

# Konečné automaty

**Definice. Deterministický konečný automat** (Deterministic Finite Automaton, DFA)  $\mathcal{M}$  je pětice  $(Q, \Sigma, \delta, q_0, F)$ , kde

- $Q$  je neprázdná konečná množina **stavů**,
- $\Sigma$  je konečná **vstupní abeceda**,
- $\delta : Q \times \Sigma \rightarrow Q$  je **parciální přechodová funkce**,
- $q_0 \in Q$  je **počáteční (iniciální) stav**,
- $F \subseteq Q$  je množina **koncových (akceptujících) stavů**.

# Příklad a zápis tabulkou

# Zápis grafem

# Výpočet konečného automatu

**Rozšířená přechodová funkce**  $\hat{\delta} : Q \times \Sigma^* \rightarrow Q$  je parciální funkce definovaná induktivně vzhledem k délce slova ze  $\Sigma^*$ :

- $\hat{\delta}(q, \varepsilon) = q$  pro každý stav  $q \in Q$
- $\hat{\delta}(q, wa) = \begin{cases} \delta(\hat{\delta}(q, w), a) & \text{je-li } \hat{\delta}(q, w) \text{ i } \delta(\hat{\delta}(q, w), a) \text{ definováno} \\ \perp & \text{jinak} \end{cases}$

Slovo  $w$  je **akceptováno** automatem  $\mathcal{M}$  právě když  $\hat{\delta}(q_0, w) \in F$ .

Slovo  $w$  je **zamítáno** automatem  $\mathcal{M}$  právě když  $\hat{\delta}(q_0, w) \notin F$ .

Jazyk **přijímaný** (**akceptovaný**, **rozpoznávaný**) automatem  $\mathcal{M}$  je

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

Jazyk, který je rozpoznatelný (nějakým) deterministickým konečným automatem, nazveme **regulární**.

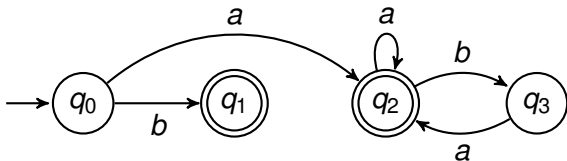
Automaty  $\mathcal{M}$  a  $\mathcal{M}'$  nazveme **ekvivalentní**, právě když  $L(\mathcal{M}) = L(\mathcal{M}')$ .

# Parcialita přechodové funkce

Přechodová funkce  $\delta$  byla zavedena jako parciální. Parcialita přechodové funkce nemá podstatný vliv na výpočetní sílu konečných automatů.

**Lemma.** Ke každému DFA  $\mathcal{M}$  existuje ekvivalentní DFA  $\mathcal{M}'$  s totální přechodovou funkcí.

Idea důkazu:



**Důkaz.** Necht'  $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ . Pak  $\mathcal{M}'$  definujeme předpisem  $\mathcal{M}' = (Q \cup \{p\}, \Sigma, \delta', q_0, F)$ , kde  $p \notin Q$  a

$$\delta'(q, a) = \begin{cases} \delta(q, a) & \text{je-li } \delta(q, a) \text{ definováno,} \\ p & \text{jinak.} \end{cases}$$

Zejména  $\delta'(p, a) = p$  pro každé  $a \in \Sigma$ .

Důkaz korektnosti:

- $\mathcal{M}'$  má totální přechodovou funkci – zřejmé z definice  $\mathcal{M}'$ .
- $\mathcal{M}$  a  $\mathcal{M}'$  jsou ekvivalentní – dokážeme.

Indukcí k délce slova ověříme, že pro každé  $q \in Q$  a  $w \in \Sigma^*$  platí

$$\hat{\delta}'(q, w) = \begin{cases} \hat{\delta}(q, w) & \text{je-li } \hat{\delta}(q, w) \text{ definováno,} \\ p & \text{jinak.} \end{cases}$$

Jelikož  $p \notin F$ , platí  $L(\mathcal{M}) = L(\mathcal{M}')$ . □



# Konstrukce konečných automatů

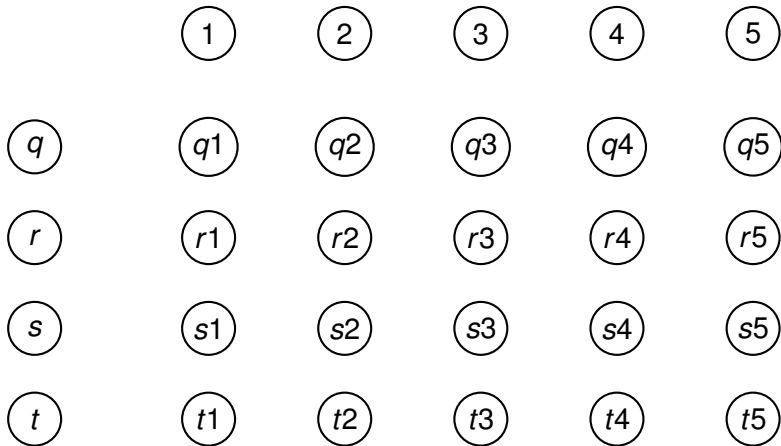
Máme za úkol sestrojít automat rozpoznávající jazyk

$$L = \{w \in \{a, b\}^* \mid w \text{ obsahuje podslovo } abaa\}$$

Označení stavů automatu zvolíme tak, aby bylo patrné, jaká část požadovaného podslova *abaa* již byla automatem přečtena:

# Příklad

$\{w \in \{a, b\}^* \mid w \text{ obsahuje podslovo } abaa \wedge (w = b \vee w \text{ začíná i končí na } a \text{ a mezi dvěma výskyty } b \text{ je vždy alespoň jedno } a)\}$



# Synchronní paralelní kompozice automatů

Pro dané automaty  $\mathcal{M}_1$  a  $\mathcal{M}_2$  umožňuje sestavit automat rozpoznávající **průnik (sjednocení, rozdíl)** jazyků  $L(\mathcal{M}_1)$  a  $L(\mathcal{M}_2)$ .

Nechť  $\mathcal{M}_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ ,  $\mathcal{M}_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$  a **přechodové funkce  $\delta_1, \delta_2$  jsou totální.**

Definujeme DFA  $\mathcal{M}_3 = (Q_3, \Sigma, \delta_3, q_3, F_3)$ , kde

- $Q_3 = Q_1 \times Q_2 = \{(p, q) \mid p \in Q_1, q \in Q_2\}$
- $F_3 = F_1 \times F_2 = \{(p, q) \mid p \in F_1, q \in F_2\}$
- $q_3 = (q_1, q_2)$
- $\delta_3((p, q), a) = (\delta_1(p, a), \delta_2(q, a))$

**Tvrzení.**  $L(\mathcal{M}_3) = L(\mathcal{M}_1) \cap L(\mathcal{M}_2)$

**Důkaz.** Nejprve dokážeme toto tvrzení:

$$\widehat{\delta}_3((q_1, q_2), w) = (p, q) \iff \widehat{\delta}_1(q_1, w) = p \wedge \widehat{\delta}_2(q_2, w) = q$$

Důkaz se provede indukcí vzhledem k  $|w|$ .

■ **Základní krok**  $|w| = 0$ :

Z definice  $\widehat{\delta}_3((q_1, q_2), \varepsilon) = (q_1, q_2)$ ,  $\widehat{\delta}_1(q_1, \varepsilon) = q_1$ ,  $\widehat{\delta}_2(q_2, \varepsilon) = q_2$ .

Pro  $w = \varepsilon$  je tedy ekvivalence platná.

■ **Indukční krok:** Necht'  $w = va$ , kde  $v \in \Sigma^*$ ,  $a \in \Sigma$ . Platí:

$$\widehat{\delta}_3((q_1, q_2), va) = (p, q) \iff$$

$$\widehat{\delta}_3((q_1, q_2), v) = (r, s) \wedge \delta_3((r, s), a) = (p, q) \iff$$

$$\widehat{\delta}_1(q_1, v) = r \wedge \widehat{\delta}_2(q_2, v) = s \wedge \delta_1(r, a) = p \wedge \delta_2(s, a) = q \iff$$

$$\widehat{\delta}_1(q_1, va) = p \wedge \widehat{\delta}_2(q_2, va) = q$$

Nyní již lze snadno dokázat vlastní tvrzení věty:

$$w \in L(\mathcal{M}_3) \iff$$

$$\widehat{\delta}_3((q_1, q_2), w) = (p, q) \text{ kde } p \in F_1 \text{ a } q \in F_2 \iff$$

$$\widehat{\delta}_1(q_1, w) = p \wedge \widehat{\delta}_2(q_2, w) = q \iff$$

$$w \in L(\mathcal{M}_1) \cap L(\mathcal{M}_2)$$



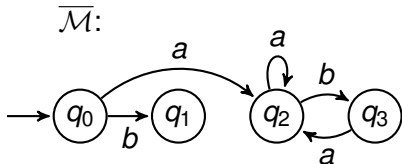
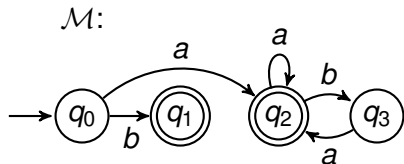
Modifikace pro **sjednocení**, tj.  $L(\mathcal{M}_3) = (\mathcal{M}_1) \cup L(\mathcal{M}_2)$ :

**DŮ:** Modifikujte konstrukci tak, aby platilo  $L(\mathcal{M}_3) = L(\mathcal{M}_1) \setminus L(\mathcal{M}_2)$ .

# Automat pro komplement

K automatu  $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$  s **totální přechodovou funkcí** sestrojíme automat  $\overline{\mathcal{M}}$  rozpoznávající jazyk  $co-L(\mathcal{M})$  jako

$$\overline{\mathcal{M}} = (Q, \Sigma, \delta, q_0, Q \setminus F).$$



# Limity konečných automatů

$$L = \{a^n b^n \mid n \geq 0\} = \{\varepsilon, ab, aabb, aaabbb, aaaabbbb \dots\}$$

*a a a a a b b b b b*

Předpokládejme, že existuje automat  $\mathcal{M}$  přijímající jazyk  $L$ .  
Nechť  $\mathcal{M}$  má  $k$  stavů.

