<KIV> KATEDRA INFORMATIKY A VÝPOČETNÍ TECHNIKY

FAKULTA APLIKOVANÝCH VĚD ZÁPADOČESKÉ UNIVERZITY V PLZNI

Předmět KIV/TI - přednáška 4

Regulární jazyky a jejich souvislost s konečnými automaty - II

Ing. Václav Vais, Ph.D.

vais@kiv.zcu.cz

Závěry z předešlých přednášek

- Každá gramatika generuje jazyk

$$L(G) = \{ w \mid w \in T^* \land S \overset{*}{\Rightarrow} w \}$$

- Gramatiky typu 3 generují jazyky rozpoznatelné konečnými automaty

  (regulární jazyky)

- Každý rozpoznávací konečný automat rozpoznává jazyk

$$L(A) = \{w \mid w \in \Sigma^* \land \delta^*(q_0, w) \in F\}$$

- Jak na základě přepisovacích pravidel gramatiky typu 3 sestrojit konečný automat?

Motivační příklad

$$G: \quad S \to \overset{1}{0A} \mid \overset{2}{1S} \mid \overset{3}{e}$$

$$A \to \overset{4}{0B} \mid \overset{5}{1A}$$

$$B \to \overset{6}{0S} \mid \overset{7}{1B}$$

Gramatika *G* je typu G3P

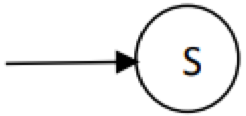
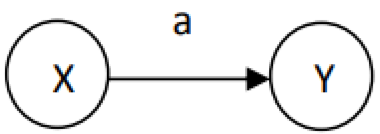
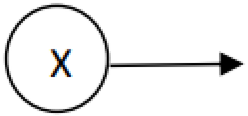
(pravá lineární gramatika).

Motivační příklad

Zkonstruujeme obrázek (= „automat“) tímto formálním postupem:

- stavy „automatu“ budou odpovídat neterminálním symbolům gramatiky
- vstupní symboly „automatu“ budou odpovídat terminálním symbolům
- počáteční stav „automatu“ bude odpovídat počátečnímu symbolu gramatiky
- v množině koncových stavů „automatu“ budou všechny stavy X , pro které je v gramatice pravidlo $X \longrightarrow e$
- přechodovou funkci vytvoříme tak, že každému pravidlu $X \longrightarrow aY$ z gramatiky bude v přechodovém grafu odpovídat hrana ze stavu X do stavu Y ohodnocená vstupním symbolem a

Motivační příklad – analogie přehledně

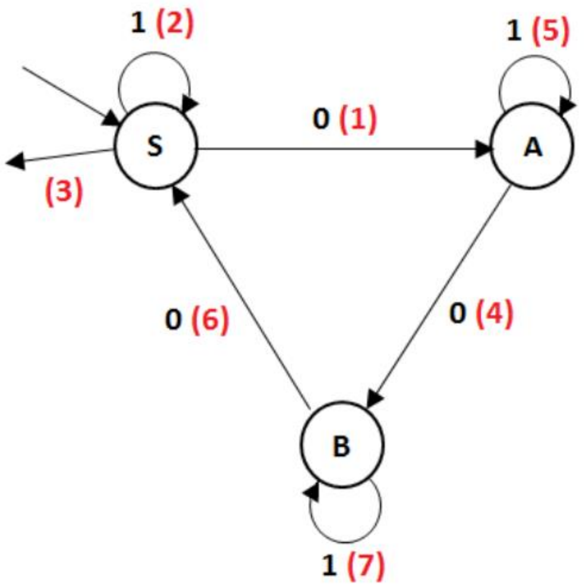
gramatika	**„automat“**
počáteční symbol S	
pravidlo $X \longrightarrow aY$	
pravidlo $X \longrightarrow e$	

Motivační příklad

$$G: \quad S \to \overset{1}{0A} \mid \overset{2}{1S} \mid \overset{3}{e}$$

$$A \to \overset{4}{0B} \mid \overset{5}{1A}$$

$$B \to \overset{6}{0S} \mid \overset{7}{1B}$$



Tento graf je (shodou okolností, nikoli obecně) přechodovým grafem konečného automatu.

