

# Konečné automaty

ZPRACUJE: Mystik

## Obsah

- 1 Konečný automat
  - 1.1 Definice (nedeterministického) KA (NKA)
  - 1.2 Varianty konečných automatů
    - 1.2.1 Rozšířený KA
    - 1.2.2 Deterministický konečný automat (DKA)
    - 1.2.3 Úplný DKA
    - 1.2.4 DKA bez nedosažitelných stavů
    - 1.2.5 Minimální/redukovaný DKA (MDKA)
- 2 Algoritmy
  - 2.1 Odstranění nedosažitelných stavů
  - 2.2 Minimalizace KA
  - 2.3 Převod NKA na DKA
  - 2.4 Konstrukce NKA k pravé regulární gramatice
  - 2.5 Konstrukce NKA k levé regulární gramatice

## Konečný automat

### Definice (nedeterministického) KA (NKA)

NKA je 5-ice  $(Q, \Sigma, \delta, q_0, F)$

$Q$  - konečná **množina stavů**

$\Sigma$  - konečná **vstupní abeceda**

$\delta$  - **funkce přechodu** (zobrazení  $Q \times \Sigma \rightarrow 2^Q$ )

$q_0$  - **počáteční stav** ( $q_0 \in Q$ )

$F$  - **množina koncových stavů** ( $F \subset Q$ )

#### Konfigurace

dvojice  $(q, w) \in Q \times \Sigma^*$  tj. stav a zbývající vstup

#### Počáteční konfigurace

dvojice  $(q_0, w)$  tj. počáteční stav a celý vstup

#### Koncová konfigurace

dvojice  $(q \in F, \epsilon)$  tj. některý koncový stav a prázdný vstup

#### Přechod ( $\vdash$ )

změna jedné konfigurace na jinou jednou aplikací funkce přechodu (přečtení jednoho znaku ze vstupu a změna stavu)

$\vdash^k$  -  $k$  přechodů po sobě

$\vdash^+$  - libovolný počet přechodů ( $>0$ ) (tranzitivní uzávěr relace přechodu)

$\vdash^*$  - libovolný počet přechodů (tranzitivní a reflexivní uzávěr relace přechodu)

#### Jazyk přijmaný KA

$$L(M) = \{w \mid (q_0, w) \vdash^* (q, \epsilon) \wedge q \in F\}$$

tj. množina všech řetězců pro které KA přejde z počáteční do koncové konfigurace

#### Rozlišitelné stavy

stavy jsou rozlišitelné pokud existuje nějaký vstupní řetězec pro který automat z jednoho stavu skončí v koncovém stavu a z druhého ne

Formálně: Stavy  $p, q$  jsou rozlišitelné pokud  $\exists w \in \Sigma^*(p, w) : (p, w) \vdash^* (p_1, \epsilon) \wedge (q, w) \vdash^* (q_1, \epsilon)$  a právě jeden z  $p_1, q_1$  je v  $F$

Jazyky přijímané KA jsou jazyky třídy 3 Chomského hierarchie (regulární jazyky)

## Varianty konečných automatů

### Rozšířený KA

Liší se definice funkce přechodu:

$\delta$  - funkce přechodu (zobrazení  $Q \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^Q$ )

tj. je rozšířený o  $\epsilon$ -přechody

### Deterministický konečný automat (DKA)

Liší se definice funkce přechodu:

$\delta$  - funkce přechodu (zobrazení  $Q \times \Sigma \rightarrow Q$ )

tj. pro každou konfiguraci existuje jen jeden možný přechod

- Jsou ekvivalentní NKA
- Každý NKA lze převést na DKA

### Úplný DKA

Funkce přechodu je definována pro všechny kombinace vstupního znaku a stavu. tj.  $\delta$  je totální funkcí na  $Q \times \Sigma$

- Pro případy, kdy neúplný KA nemohl pokračovat (a tím odmítl vstup) zavádíme spec. nekonečný stav ERROR do kterého, když se automat dostane tak jej nemůže opustit (tj. po přečtení celého vstupu je stále v nekonečném stavu a tak vstup odmítne)

