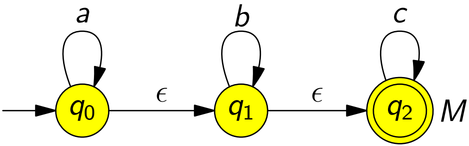
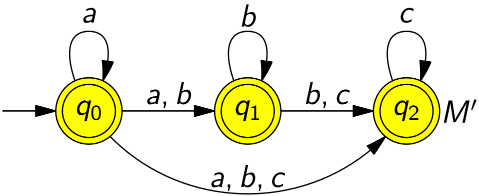
NFAs mit ϵ-Übergängen



Dies ist kein NFA!

Ziel: Erkenne die Sprache $a^* b^* c^*$.



NFA ist komplizierter!

Definition

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

1. $\delta\colon Q \times (\Sigma \cup \{\epsilon\}) \to 2^Q$,
2. Q, Σ, q_0, F wie bei NFAs.

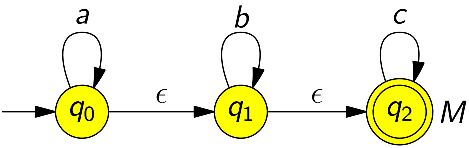
Für $q \in Q$:

$$\epsilon\text{-Hülle}(q) := \{\, p \in Q \mid \text{es gibt } q_1, \ldots, q_n \text{ mit } q_{i+1} \in \delta(q_i, \epsilon) \text{ für } 1 \leq i < n \text{ und } q = q_1,\ p = q_n \,\}$$

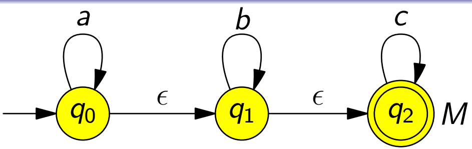
Für $S \subseteq Q$:

$$\epsilon\text{-Hülle}(S) := \bigcup_{q \in S} \epsilon\text{-Hülle}(q)$$

Beispiel



- ϵ-Hülle$(q_0) = \{q_0, q_1, q_2\}$
- ϵ-Hülle$(q_1) = \{q_1, q_2\}$
- ϵ-Hülle$(q_2) = \{q_2\}$
- ϵ-Hülle$(\{q_1, q_2\}) = \{q_1, q_2\}$



Definition

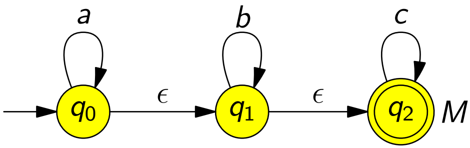
Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein NFA mit ϵ-Übergängen.
Es sei $q \in Q$, $w \in \Sigma^*$ und $a \in \Sigma$.

- $\hat{\delta}(q, \epsilon) = \epsilon\text{-Hülle}(q)$
- $\hat{\delta}(q, wa) = \displaystyle\bigcup_{p \in \hat{\delta}(q,w)} \epsilon\text{-Hülle}(\delta(p, a))$

Informell:
$\hat{\delta}(q, a)$ sind Zustände, die von q erreichbar sind:

1. Zunächst über ϵ-Transitionen
2. Dann über eine a-Transistion
3. Dann über ϵ-Transitionen

Beispiel



- $\delta(q_0, a) = \{q_0\}$
- $\hat{\delta}(q_0, a) = \{q_0, q_1, q_2\}$
- $\delta(q_0, b) = \emptyset$
- $\hat{\delta}(q_0, b) = \{q_1, q_2\}$
- $\delta(q_0, \epsilon) = \{q_1\}$
- $\hat{\delta}(q_0, \epsilon) = \{q_0, q_1, q_2\}$

Theorem

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein NFA mit ϵ-Übergängen.
Dann gibt es einen NFA M' mit $L(M') = L(M)$.

Beweis.

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

- $\delta'(q, a) = \hat{\delta}(q, a)$,
- $F' = \{ q \in Q \mid \epsilon\text{-Hülle}(q) \cap F \neq \emptyset \}$.

