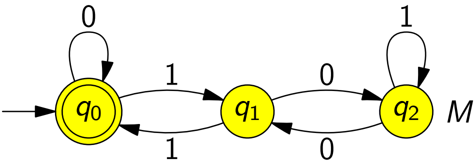
Übersicht

Deterministische endliche Automaten



DFA: Modell für Spracherkennung

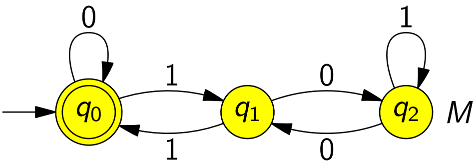
Formale Definition eines DFAs

Definition 2.2.1

Ein deterministischer endlicher Automat (DFA) ist ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$ mit

- *Q, der endlichen Menge der Zustände,*
- *Σ, dem Eingabealphabet,*
- *$\delta \colon Q \times \Sigma \to Q$, der Übergangsfunktion,*
- *$q_0 \in Q$, dem Anfangszustand und*
- *$F \subseteq Q$, der Menge der Endzustände.*

Beispiel



$M = (Q, \Sigma, \delta, q_0, F)$ wobei

- $Q = \{q_0, q_1, q_2\}$
- $\Sigma = \{0, 1\}$
- $\delta\colon Q \times \Sigma \to Q,\ \delta(q_0, 0) = q_0,\ \delta(q_0, 1) = q_1,$
  $\delta(q_1, 0) = q_2,\ \delta(q_1, 1) = q_0,$
  $\delta(q_2, 0) = q_1,\ \delta(q_2, 1) = q_2$
- $F = \{q_0\}$

Die Sprache eines DFA

Definition 2.2.2

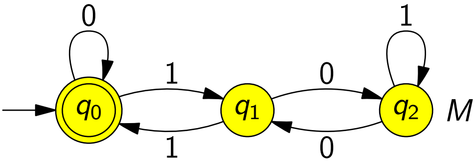
Es sei $M = (Q, \Sigma, \delta, q_0, F)$ *ein DFA.*

Erweitere $\delta\colon Q \times \Sigma \to Q$ *auf* $\hat{\delta}\colon Q \times \Sigma^* \to Q$*:*

1. $\hat{\delta}(q, \epsilon) = q$
2. $\hat{\delta}(q, w\,a) = \delta(\hat{\delta}(q, w), a)$ *für* $a \in \Sigma$, $w \in \Sigma^*$.

Definiere die Sprache von M*, in Zeichen* $L(M)$ *als*

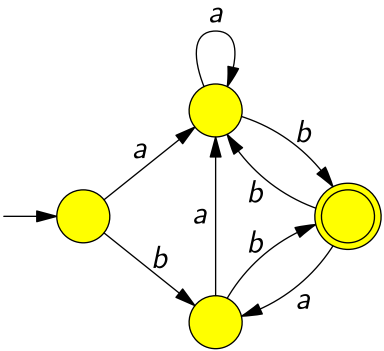
$$L(M) = \{w \in \Sigma^* \mid \hat{\delta}(q_0, w) \in F\}.$$



Wird das Wort 010 akzeptiert?

$$\begin{aligned}\hat{\delta}(q_0, 010) &= \delta(\hat{\delta}(q_0, 01), 0)\\ &= \delta(\delta(\delta(q_0, 0), 1), 0)\\ &= \delta(\delta(q_0, 1), 0)\\ &= \delta(q_1, 0) = q_2\end{aligned}$$

Nein, denn $\hat{\delta}(q_0, 010) \notin F$



Als regulärer Ausdruck: Welche Sprache wird akzeptiert?

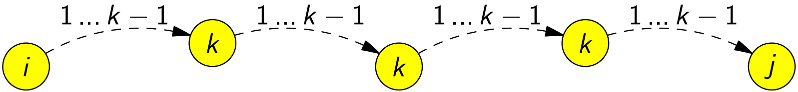
$M = (Q, \Sigma, \delta, q_1, F)$ mit $Q = \{q_1, \cdots, q_n\}$.

Konstruiere einen regulären Ausdruck, der $L(M)$ erzeugt.

Definiere $L_{ij}^k \subseteq \Sigma^*$ für $1 \leq i, j \leq n$, $0 \leq k \leq n$:

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

$$L_{ij}^k := L_{ij}^{k-1} \cup L_{ik}^{k-1}(L_{kk}^{k-1})^* L_{kj}^{k-1} \text{ für } k > 0$$



$M = (Q, \Sigma, \delta, q_1, F)$ mit $Q = \{q_1, \cdots, q_n\}$.

Konstruiere einen regulären Ausdruck, der $L(M)$ erzeugt.