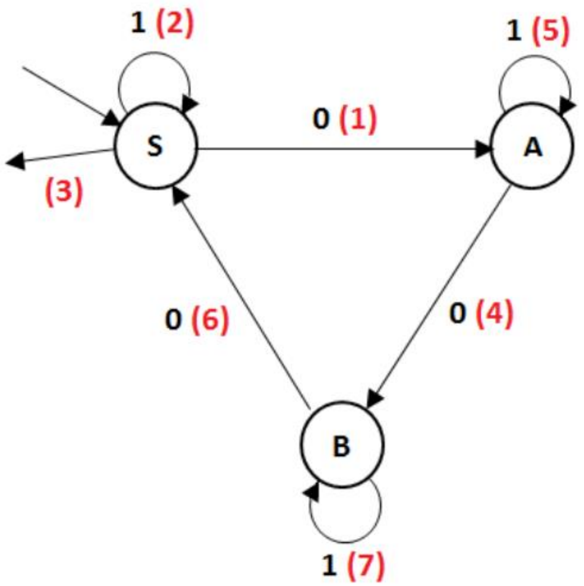
Budeme zkoumat generování řetězců gramatikou a zpracování týchž řetězců automatem.

Motivační příklad

$$G: \quad S \to \overset{1}{0A} \mid \overset{2}{1S} \mid \overset{3}{e}$$

$$A \to \overset{4}{0B} \mid \overset{5}{1A}$$

$$B \to \overset{6}{0S} \mid \overset{7}{1B}$$



Odvození slova *00101* v gramatice *G*:

$$S \overset{1}{\Rightarrow} 0A \overset{4}{\Rightarrow} 00B \overset{7}{\Rightarrow} 001B \overset{6}{\Rightarrow} 0010S \overset{2}{\Rightarrow} 00101S \overset{3}{\Rightarrow} 00101$$

Zpracování slova *00101* v automatu:

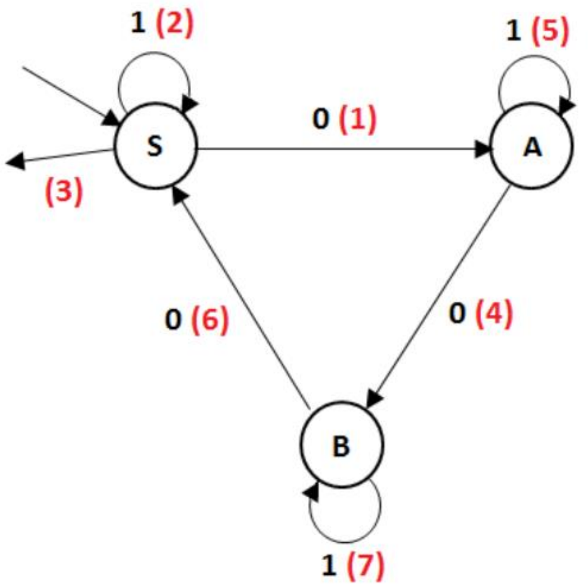
$$(S,00101) \overset{1}{\mapsto} (A,0101) \overset{4}{\mapsto} (B,101) \overset{7}{\mapsto} (B,01) \overset{6}{\mapsto} (S,1) \overset{2}{\mapsto} (S,e)$$

Motivační příklad

$$G: \quad S \to \overset{1}{0A} \mid \overset{2}{1S} \mid \overset{3}{e}$$

$$A \to \overset{4}{0B} \mid \overset{5}{1A}$$

$$B \to \overset{6}{0S} \mid \overset{7}{1B}$$



Odvození slova *00101* v gramatice *G*:

