Der Produktautomat

Definition

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

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

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

Theorem

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

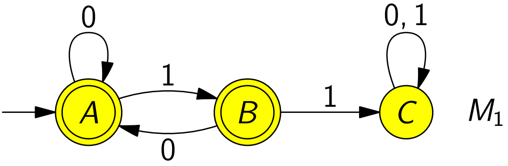
$$\begin{aligned} w \in L(M) &\Leftrightarrow \hat{\delta}((q_0', q_0''), w) \in F' \times F'' \\ &\Leftrightarrow (\hat{\delta}'(q_0', w), \hat{\delta}''(q_0'', w)) \in F' \times F'' \\ &\Leftrightarrow \hat{\delta}'(q_0', w) \in F' \text{ und } \hat{\delta}''(q_0'', w) \in F'' \\ &\Leftrightarrow w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

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

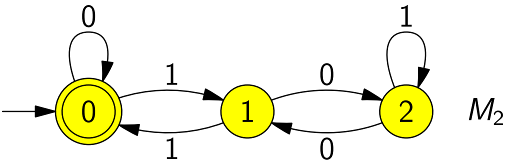
Beispiel

Konstruiere DFA für Sprache aller w mit:

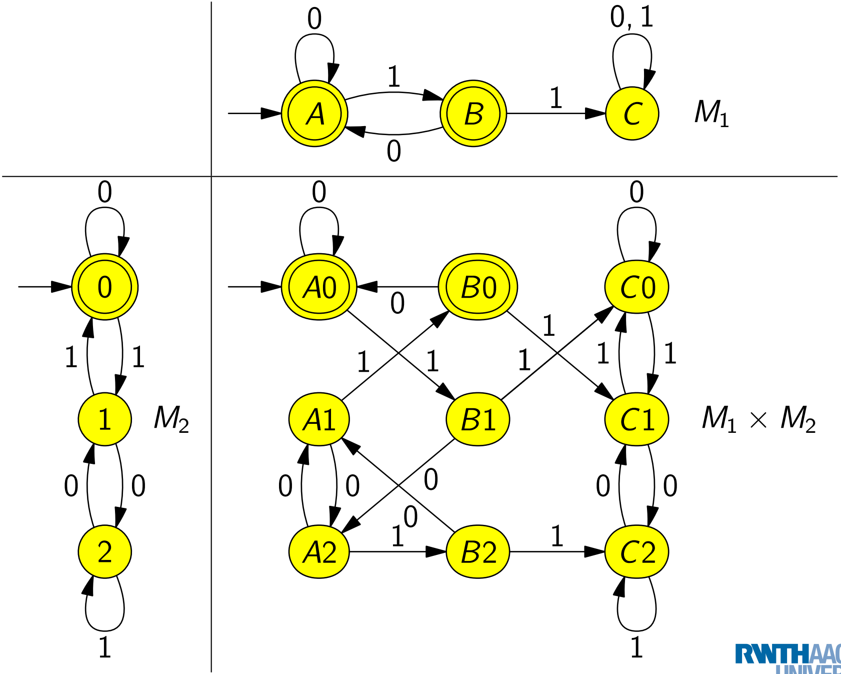
1. Es kommt 11 nicht als Unterwort in w vor.



2. Als Binärzahl ist w durch drei teilbar.



Konstruiere $M_1 \times M_2$!

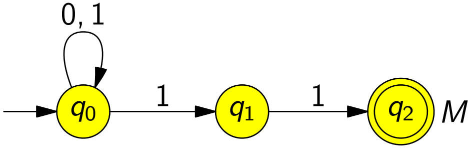


A
B
C
M_1
0
1
2
M_2
A0
B0
C0
A1
B1
C1
A2
B2
C2
$M_1 \times M_2$

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)

Nichtdeterministische endliche Automaten (NFAs)



Dies ist kein DFA!

1. Zwei Transitionen mit 1 aus q_0
2. Keine Transition mit 0 aus q_1

Welche Sprache soll M erkennen?