Uvažme výpočet  $\mathcal{M}$  na slově  $a^n b^n$  kde  $n > k$ .

*aaaaaaaaaaaaaaaaaaaaa bbbbbbbbbbbbbbbbbbbbbb*

Protože  $n > k$ , musí existovat (z Dirichletova principu) stav  $p$  takový, že při čtení iniciální posloupnosti symbolů  $a$  projde automat stavem  $p$  (alespoň) dvakrát.



*aaaaaaaaa aaaaaaaaa aaaabbbbbbbbbbbbbbbbbbbbbb*

Platí  $\hat{\delta}(q_0, x) = p \quad \hat{\delta}(p, y) = p \quad \hat{\delta}(p, z) = r \in F$

Pak ale  $\hat{\delta}(q_0, xz) = \hat{\delta}(\hat{\delta}(q_0, x), z) = \hat{\delta}(p, z) = r \in F$

*aaaaaaaaa aaaabbbbbbbbbbbbbbbbbbbbbb*

Analogicky můžeme „vsunout“ slovo  $y$ :

$$\begin{aligned}\hat{\delta}(q_0, xyxyz) &= \hat{\delta}(\hat{\delta}(\hat{\delta}(\hat{\delta}(\hat{\delta}(q_0, x), y), y), y), y), z) \\ &= \hat{\delta}(\hat{\delta}(\hat{\delta}(\hat{\delta}(p, y), y), y), z) \\ &= \hat{\delta}(\hat{\delta}(\hat{\delta}(p, y), y), z) \\ &= \hat{\delta}(\hat{\delta}(p, y), z) \\ &= \hat{\delta}(p, z) \\ &= r \in F\end{aligned}$$

## Lemma. [o vkládání, pumping lemma]

Nechť  $L$  je regulární jazyk.

Pak existuje  $n \in \mathbb{N}$  takové, že libovolné slovo  $w \in L$  délky alespoň  $n$  lze psát ve tvaru  $w = xyz$ , kde  $|xy| \leq n$ ,  $y \neq \varepsilon$  a  $xy^iz \in L$  pro každé  $i \in \mathbb{N}_0$ .

**Důkaz.** Nechť DFA  $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$  rozpoznává jazyk  $L$ .

Položme  $n = \text{card}(Q)$ .

Pro libovolné slovo  $w \in L$  délky alespoň  $n$  platí, že automat  $\mathcal{M}$  projde při akceptování slova  $w$  (alespoň) dvakrát stejným stavem.

Slovo  $w$  tedy můžeme rozdělit na tři části:  $w = xyz$ , kde  $y \neq \varepsilon$  a  $\hat{\delta}(q_0, x) = p$ ,  $\hat{\delta}(p, y) = p$  a  $\hat{\delta}(p, z) = r \in F$ . Je zřejmé, že ke zopakování nějakého stavu dojde nejpozději po zpracování prvních  $n$  znaků a tedy dostáváme  $|xy| \leq n$ .

Dále  $\hat{\delta}(p, y^i) = p$  pro libovolné  $i \in \mathbb{N}_0$ , proto také  $\hat{\delta}(q_0, xy^iz) = r$ , tj.  $xy^iz \in L(\mathcal{M})$  pro každé  $i \in \mathbb{N}_0$ . □

$L$  je regulární  $\implies \exists n \in \mathbb{N}$ .

$$\forall w \in L. (|w| \geq n \implies \\ \exists x, y, z. (w = xyz \wedge y \neq \varepsilon \wedge |xy| \leq n \wedge \\ \forall i \geq 0. xy^i z \in L))$$

**Pomocí Lemmatu lze dokázat, že nějaký jazyk není regulární.**

Nechť pro jazyk  $L$  platí:

- pro libovolné  $n \in \mathbb{N}$
- existuje takové slovo  $w \in L$  délky alespoň  $n$ , pro které platí, že
- při libovolném rozdělení slova  $w$  na takové tři části  $x, y, z$ , že  $|xy| \leq n$  a  $y \neq \varepsilon$ ,
- existuje alespoň jedno  $i \in \mathbb{N}_0$  takové, že  $xy^i z \notin L$ .

Pak z Lemma o vkládání plyne, že  $L$  není regulární.

# Příklad důkazu ne-regularity pomocí Lemmatu o vkládání

$$L = \{uc^m u^R \mid u \in \{a, b\}^*, m > 0\}$$