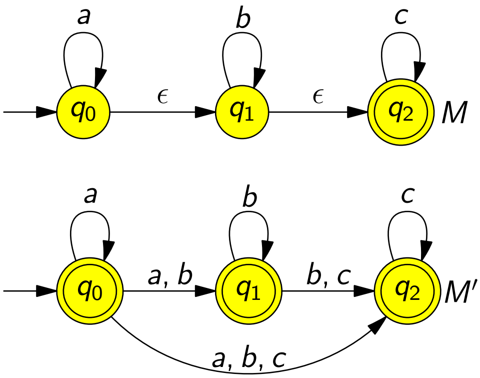
□

Informell:

$p \in \delta'(q, a)$ gdw. in M gibt es Pfad von q nach p, der

1. zunächst mit ϵ beschriftet ist,
2. dann einen a-Übergang hat,
3. dann wieder mit ϵ beschriftet ist.

Beispiel

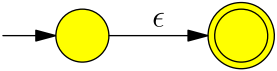


Die Thompson-Konstruktion

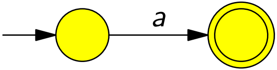
Gegeben regulärer Ausdruck r.
Konstruktion eines NFA M mit $L(M) = L(r)$.
Vorgehen: Induktiv über Aufbau von r.

- $r = \emptyset$:

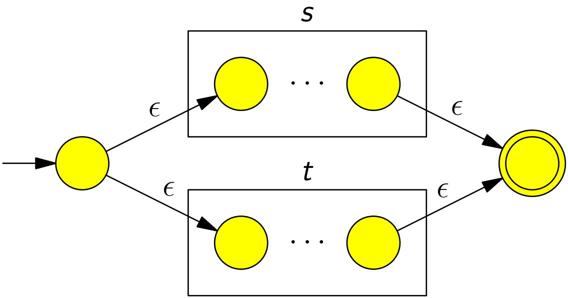
- $r = \epsilon$:



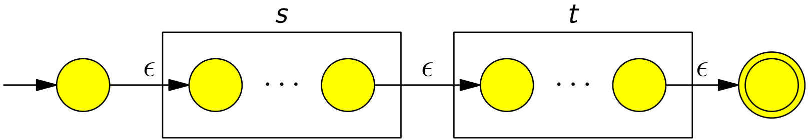
- $r = a$:



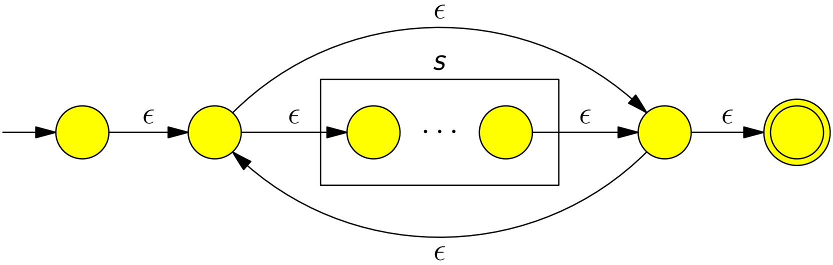
- $r = s + t$:



- $r = st$:



- $r = s^*$:



Theorem

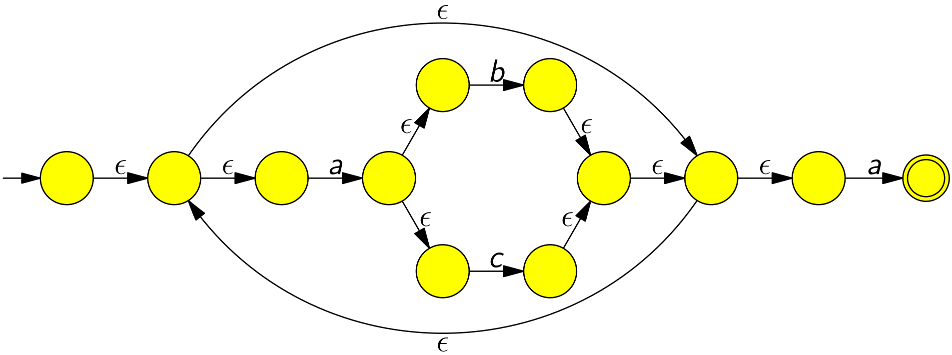
Zu jedem regulären Ausdruck r gibt es einen NFA mit ϵ-Kanten M, so daß $L(M) = L(r)$.

Beweis.

Thompson-Konstruktion.