= posloupnost přímých přepsání 1 4 7 6 2 3

Zpracování slova *00101* v automatu:

= posloupnost přechodů 1 4 7 6 2

Motivační příklad - závěry

- Z uvedeného je zřejmé:
  - ke každému řetězci generovanému gramatikou *G* existuje v automatu posloupnost přechodů končící v koncovém stavu (důsledek toho, že jsme „přechodový graf vytvořili podle uvedených analogií), tedy $L(A) = L(G)$.
  - na základě uvedených analogií lze sestrojit KA (myšleno KA ve smyslu naší definice z přednášky 1) pouze ke gramatikám typu G3P s pravidly v regulárním tvaru $X \to aY$ nebo $X \to e$, kde $X, Y \in N$ a $a \in T$, kde navíc pro žádný neterminální symbol neexistuje více než jedno pravidlo se stejným terminálním symbolem na pravé straně.

Cesta od gramatiky G3P ke KA (obecný postup)

- ukážeme, že ke každé gramatice typu G3P existuje exvivalentní gramatiky s pravidly ve tvaru $X \to aY$ nebo $X \to e$, kde $X, Y \in N$ a $a \in T$ (terminologie: regulární tvar pravidel, gramatika v regulárním tvaru)
- pojem konečného automatu zobecníme na nedeterministický konečný automat, který bude připouštět nejednoznačné přechody
- ukážeme, že ke každému nedeterministickému konečnému automatu existuje ekvivalentní konečný automat ve smyslu naší dosavadní definice

Převod gramatiky G3P na regulární tvar

<u>Tvrzení:</u> Ke každé gramatice $G = (N, T, S, P)$ typu G3P existuje gramatika $G' = (N', T, S, P')$ s pravidly v regulárním tvaru $X \to aY$ nebo $X \to e$, kde $X, Y \in N$ a $a \in T$, taková, že $L(G) = L(G')$.

<u>Důkaz (= postup pro konstrukci ekvivalentní gramatiky v regulárním tvaru) :</u>

1. Množina terminálních symbolů T a počáteční symbol S jsou v G' stejné jako v G.
2. Množinu přepisovacích pravidel P' zkonstruujeme takto:
   a. Do P' zařadíme všechna pravidla z P, která jsou v požadovaném regulárním tvaru $X \to aY$ nebo $X \to e$, kde $X, Y \in N$ a $a \in T$.
   b. Za každé pravidlo $X \to x_1 x_2 \ldots x_{n-1} x_n Y$ z P ($X, Y \in N$, $x_i \in T$) přidáme do P' soustavu pravidel $X \to x_1 X_1$, $X_1 \to x_2 X_2$, ………, $X_{n-2} \to x_{n-1} X_{n-1}$, $X_{n-1} \to x_n Y$ ($X_i \in N$).

Převod gramatiky G3P na regulární tvar

c. Za každé pravidlo $X \rightarrow z_1 z_2 \ldots. z_n$ z P ($X \in N$, $z_i \in T$) přidáme do P' soustavu pravidel $X \rightarrow z_1 Z_1$, $Z_1 \rightarrow z_2 Z_2$, ..., $Z_{n-1} \rightarrow z_n Z_n$, $Z_n \rightarrow e$ ($Z_i \in N$) .

d. Místo pravidel tvaru $X \rightarrow Y$ z P přidáme do P' soustavu pravidel ve tvaru $Z' \rightarrow z\, Z''$ $\forall\ Z' \in U(Y)$ $\forall\ Y \rightarrow z\, Z'' \in P$ kde $U(Y)$ = $\{X \mid X \overset{*}{\Rightarrow} Y\}$

3. Množina neterminálních symbolů N' vznikne obohacením množiny N o všechny nové neterminální symboly vytvořené v bodech 2a, 2b a 2c.