### DKA bez nedosažitelných stavů

Je DKA ve kterém:

- eliminujeme nedosažitelné stavy

### Minimální/redukovaný DKA (MDKA)

Je DKA bez nedosažitelných stavů ve kterém:

- redukuje nerozlišitelné stavy
- odstraníme přebytečné stavy (stavy nemající vliv na přijímání vstupního řetězce)

## Algoritmy

### Odstranění nedosažitelných stavů

Převod DKA  $(Q, \Sigma, \delta, q_0, F)$  na DKA bez nedosažitelných stavů  $(S', \Sigma, \delta', q_0, F')$

1) Najdeme všechny dostupné stavy ( $S_i$ )

$i := 0$

$S_0 := \{q_0\}$  (tj. v nultém kroku je dostupný pouze počáteční stav)

repeat

$S_{i+1} := S_i \cup \{q \mid \exists p \in S_i \exists a \in \Sigma : \delta(p, a) = q\}$  (tj. do množiny dostupných stavů přidáme všechny stavy do kterých se dá libovolným symbolem přejít z některého dostupného stavu z minulého kroku)

$i := i + 1$  (tj. přejdeme k dalšímu kroku)

unitil  $S_i = S_{i-1}$  (tj. končíme pokud se nám v posledním kroku množina dostupných stavů nezměnila)

2) Nový automat sestavíme jako:

$S' = S_i$  (tj. použijeme pouze dosažitelné stavy)

použijeme pouze přechody obsahující dosažitelné stavy

použijeme pouze koncové stavy z množiny dosažitelných stavů ( $F' = F \cap S_i$ )

## Minimalizace KA

Převod DKA  $(Q, \Sigma, \delta, q_0, F)$  na MDKA  $(Q', \Sigma, \delta', q_0', F')$

1) Převedeme na DKA bez nedostupných stavů

2) Určíme relaci nerozlišitelnosti stavů ( $\equiv^i$ )

$i := 0$

$\equiv^0 := \{(p, q) | p \in F \Leftrightarrow q \in F\}$  (tj. na začátku rozlišujeme pouze dvě třídy - koncové a nekoncové stavy)

repeat

$\equiv^{i+1} := \{(p, q) | p \equiv^i q \forall a \in \Sigma : \delta(p, a) \equiv^i \delta(q, a)\}$  (tj. prvky zůstanou ekvivalentní pouze pokud v minulém kroku vyšly ekvivalentní a všechny možné výsledky odpovídajících přechodů z nich jsou také ekvivalentní)

$i := i + 1$

unitil  $\equiv^i = \equiv^{i-1}$  (tj. končíme pokud v posledním kroku nedošlo k žádným změnám tříd ekvivalence)

3) Sestavíme nový automat jako:

$Q' = Q / \equiv^i$  (tj. nové stavy odpovídají třídám ekvivalence původních stavů podle relace nerozlišitelnosti)

$\forall p, q \in Q' \forall a \in \Sigma : \delta'([p], a) = [q] \Leftrightarrow \delta(p, a) = q$  (tj. nové přechody jsou mezi těmi třídami ekvivalence u kterých existuje alespoň jeden odpovídající přechod mezi některými jejich prvky)

$q_0' = [q_0]$  (tj. počáteční stav je ta třída ekvivalence, která obsahuje původní počáteční stav)

$F' = \{[q] | q \in F\}$  (tj. koncové stavy jsou ty třídy ekvivalence, která obsahují některý z původních koncových stavů)

## Převod NKA na DKA

Převod NKA  $(Q, \Sigma, \delta, q_0, F)$  na DKA  $(Q', \Sigma, \delta', q_0', F')$

1)  $Q' = (2^Q \setminus \{\emptyset\}) \cup \{nedef.\}$

(tj. nové stavy označujeme jako množiny původních stavů, místo prázdné množiny máme spec. stav {nedef.})

2) Pro všechna  $S \in 2^Q \setminus \emptyset$  a pro všechna  $a \in \Sigma$  polož:

$$\delta'(S, a) = \bigcup_{q \in S} \delta(q, a)$$

Je-li  $\delta'(S, a) = \{\emptyset\}$  polož  $\delta'(S, a) = nedef.$

(tj. pro všechny možné stavy  $S$  DKA a všechny možné znaky  $a$  na vstupu určíme následující stav jako sjednocení všech stavů do kterých vedou přechody se znakem  $a$  ze stavů v množině  $S$ . Pokud žádné takové přechody neexistují je přechod nedefiovaný.)