Korrektheit:
Strukturelle Induktion über den Aufbau regulärer Ausdrücke. $\square$

Beispiel



$$(a(b+c))^*a$$

Größe des NFA linear in der Länge des regulären Ausdrucks!

Robustheit regulärer Sprachen

Theorem

DFAs, NFAs, NFAs mit ϵ-Übergängen und reguläre Ausdrücke charaktersieren jeweils die regulären Sprachen.

Beweis.

1. regulärer Ausdruck $\rightarrow$ ϵ-NFA: Thompson-Konstruktion
2. ϵ-NFA $\rightarrow$ NFA: Eliminierung von ϵ-Kanten
3. NFA $\rightarrow$ DFA: Potenzautomat
4. DFA $\rightarrow$ regulärer Ausdruck: L_{ij}^{k}-Konstruktion

$\square$

Robustheit regulärer Sprachen

Theorem

Die Reguläre Sprachen sind abgeschlossen unter Vereinigung, Schnitt, Konkatenation, Kleene'scher Hülle, Komplement, Differenz und Homomorphismen.

- Vereinigung: Reguläre Ausdrücke
- Schnitt: DFAs, Produktautomat
- Konkatenation: Reguläre Ausdrücke
- Kleene'sche Hülle: Reguläre Ausdrücke
- Komplement: DFAs
- Differenz: Komplement und Schnitt
- Homomorphismen: Reguläre Ausdrücke

Simulation eines NFA

```
S := { q0};
while(es gibt noch ein Zeichen) {
  c := lese Zeichen;
  H := ∅;
  for(q in S) { H := H ∪ δ(q, c); }
  S := H;
}
if(S ∩ F ≠ ∅) return 1;
return 0;
```

Datenstruktur für H:

- Stack (FIFO-Queue) und
- Bitfeld

Laufzeit: $O(|Q| \cdot |w|)$, falls $|\Sigma|$ konstant.

Einige Zwischenfragen

Welche Konstruktionen funktionieren auch für NFAs?

1. Komplementärautomat Nein
2. Produktautomat Ja
3. L_{ij}^{k}-Konstruktion Ja

Wer hat die Nase vorne? NFA oder DFA?

1. Vereinigung zweier Sprachen NFA
2. Schnitt zweier Sprachen DFA
3. Konstruktion aus einem regulären Ausdruck NFA
4. Verwandeln in einen regulären Ausdruck egal
5. Komplementieren DFA
6. Simulieren DFA
7. Größe NFA

Die Myhill–Nerode-Relation $\equiv_L$

Definition

Es sei $L \subseteq \Sigma^*$.

Definiere $\equiv_L \subseteq \Sigma^* \times \Sigma^*$ vermöge

$$u \equiv_L v \iff uw \in L \Leftrightarrow vw \in L \text{ für alle } w \in \Sigma^*.$$

Der *Index* einer Äquivalenzrelation ist die Anzahl ihrer Äquivalenzklassen.

Interessanter Fall: $\equiv_L$ hat endlichen Index.

Beispiel 1

Es sei $L = 0^*1^*$.

- $001 \equiv_L 0111$
- $010 \not\equiv_L 0111$, denn $010 \notin L$, $0111 \in L$.
- $00 \not\equiv_L 00001$, denn $000 \in L$, $000010 \notin L$.

Wieviele Äquivalenzklassen hat $\equiv_L$?

Drei:

1. 0^*
2. 0^*1^+
3. $0^*1^+0(0+1)^*$

Beispiel 2

Was ist der Index von $\equiv_L$ für diese Sprachen?

1. $L = \{0, 1\}^*$
2. $L = \{\, a^p \mid p \text{ ist eine Primzahl} \,\}$
3. $L = \emptyset$
4. $L = \{\, w \in \{a, b, c\}^* \mid |w| \text{ ist Vielfaches von } 7 \,\}$
5. $L = \{3, 3., 3.1, 3.14, 3.141, 3.1415, 3.14159, \dots\}$
6. $L = \{\, a^n b^n \mid n \geq 0 \,\}$
7. $L = \{\, a^n b^m \mid n \geq m \geq 0 \,\}$
8. $L = \{\, a^n b^m \mid |n - m| < 5 \,\}$

Lemma (A)

$L \subseteq \Sigma^*$ regulär $\Longrightarrow$ $\equiv_L$ hat endlichen Index.