Definiere $L_{ij}^k \subseteq \Sigma^*$ für $1 \leq i, j \leq n$, $0 \leq k \leq n$:

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

$$L_{ij}^k := L_{ij}^{k-1} \cup L_{ik}^{k-1} (L_{kk}^{k-1})^* L_{kj}^{k-1} \text{ für } k > 0$$

Spätere Behauptung:

$$L(M) = \bigcup_{q_j \in F} L_{1j}^n$$

$L_{11}^0 : \epsilon$	$L_{21}^0 : \emptyset$	$L_{31}^0 : \emptyset$	$L_{41}^0 : \emptyset$
$L_{12}^0 : a$	$L_{22}^0 : a + \epsilon$	$L_{32}^0 : a$	$L_{42}^0 : b$
$L_{13}^0 : b$	$L_{23}^0 : \emptyset$	$L_{33}^0 : \epsilon$	$L_{43}^0 : a$
$L_{14}^0 : \emptyset$	$L_{24}^0 : b$	$L_{34}^0 : b$	$L_{44}^0 : \epsilon$

$$L_{ij}^1 = L_{ij}^0$$

$L_{11}^2 : \epsilon$	$L_{21}^2 : \emptyset$	$L_{31}^2 : \emptyset$	$L_{41}^2 : \emptyset$
$L_{12}^2 : a^+$	$L_{22}^2 : a^*$	$L_{32}^2 : a^+$	$L_{42}^2 : b\,a^*$
$L_{13}^2 : b$	$L_{23}^2 : \emptyset$	$L_{33}^2 : \epsilon$	$L_{43}^2 : a$
$L_{14}^2 : a^+\,b$	$L_{24}^2 : a^*\,b$	$L_{34}^2 : a^*\,b$	$L_{44}^2 : \epsilon + b\,a^*\,b$

$L_{11}^3 : \epsilon$	$L_{21}^3 : \emptyset$	$L_{31}^3 : \emptyset$	$L_{41}^3 : \emptyset$
$L_{12}^3 : a^+ + b\,a^+$	$L_{22}^3 : a^*$	$L_{32}^3 : a^+$	$L_{42}^3 : b\,a^* + a^+\,b$
$L_{13}^3 : b$	$L_{23}^3 : \emptyset$	$L_{33}^3 : \epsilon$	$L_{43}^3 : a$
$L_{14}^3 : a^+\,b + b\,a^*\,b$	$L_{24}^3 : a^*\,b$	$L_{34}^3 : a^*\,b$	$L_{44}^3 : \epsilon + b\,a^*\,b + a^+\,b$

$$L(M) = L_{14}^4 = L_{14}^3 \cup L_{14}^3\,(L_{44}^3)^*\,L_{44}^3 = (a^+\,b + b\,a^*\,b)^+$$

DFAs erkennen reguläre Sprachen

Satz 2.2.3

M ein DFA $\Rightarrow L(M)$ regulär

Beweis

Idee: $w \in L_{ij}^k$ genau dann wenn

1. $\hat{\delta}(q_i, w) = q_j$ und
2. $\hat{\delta}(q_i, u) = q_m$ mit $m \leq k$ für alle $u \sqsubseteq w$, $u \neq \epsilon$, $u \neq w$ ($u \sqsubseteq v$ gdw. $u\,x = v$ für ein x)

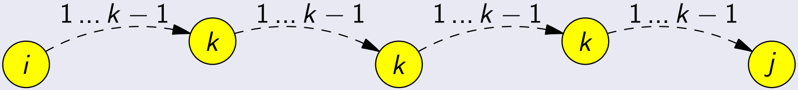
Anschaulich:

1. w bringt M von q_i nach q_j.
2. Dazwischen durchläuft M nur Zustände aus $\{q_1, \ldots, q_k\}$.

Beweis.

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

$$L_{ij}^{k} := L_{ij}^{k-1} \cup L_{ik}^{k-1} (L_{kk}^{k-1})^{*} L_{kj}^{k-1} \text{ für } k > 0$$



Korrektheit: Induktion über k.

Es ist leicht, reguläre Ausdrücke für L_{ij}^{k} zu finden.

Regulärer Ausdruck für $L(M) = \bigcup_{q_j \in F} L_{1j}^{n}$. □

Der Komplementärautomat

Satz 2.2.4

Es sei $M = (Q, \Sigma, \delta, q_0, F)$ *ein DFA.*
Dann ist $\Sigma^* \setminus L(M)$ *regulär.*

$\Sigma^* \setminus L$ *ist das Komplement von* L*.*

Beweis.