3)  $F' = \{S, S \in 2^Q \wedge S \cap F \neq \emptyset\}$

(tj. za koncové stavy DKA považujeme všechny stavy jejichž množina obsahuje alespoň jeden koncový stav z NKA)

**Alternativní přístup**

(využíváme skutečnost, že většina stavů z  $2^Q$  je nedostupná)

1) Stavy DKA vznikají jako množiny stavů NKA

2) Určíme počáteční stav DKA jako  $\{q_0\}$  a přidáme si ho na seznam nezpracovaných stavů

3) Pro každý nezpracovaný stav X:

vezmeme všechny stavy NKA, které jsou součástí množiny X - označíme  $\{X\}$

podíváme se jaké všechny symboly se vyskytují u přechodů vedoucích z  $\{X\}$

pro každý z těchto symbolů pak:

určíme množinu stavů NKA (označíme Y) do kterých se lze dostat nějakým přechodem s tímto symbolem z některého z některého stavu v  $\{X\}$

do DKA přidáme přechod s tímto symbolem vedoucí z X do Y

pokud Y dosud není součástí DKA přidáme ho na seznam nezpracovaných stavů

stav X označíme za zpracovaný a přidáme ho do množiny stavů DKA

4) Pokud jsou nějaké nezpracované stavy jdeme zpět ke 3

5) Za koncové stavy DKA považujeme všechny stavy jejichž množina obsahuje alespoň jeden koncový stav z NKA

## Konstrukce NKA k pravé regulární gramatice

Gramatika  $g = (N, \Sigma, P, S)$  na NKA  $(Q, \Sigma, \delta, q_0, F)$

1)  $Q = N \cup \{q_F\}$  (stavy NKA odpovídají nonterminálům abecedy s přidaným  $q_F$ )

2)  $\Sigma = \Sigma$  (tj. vstupní abeceda a množina terminálů jsou stejné)

3)  $q_0 = S$  (tj. počáteční stav odpovídá počátečnímu nonterminálu gramatiky)

4) Je-li  $S \rightarrow \epsilon$  pravidlo z P pak  $F = \{S, q_F\}$  jinak  $F = \{q_F\}$  (tj. koncové stavy jsou  $q_F$ ) a v případě, že v gramtice je i prázdný řetězec tak i S

5) funkci přechodů definujeme:

Je-li  $A \rightarrow aB$  pravidlo z P, pak  $\delta(A, a)$  obsahuje B

Je-li  $A \rightarrow a$  pravidlo z P, pak  $\delta(A, a)$  obsahuje  $q_F$

## Konstrukce NKA k levé regulární gramatice

Gramatika  $g = (N, \Sigma, P, S)$  na NKA  $(Q, \Sigma, \delta, q_0, F)$

1)  $Q = N \cup \{q_0\}$  (stavy NKA odpovídají nonterminálům abecedy s přidaným  $q_0$ )

2)  $\Sigma = \Sigma$  (tj. vstupní abeceda a množina terminálů jsou stejné)

3)  $q_0$  je počáteční stav

4) Je-li  $S \rightarrow \epsilon$  pravidlo z P pak  $F = \{S, q_0\}$  jinak  $F = \{S\}$  (tj. koncové stavy jsou S) a v případě, že v gramtice je i prázdný řetězec tak i  $q_0$

5) funkci přechodů definujeme:

Je-li  $A \rightarrow Ba$  pravidlo z P, pak  $\delta(B, a)$  obsahuje A

Je-li  $A \rightarrow a$  pravidlo z P, pak  $\delta(q_0, a)$  obsahuje A