Beweis.

1. L regulär und $L = L(M)$ mit DFA $M = (Q, \Sigma, \delta, q_0, F)$.
2. Definiere $u \sim v \iff \hat{\delta}(q_0, u) = \hat{\delta}(q_0, v)$.
3. $u \sim v \Rightarrow u \equiv_L v$, denn $uw \in L \iff vw \in L$ falls $\hat{\delta}(q_0, u) = \hat{\delta}(q_0, v)$.
4. Also hat $\sim$ mindestens so viele Äquivalenzklassen wie $\equiv_L$.
5. $\sim$ hat aber endlichen Index.

$\square$

Lemma (B)

$L \subseteq \Sigma^*$ *regulär* $\Longleftarrow$ $\equiv_L$ *hat endlichen Index.*

Beweis.

1. $L \subseteq \Sigma^*$ und Index von $\equiv_L$ sei endlich.
2. Konstruiere $M = (Q, \Sigma, \delta, [\epsilon]_{\equiv_L}, F)$ mit
   - $Q = \{ [w]_{\equiv_L} \mid w \in \Sigma^* \}$
   - $\delta \colon Q \times \Sigma \to Q, ([w]_{\equiv_L}, a) \mapsto [wa]_{\equiv_L}$
   - $F = \{ [w]_{\equiv_L} \mid w \in L \}$
3. Q endlich, da Index von $\equiv_L$ endlich.
4. δ wohldefiniert, da $[u]_{\equiv_L} = [v]_{\equiv_L} \Rightarrow [ua]_{\equiv_L} = [va]_{\equiv_L}$
5. $L(M) = L$, da $\hat{\delta}([\epsilon]_{\equiv_L}, w) = [w]_{\equiv_L}$.

$\square$

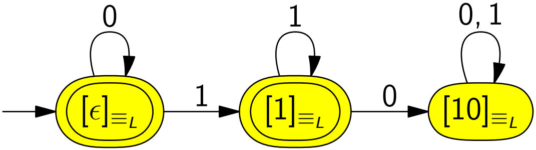
Beispiel

Es sei $L = 0^*1^*$.

$\equiv_L$ hat die Äquivalenzklassen

1. $[\epsilon]_{\equiv_L} = 0^*$,
2. $[1]_{\equiv_L} = 0^*1^+$ und
3. $[10]_{\equiv_L} = 0^*1^+0(0+1)^*$.

Der Myhill–Nerode–Automat:



Der Satz von Myhill–Nerode

Theorem

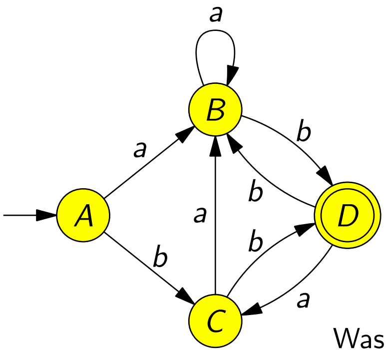
1. *$L \subseteq \Sigma^*$ ist genau dann regulär, wenn $\equiv_L$ endlichen Index hat.*
2. *M ein DFA $\Longrightarrow \sim_M$ ist eine Verfeinerung von $\equiv_{L(M)}$.*
3. *Es gibt zu jeder regulären Sprache $L \in \Sigma^*$ einen bis auf Isomorphie eindeutigen DFA $M = (Q, \Sigma, \delta, q_0, F)$ mit $L = L(M)$.*

Beweis.

1. Folgt aus Lemma A und B.
2. Beweis von Lemma A: $u \sim v \Rightarrow u \equiv_L v$.
3. Da $\sim$ eine Verfeinerung von $\equiv_L$ ist, muß $\sim = \equiv_L$ gelten, wenn ihre Indexe gleich sind.

$\square$

Beispiel



Was sind die Äquivalenzklassen von $\sim$?

Natürlich $[\epsilon]_\sim$, $[a]_\sim$, $[b]_\sim$ und $[ab]_\sim \ldots$

Was sind die Äquivalenzklassen von $\equiv_{L(M)}$?

Es sind $[\epsilon]_\sim$, $[a]_\sim \cup [b]_\sim$ und $[ab]_\sim$.