$M' := (Q, \Sigma, \delta, q_0, Q \setminus F)$.
Es gilt $L(M') = \Sigma^* \setminus L(M)$:

$$\hat{\delta}(q_0, w) \in F \Leftrightarrow \hat{\delta}(q_0, w) \notin Q \setminus F$$

$\square$

Der Produktautomat

Definition 2.2.5

Es seien $M' = (Q', \Sigma, \delta', q_0', F')$ *und* $M'' = (Q'', \Sigma, \delta'', q_0'', F'')$ *zwei DFAs.*

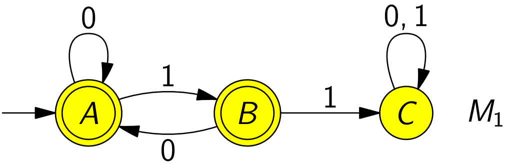
Wir definieren den Produktautomaten $M = M' \times M''$:

$M = (Q' \times Q'', \Sigma, \delta, (q_0', q_0''), F' \times F'')$
mit $\delta((q, p), a) = (\delta'(q, a), \delta''(p, a))$.

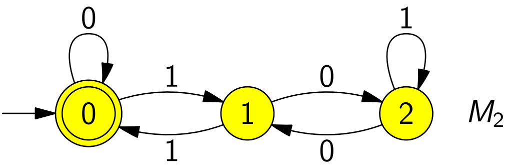
Beispiel

Konstruiere DFA für Sprache aller w mit:

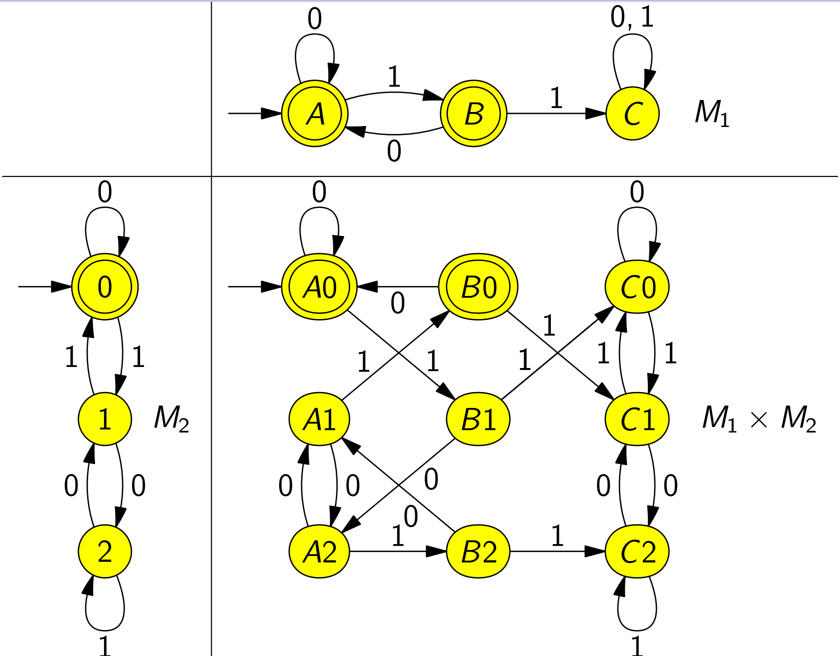
1. Es kommt 11 nicht als Unterwort in w vor.



2. Als Binärzahl ist w durch drei teilbar.



Konstruiere $M_1 \times M_2$!


A
B
C
0
1
0, 1
M_1
0
1
2
M_2
A0
B0
C0
A1
B1
C1
A2
B2
C2
$M_1 \times M_2$

Satz 2.2.6

Wenn M' und M'' DFAs sind und $M = M' \times M''$, dann $L(M) = L(M') \cap L(M'')$.

Beweis.

$\hat{\delta}((q_0', q_0''), w) = (\hat{\delta}'(q_0', w), \hat{\delta}''(q_0'', w))$
(Induktion über $|w|$)
und damit

$$\begin{aligned} w \in L(M) &\Leftrightarrow \hat{\delta}((q_0', q_0''), w) \in F' \times F'' \\ &\Leftrightarrow (\hat{\delta}'(q_0', w), \hat{\delta}''(q_0'', w)) \in F' \times F'' \\ &\Leftrightarrow \hat{\delta}'(q_0', w) \in F' \text{ und } \hat{\delta}''(q_0'', w) \in F'' \\ &\Leftrightarrow w \in L(M') \text{ und } w \in L(M''). \end{aligned}$$

Daher ist $L(M) = L(M') \cap L(M'')$. $\square$

Einige Vorteile endlicher deterministischer Automaten:

- durch Computer *schnell* simulierbar
- wenig Speicher benötigt:
  Tabelle für δ (read-only), aktueller Zustand
- Eingabe kann vergessen werden, nur von links nach rechts lesen
- Sie können schön visualisiert werden
- Sie können automatisch generiert werden (z.B. lex, egrep)