Převod gramatiky G3P na regulární tvar

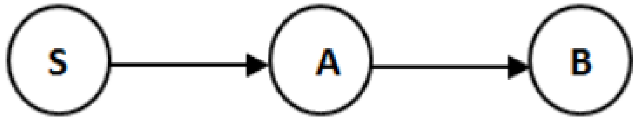
Poznámky k praktickému provedení transformce:

- Provedení bodů 2b a 2c je triviální (postup lze popsat jako „postupné odřezávání“ terminálních symbolů)
- K provedení bodu 2d je u složitějších gramatik vhodné nakreslit pomocný graf, který bude zobrazovat pravidla tvaru $X \rightarrow Y$, a z něj vyčíst množiny $U(X)$
- Důležitým faktorem je udržet si přehled v tom, která pravidla jsem již zpracovali (tj. nahradili ekvivalentními soustavami pravidel), je tedy vhodné si zpracovaná pravidla v zadání označovat (odškrtávat)

Převod gramatiky G3P na regulární tvar

- Ilustrace postupu podle pravidla 2d:

Pro pravidla $S \to A$, $A \to B$ v G bude pomocný graf vypadat takto:



Za každé pravidlo $X \to Y$ v G je v pomocném grafu orientovaná hrana z vrcholu X do vrvholu Y . Množiny $U(X)$ pak snadno vytvoříme jako množiny všech (cizích) vrcholů, z nichž vede cesta do vrcholu X. Z teorie grafů víme, že z každého vrcholu X vede také cesta délky 0 do vrcholu X, v našem případě cesty délky 0 nepřinášejí potřebu nových přepisovacích pravidel, proto sám vrchol X prvkem množiny $U(X)$ nebude. Z pomocného grafu snadno zjistíme, že $U(S) = \emptyset$, $U(A) = \{S\}$, $U(B) = \{S, A\}$.

Převod gramatiky G3P na regulární tvar

<u>Příklad:</u> Gramatiku G transformujte na regulární tvar:

$$
\begin{aligned}
G: \quad & S \to \overset{1}{A} \mid \overset{2}{B} \mid \overset{3}{aaS} \\
& A \to \overset{4}{C} \mid \overset{5}{abB} \\
& B \to \overset{6}{bB} \mid \overset{7}{c} \\
& C \to \overset{8}{cC} \mid \overset{9}{bc}
\end{aligned}
$$

Převod gramatiky G3P na regulární tvar

- podle bodu 2a vložíme do P' pravidla, která již regulární tvar mají:

  $B \rightarrow bB$ (6) a $C \rightarrow cC$ (8)

- podle bodu 2b transformujeme pravidla (3) a (5):

  místo $S \rightarrow aaS$ (3) vložíme do P' $S \rightarrow aS_1$ a $S_1 \rightarrow aS$

  místo $A \rightarrow abB$ (5) vložíme do P' $A \rightarrow aA_1$ a $A_1 \rightarrow bB$

- podle bodu 2c transformujeme pravidla (7) a (9):

  místo $B \rightarrow c$ (7) vložíme do P' $B \rightarrow cB_1$ a $B_1 \rightarrow e$

  místo $C \rightarrow bc$ (9) vložíme do P' $C \rightarrow bC_1$, $C_1 \rightarrow c\,C_2$ a $C_2 \rightarrow e$

Převod gramatiky G3P na regulární tvar

- před provedením kroku 2d pravidla dosud zařazená do P' seřadíme:

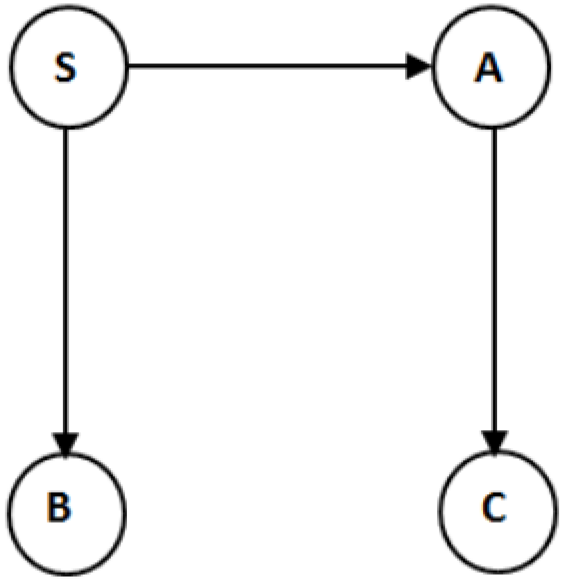
$$
\begin{array}{lll}
S \to aS_1 \;, & S_1 \to aS & \\
A \to aA_1 \;, & A_1 \to bB & \\
B \to bB \mid cB_1 \;, & B_1 \to e & \\
C \to cC \mid bC_1 \;, & C_1 \to cC_2 \;, & C_2 \to e
\end{array}
$$

- zbývající nezpracovaná pravidla v gramatice *G*:

$$S \to A,\; S \to B\;, A \to C\,.$$

Převod gramatiky G3P na regulární tvar

- nezpracovaná pravidla $S \to A,\ S \to B,\ A \to C$ znázorníme v grafu:



$$U(S) = \emptyset$$
$$U(A) = \{S\}$$
$$U(B) = \{S\}$$
$$U(C) = \{S, A\}$$

Převod gramatiky G3P na regulární tvar

- „prokombinujeme“ prvky z *U(Y)* s pravými stranami *Y* pravidel:

	protože v G platí	a	protože v G´ jsou
$U(S) = \emptyset$			$S \to aS_1$
$U(A) = \{S\}$	$S \overset{*}{\Rightarrow} A$		$A \to aA_1$
$U(B) = \{S\}$	$S \overset{*}{\Rightarrow} B$		$B \to bB \mid cB_1$
$U(C) = \{S, A\}$	$S \overset{*}{\Rightarrow} C$, $A \overset{*}{\Rightarrow} C$		$C \to cC \mid bC_1$

do P' tedy přidáme pravidla

$$S \to aA_1 \qquad S \to bB \mid cB_1 \qquad S \to cC \mid bC_1 \qquad A \to cC \mid bC_1$$

Převod gramatiky G3P na regulární tvar

- pravidla v P' seřadíme a očíslujeme:

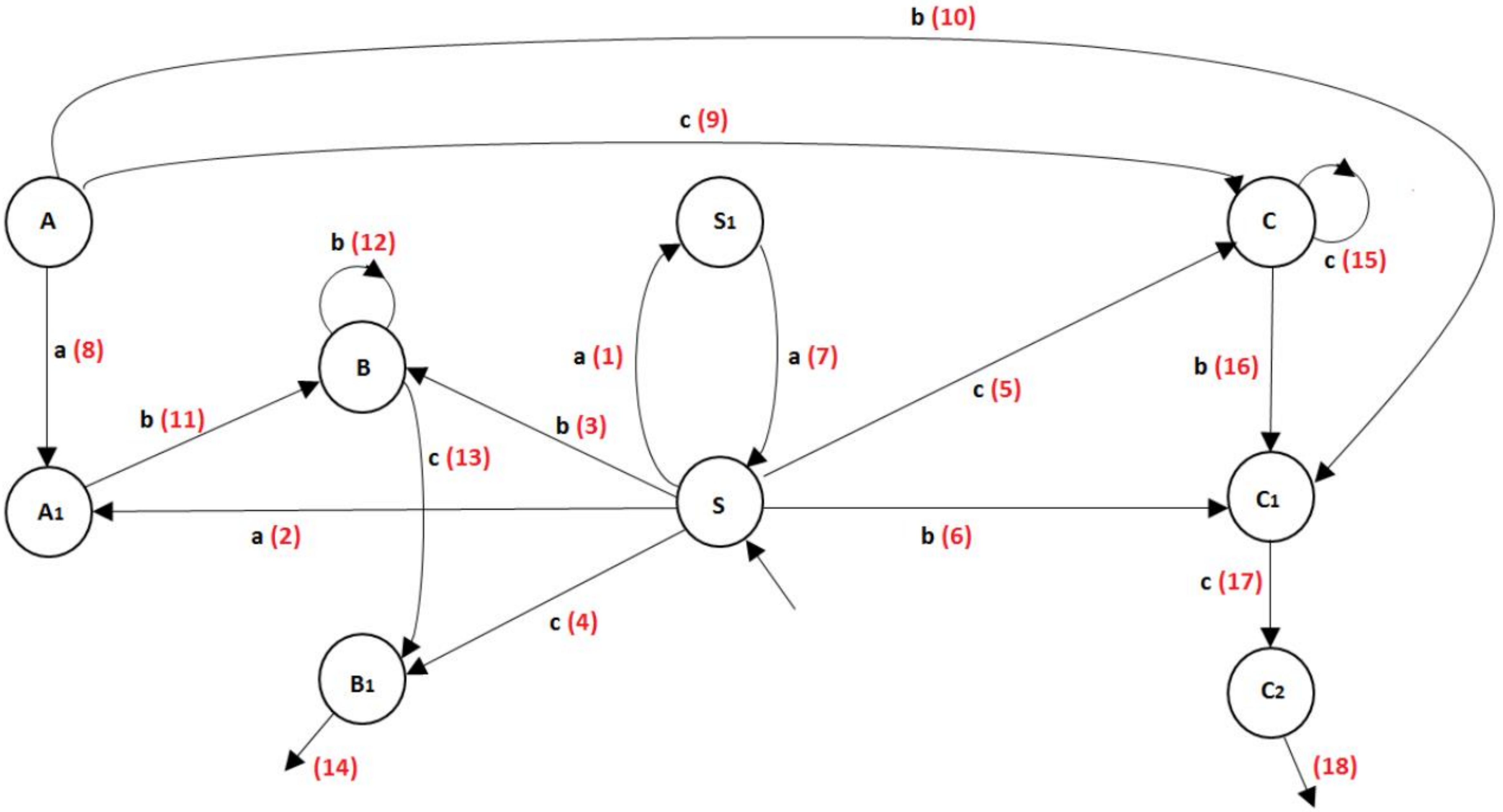
$$G': \quad S \rightarrow \overset{1}{aS_1} \mid \overset{2}{aA_1} \mid \overset{3}{bB} \mid \overset{4}{cB_1} \mid \overset{5}{cC} \mid \overset{6}{bC_1} \qquad S_1 \rightarrow \overset{7}{aS}$$

$$A \rightarrow \overset{8}{aA_1} \mid \overset{9}{cC} \mid \overset{10}{bC_1} \qquad A_1 \rightarrow \overset{11}{bB}$$

$$B \rightarrow \overset{12}{bB} \mid \overset{13}{cB_1} \qquad B_1 \rightarrow \overset{14}{e}$$

$$C \rightarrow \overset{15}{cC} \mid \overset{16}{bC_1} \qquad C_1 \rightarrow \overset{17}{cC_2}\,, \quad C_2 \rightarrow \overset{18}{e}$$

- a na základě analogií vytvoříme přechodový graf ....


b (10)
c (9)
A
S1
C
b (12)
c (15)
a (8)
B
a (1)
a (7)
b (16)
c (5)
b (11)
c (13)
b (3)
A1
S
C1
a (2)
b (6)
c (17)
c (4)
B1
C2
(14)
(18)

Nedeterministický rozpoznávací konečný automat – definice

$$A = (Q, \Sigma, \delta, S, F)$$

kde Q je konečná neprázdná množina stavů

Σ je konečná neprázdná množina vstupních symbolů

$S \subseteq Q$ je množina počátečních stavů

$\delta : Q \times (\Sigma \cup \{e\}) \rightarrow 2^Q$ je přechodová funkce

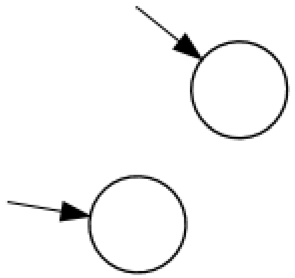
$F \subseteq Q$ je množina koncových stavů

