

Konstrukce minimálního konečného automatu

Definice 2.18. Nechť $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ je konečný automat. Stav $q \in Q$ nazveme **dosažitelný**, pokud existuje $w \in \Sigma^*$ takové, že $\hat{\delta}(q_0, w) = q$. Stav je **nedosažitelný**, pokud není dosažitelný.

Příklad

Algoritmus pro eliminaci nedosažitelných stavů FA

Vstup: Konečný automat $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$.

Výstup: Ekvivalentní automat \mathcal{M}' bez nedosažitelných stavů.

1 $i := 0$

2 $S_i := \{q_0\}$

3 **repeat** $S_{i+1} := S_i \cup \{q \mid \exists p \in S_i, a \in \Sigma : \delta(p, a) = q\}$

4 $i := i + 1$

5 **until** $S_i = S_{i-1}$

6 $Q' := S_i$

7 $\mathcal{M}' := (Q', \Sigma, \delta|_{Q'}, q_0, F \cap Q')$

Korektnost: algoritmus je správný a konečný.

Příklad

Eliminace ekvivalentních stavů

Nechť $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ je FA bez nedosažitelných stavů, jehož přechodová funkce je totální.

Definice 2.32. Stavy p, q nazveme **jazykově ekvivalentní**, psáno $p \equiv q$, pokud

$$p \equiv q \iff \forall w \in \Sigma^* : (\hat{\delta}(p, w) \in F \iff \hat{\delta}(q, w) \in F).$$

$$p \equiv q \iff \forall w \in \Sigma^* : (\hat{\delta}(p, w) \in F \iff \hat{\delta}(q, w) \in F)$$

Definice 2.34. **Reduktem** automatu $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ nazveme konečný automat $\mathcal{M}/\equiv = (Q/\equiv, \Sigma, \eta, [q_0], F/\equiv)$, kde:

- Stavy jsou třídy rozkladu Q/\equiv (třída obsahující stav q je $[q]$).
- Přejchodová funkce η je funkce splňující:

$$\forall p, q \in Q, \forall a \in \Sigma : \delta(q, a) = p \implies \eta([q], a) = [p].$$

- Počáteční stav je třída rozkladu Q/\equiv obsahující stav q_0 .
- Koncové stavy jsou právě ty třídy rozkladu Q/\equiv , které obsahují alespoň jeden koncový stav.

Věta 2.37. Nechť $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ je FA bez nedosažitelných stavů s totální přechodovou funkcí. Pak $L(\mathcal{M}) = L(\mathcal{M}/\equiv)$.

Algoritmus konstrukce minimálního automatu

Definice 2.38. Pro každé $i \in \mathbb{N}_0$ definujeme binární relaci \equiv_i na Q předpisem

$$p \equiv_i q \stackrel{\text{def}}{\iff} \forall w \in \Sigma^*. |w| \leq i : (\hat{\delta}(p, w) \in F \iff \hat{\delta}(q, w) \in F)$$

- $p \equiv_i q$ právě když p a q nelze “rozlišit” žádným slovem délky $\leq i$
- $p \equiv q$ právě když $p \equiv_i q$ pro každé $i \in \mathbb{N}_0$. ($\equiv = \bigcap_{i=0}^{\infty} \equiv_i$)
- 1. $\equiv_0 = \{(p, q) \mid p \in F \iff q \in F\}$
2. $\equiv_{i+1} = \{(p, q) \mid p \equiv_i q \wedge \forall a \in \Sigma : \delta(p, a) \equiv_i \delta(q, a)\}$

Algoritmus konstrukce minimálního automatu

Vstup: Konečný automat $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ bez nedosažitelných stavů s totální přechodovou funkcí.

Výstup: Redukt \mathcal{M}/\equiv .

1 $i := 0$

2 $\equiv_0 := \{(p, q) \mid p \in F \iff q \in F\}$

3 **repeat**

4 $\equiv_{i+1} := \{(p, q) \mid p \equiv_i q \wedge \forall a \in \Sigma : \delta(p, a) \equiv_i \delta(q, a)\}$

5 $i := i + 1$

6 **until** $\equiv_i = \equiv_{i-1}$

7 $\equiv := \equiv_i$

8 $\mathcal{M}/\equiv := (Q/\equiv, \Sigma, \eta, [q_0], F/\equiv)$

Korektnost algoritmu: důkaz vynechán.

Intuice

Příklad

	\mathcal{M}	a	b
\rightarrow	1	2	—
	2	3	4
\leftarrow	3	6	5
	4	3	2
\leftarrow	5	6	3
\leftarrow	6	2	—
	7	6	1

	\mathcal{M}'	a	b
\rightarrow	1	2	N
	2	3	4
\leftarrow	3	6	5
	4	3	2
\leftarrow	5	6	3
\leftarrow	6	2	N
	N	N	N

