulären Ausdrücke für  $k = 2$ :

$$\begin{aligned} \alpha^2[1,1] &= \alpha^1[1,1] + \alpha^1[1,2] (\alpha^1[2,2])^* \alpha^1[2,1] \\ &\equiv \varepsilon + (0+1)(\varepsilon+1+00+01)^*0 \\ \alpha^2[1,2] &= \alpha^1[1,2] + \alpha^1[1,2] (\alpha^1[2,2])^* \alpha^1[2,2] \\ &\equiv 0+1 + (0+1)(\varepsilon+1+00+01)^*(\varepsilon+1+00+01) \end{aligned}$$

Analog können  $\alpha^2[2,1]$  und  $\alpha^2[2,2]$  (bzw. reguläre Ausdrücke, die zu  $\alpha^2[2,1]$  und  $\alpha^2[2,2]$  äquivalent sind) berechnet werden. ■

Wie das obige Beispiel zeigt, können die regulären Ausdrücke  $\alpha^k[i,j]$  extrem lang werden. Tatsächlich ist die Länge  $\alpha^k[i,j]$  exponentiell in  $k$ , sofern keine syntaktischen Vereinfachungen an den Zwischenergebnissen vorgenommen werden. Dies erklärt sich daraus, dass die Ausdrücke  $\alpha^k[\dots]$  jeweils vier Teilausdrücke  $\alpha^{k-1}[\dots]$  enthalten. Daher ist eine untere Schranke für die minimale Länge  $T(k)$  der Ausdrücke  $\alpha^k[i,j]$  durch die Rekurrenz

$$T(0) \geq 1, \quad T(k+1) \geq 4T(k) + 1 \quad \text{für } k \geq 0$$

gegeben. Durch Induktion nach  $k$  kann nun gezeigt werden, dass  $T(k) \geq 4^k$ . Wir zeigen hier kurz, wie man – ohne Raten – zu der Lösung  $T(k) \geq 4^k$  der obigen Rekurrenz gelangt. Hierzu betrachten wir ein hinreichend großes  $k$  und setzen sukzessive die Ungleichungen für  $T(k)$ ,  $T(k-1)$ , etc. ein:

$$\begin{aligned} T(k) &\geq 4 T(k-1) + 1 && \geq 4 \cdot (4T(k-2) + 1) + 1 \\ &\geq 4^2 T(k-2) + 4 + 1 && \geq 4^2 \cdot (4T(k-3) + 1) + 4 + 1 \\ &\geq 4^3 T(k-3) + 4^2 + 4 + 1 && \geq 4^4 \cdot (4T(k-4) + 1) + 4^2 + 4 + 1 \\ &\vdots \\ &\geq 4^\ell T(k-\ell) + 4^{\ell-1} + 4^{\ell-2} + \dots + 4^2 + 4 + 1 \end{aligned}$$

Mit  $k = \ell$  und der Ungleichung  $T(0) \geq 1$  erhalten wir eine Teilsumme der geometrischen Reihe als untere Schranke für  $T(k)$ :

$$T(k) \geq 4^k + 4^{k-1} + \dots + 4^2 + 4 + 1,$$

also ist  $T(k) \geq 4^k$ . In analoger Weise kann man eine obere Schranke für die Länge der regulären Ausdrücke  $\alpha^k[i, j]$  berechnen. Sei  $S(k)$  die maximale Länge, die die Ausdrücke  $\alpha^k[i, j]$  haben können. Man kann nun die Rekurrenz  $S(0) = \mathcal{O}(|\Sigma|)$  und  $S(k) \leq 4S(k-1) + C$  verwenden, wobei  $C$  eine Konstante ist, deren exakter Wert für eine asymptotische obere Schranke von  $S(k)$  unerheblich ist. Die obere Schranke  $\mathcal{O}(|\Sigma|)$  für die Länge der Ausdrücke  $\alpha^0[i, j]$  ergibt sich aus der Tatsache, dass der längste mögliche Ausdruck für  $\alpha^0[i, j]$  die Form  $a_1 + a_2 + \dots + a_r + \varepsilon$  hat, wobei  $\Sigma = \{a_1, \dots, a_r\}$  und  $r = |\Sigma|$ . Die präzise Länge dieses Ausdrucks ist  $(r+1) + 3r = 4r + 1$ , da für jedes Summenzeichen drei Längeneinheiten zu veranschlagen sind, von denen zwei auf die unterdrückten Klammern und eine Längeneinheit auf das Zeichen  $+$  entfallen. Entscheidend ist hier jedoch lediglich, dass die asymptotische Länge durch ein Vielfaches von  $r = |\Sigma|$  nach oben beschränkt ist. Die Rekurrenz  $S(0) = \mathcal{O}(|\Sigma|)$  und  $S(k) \leq 4 \cdot S(k-1) + C$  hat die Lösung  $S(k) = \mathcal{O}(|\Sigma|4^k)$ . Betrachtet man  $\Sigma$  als festes Alphabet, so kann  $|\Sigma|$  wie eine Konstante behandelt werden und es folgt  $S(k) = \mathcal{O}(4^k)$ . Die berechneten regulären Ausdrücke haben also im besten und im schlimmsten Fall exponentielle Länge. Die Kosten für die Laufzeit des Verfahrens können daher mit

$$\sum_{k=0}^{n-1} n^2 \cdot 4^k = n^2 \cdot \sum_{k=0}^{n-1} 4^k = n^2 \cdot \left( \frac{4^n}{3} - \frac{1}{3} \right)$$

angesetzt werden. Die Laufzeit ist also asymptotisch durch  $n^2 \cdot 4^n$  nach oben und unten beschränkt. Dasselbe gilt für den Platzbedarf.