Nedeterministický rozpoznávací konečný automat – poznámky

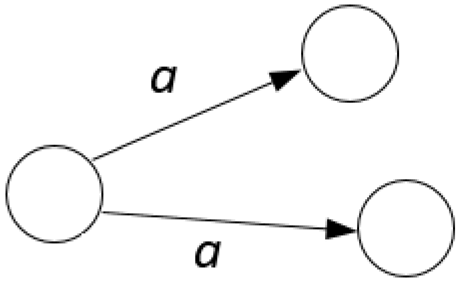
- Na začátku své činnosti se automat nachází v některém ze stavů z množiny počátečních stavů S
- Je-li v přechodovém grafu e-hrana, znamená to, že tento přechod automat může (ale nemusí) provést samovolně, aniž by zpracoval symbol ze vstupního řetězce
- Obor hodnot přechodové funkce 2^Q představuje množinu všech podmnožin stavové množiny Q; tedy pro konkrétní stav a vstupní písmeno může být hodnotou přechodové funkce více než jeden následující stav
- POZOR !! Automat je vždy jen v jednom stavu, přejde do některého z nich.

Typy nedeterminismu v definici NKA

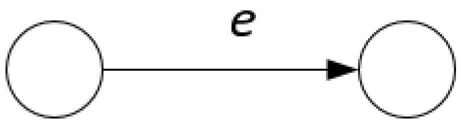
- nejednoznačně určený počáteční stav



- nejednoznačné přechody



- možnost samovolného přechodu (*e*-přechody)



Kdy je řetězec *w* akceptován NKA?

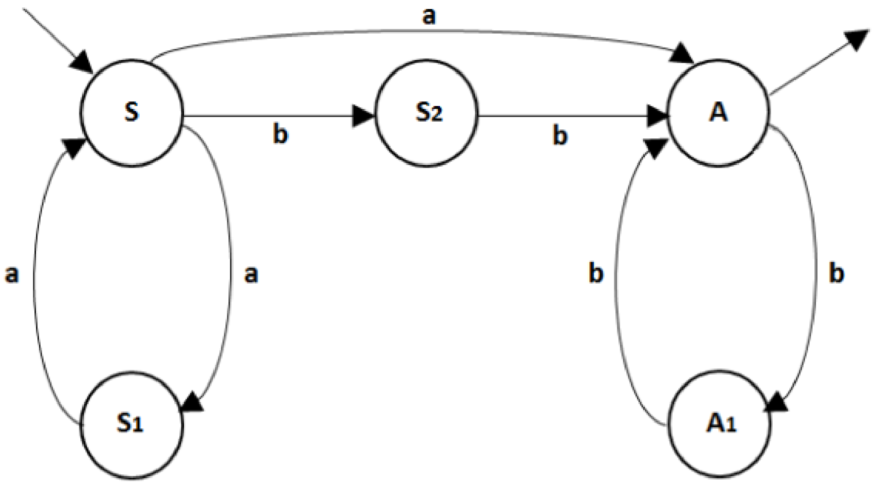
- U deterministického KA jsme byli vždy schopni určit, do jakého stavu se zpracováním vstupního řetězce *w* automat dostal.
- U nedeterministického KA tomu tak není
- Je třeba najít nový pohled na pojmy
  - akceptovaný řetězec
  - jazyk rozpoznávaný NKA
- Jak akceptovaný řetězec definovat? Řetězec *w* by měl být akceptován NKA právě tehdy, když jej lze odvodit ve výchozí gramatice.

Kdy je řetězec *w* akceptován NKA?