	\equiv_0	a	b
I	1	I	I
	2	II	I
	4	II	I
	N	I	I
II	3	II	II
	5	II	II
	6	I	I

Příklad

	\equiv_1	a	b
I	1	II	I
	N	I	I
II	2	III	II
	4	III	II
III	3	IV	III
	5	IV	III
IV	6	II	I

	\equiv_2	a	b
I	1	III	II
II	N	II	II
III	2	IV	III
	4	IV	III
IV	3	V	IV
	5	V	IV
V	6	III	II

	\mathcal{M}/\equiv	a	b
	I	III	II
	II	II	II
	III	IV	III
←	IV	V	IV
←	V	III	II

Kanonický tvar konečných automatů

Motivace

\mathcal{M}_1	a	c	b
I	IV	III	I
→ II	V	III	III
← III	II	I	I
← IV	V	I	II
V	II	V	V

\mathcal{M}_2	a	b	c
→ I	III	V	V
II	IV	II	V
III	I	III	III
← IV	III	I	II
← V	I	II	II

Kanonický tvar konečných automatů

	<i>a</i>	<i>c</i>	<i>b</i>
I	IV	III	I
→ II	V	III	III
← III	II	I	I
← IV	V	I	II
V	II	V	V

	<i>a</i>	<i>b</i>	<i>c</i>
→ A			
B			
C			
D			
E			

Rozšíření konečných automatů I

Nedeterministické konečné automaty

Definice 2.42. **Nedeterministický konečný automat (NFA)** je $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$, kde význam všech složek je stejný jako v definici FA s výjimkou přechodové funkce δ . Ta je definována jako (totální) zobrazení $\delta : Q \times \Sigma \rightarrow 2^Q$.

Rozšířená přechodová funkce $\hat{\delta} : Q \times \Sigma^* \rightarrow 2^Q$:

- $\hat{\delta}(q, \varepsilon) = \{q\}$
 - $\hat{\delta}(q, wa) = \bigcup_{p \in \hat{\delta}(q, w)} \delta(p, a)$
-

Jazyk přijímaný nedeterministickým konečným automatem \mathcal{M} je

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

Příklad

Ekvivalence DFA a NFA

Věta 2.43. Pro každý NFA $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ existuje ekvivalentní deterministický FA (DFA).

Důkaz. Definujeme deterministický FA $\mathcal{M}' = (Q', \Sigma, \delta', \{q_0\}, F')$

- $Q' = 2^Q$, tj. stavy automatu \mathcal{M}' jsou všechny podmnožiny Q .
- $\delta'(P, a) = \bigcup_{q \in P} \delta(q, a)$.
- Množina koncových stavů F' je tvořena právě těmi podmnožinami Q , které obsahují nějaký prvek množiny F .

Korektnost

- \mathcal{M}' je deterministický konečný automat.

- \mathcal{M} a \mathcal{M}' jsou ekvivalentní:

indukcí k délce $w \in \Sigma^*$ dokážeme: $\hat{\delta}(q_0, w) = \hat{\delta}'(\{q_0\}, w)$

Základní krok $|w| = 0$: Platí $\hat{\delta}(q_0, \varepsilon) = \{q_0\} = \hat{\delta}'(\{q_0\}, \varepsilon)$.

Indukční krok: Nechť $w = va$, pak

$$\begin{aligned}\hat{\delta}(q_0, va) &= \bigcup_{p \in \hat{\delta}(q_0, v)} \delta(p, a) = \delta'(\hat{\delta}(q_0, v), a) \text{ (viz definice } \delta') \\ &= \delta'(\hat{\delta}'(q_0, v), a) \text{ (indukční předpoklad)} = \hat{\delta}'(q_0, va).\end{aligned}$$

Pak $L(\mathcal{M}) = L(\mathcal{M}')$, neboť

$$w \in L(\mathcal{M}) \iff \hat{\delta}(q_0, w) \cap F \neq \emptyset \iff$$

$$\hat{\delta}'(\{q_0\}, w) \cap F \neq \emptyset \iff \hat{\delta}'(\{q_0\}, w) \in F' \iff w \in L(\mathcal{M}'). \quad \square$$

Algoritmus transformace NFA na ekvivalent. DFA

Vstup: Nedeterministický FA $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$.

Výstup: Ekvivalentní deterministický FA $\mathcal{M}' = (Q', \Sigma, \delta', \{q_0\}, F')$
bez nedosažitelných stavů a s totální přechodovou funkcí.

```
1  $Q' := \{\{q_0\}\}; \quad \delta' := \emptyset; \quad F' := \emptyset; \quad Done := \emptyset;$   
2 while  $(Q' \setminus Done) \neq \emptyset$  do  
3      $M :=$  libovolný prvek množiny  $Q' \setminus Done$   
4     if  $M \cap F \neq \emptyset$  then  $F' := F' \cup \{M\}$  fi  
5     foreach  $a \in \Sigma$  do  
6          $N := \bigcup_{p \in M} \delta(p, a)$   
7          $Q' := Q' \cup \{N\}$   
8          $\delta' := \delta' \cup \{((M, a), N)\}$  od  
9      $Done := Done \cup \{M\}$   
10 od  
11  $\mathcal{M}' := (Q', \Sigma, \delta', \{q_0\}, F')$ 
```

Důsledky determinizace konečných automatů

Věta 2.44. Pro každé $n \in \mathbb{N}$ existuje NFA o n stavech takový, že ekvivalentní DFA má i po minimalizaci 2^n stavů.

Důkaz: vynechán.