Nichtdeterministische endliche Automaten (NFAs)

Definition

Ein NFA ist ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$ mit

- Q Menge der *Zustände*
- Σ *Eingabealphabet*
- $\delta\colon Q \times \Sigma \to 2^Q$ *Übergangsfunktion*
- $q_0 \in Q$ *Startzustand*
- $F \subseteq Q$ *Endzustände*

Definition

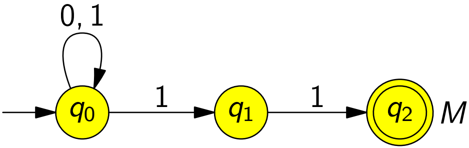
Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein NFA.

$\hat{\delta}\colon Q \times \Sigma^* \to 2^Q$ definiert durch

- $\hat{\delta}(q, \epsilon) = \{q\}$
- $\hat{\delta}(q, wa) = \{p \mid \text{es gibt } r \in \hat{\delta}(q, w) \text{ und } p \in \delta(r, a)\}$

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

Beispiel



- $\delta(q_0, 0) = \{q_0\}$
- $\delta(q_0, 1) = \{q_0, q_1\}$
- $\hat{\delta}(q_0, 010110101101) = \{q_0, q_1\}$
- $\hat{\delta}(q_0, 11111) = \{q_0, q_1, q_2\}$

$L(M) = (0 + 1)^*11$

Der Potenzautomat

Definition

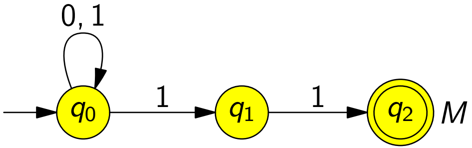
Sei M ein NFA, $M = (Q, \Sigma, \delta, q_0, F)$

Der zugehörige *Potenzautomat* M' ist so aufgebaut:

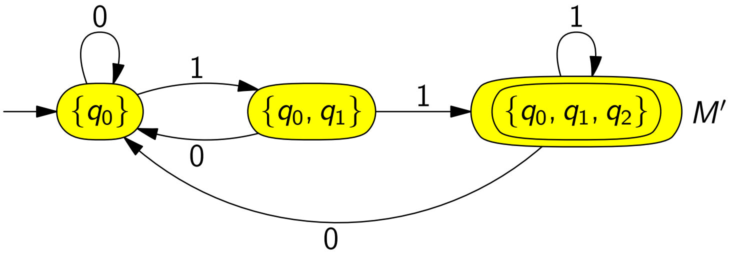
- $M' = (2^Q, \Sigma, \delta', \{q_0\}, F')$ mit
- $\delta' : 2^Q \times \Sigma \to 2^Q, (S, a) \mapsto \bigcup_{q \in S} \delta(q, a)$
- $F' = \{S \subseteq Q \mid S \cap F \neq \emptyset\}$

Der Potenzautomat ist ein DFA!

Beispiel



Der Potenzautomat hat die Zustände $\emptyset$, $\{q_0\}$, $\{q_1\}$, $\{q_2\}$, $\{q_0, q_1\}$, $\{q_0, q_2\}$, $\{q_1, q_2\}$ und $\{q_0, q_1, q_2\}$ und sieht so aus:



Nichterreichbare Zustände weggelassen!

Theorem

Zu jedem NFA M gibt es einen DFA M' mit $L(M) = L(M')$

Beweis.

$L(M) = L(M')$ für den Potenzautomaten M':

- $M = (Q, \Sigma, \delta, q_0, F)$
- $M' = (2^Q, \Sigma, \delta', \{q_0\}, F')$ mit
- $\delta' : 2^Q \times \Sigma \to 2^Q, (S, a) \mapsto \bigcup_{q \in S} \delta(q, a)$
- $F' = \{S \subseteq Q \mid S \cap F \neq \emptyset\}$

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

Daher:

$\hat{\delta}'(\{q_0\}, w) \in F' \iff \hat{\delta}(q_0, w) \cap F \neq \emptyset$ $\square$

Vergleich: DFA und NFA

Vorteile eines DFA:

- Effizient simulierbar

Vorteile eines NFA:

- Oft kleiner als DFA
- Einfacher zu entwerfen
- Halbwegs effizient simulierbar