<u>Příklad:</u>

$$G: \quad S \to \overset{1}{aaS} \mid \overset{2}{bbA} \mid \overset{3}{aA}$$

$$A \to \overset{4}{bbA} \mid \overset{5}{e}$$



$S \overset{3}{\Rightarrow} aA \overset{5}{\Rightarrow} a$, tedy $a \in L(G)$

$$(S,a) \overset{3}{\mapsto} (A,e) \qquad \overset{(5)}{A \in F}$$

$$(S,a) \overset{1}{\mapsto} (S_1,e) \qquad S_1 \notin F$$

Kdy je řetězec *w* akceptován NKA?

- Řetězec *w* je NKA akceptován právě tehdy, jestliže v přechodovém grafu existuje alespoň jedna cesta, jejíž hrany jsou ohodnoceny písmeny řetězce *w* (nebo symboly *e*), která začíná v některém z počátečních stavů a končí v některém z koncových stavů.
- Přesnější formulace pro NKA bez e-hran:

  Řetězec $w = w_1 w_2 \ldots\ldots w_n \in \Sigma^*$ je NKA akceptován právě tehdy, když existuje posloupnost stavů $q_0 q_1 q_2 \ldots\ldots q_n$ taková, že

  $q_0 \in S$, $q_n \in F$ a $q_{i+1} \in \delta(q_i, w_{i+1})$ $\forall\ i = 0, 1, \ldots\ldots, n-1$.

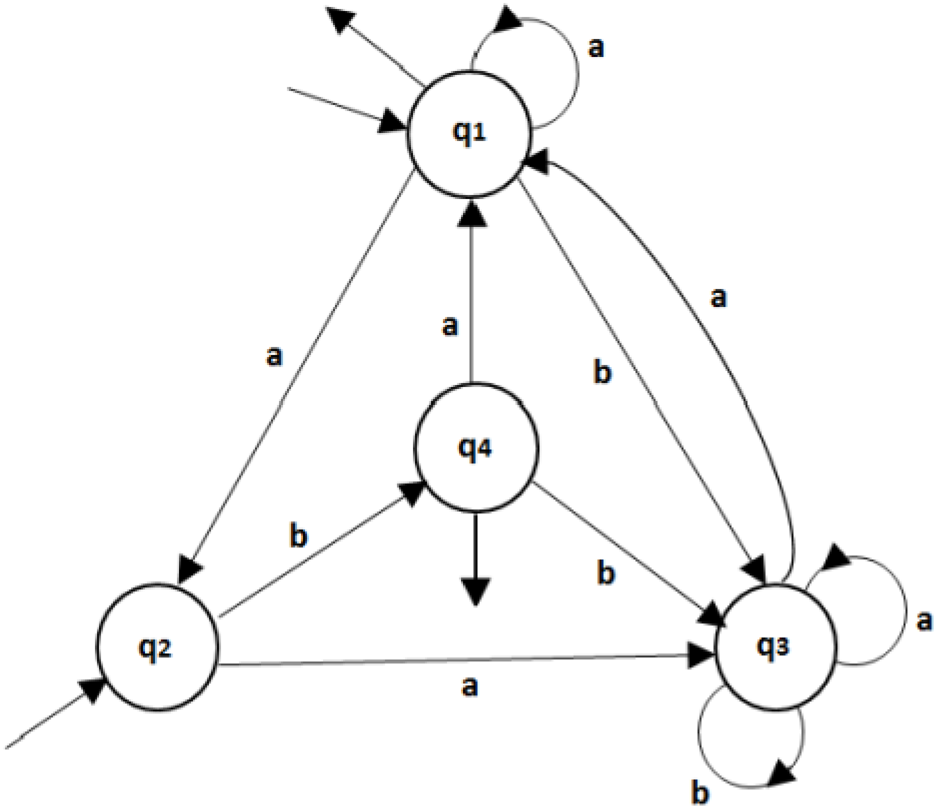
Kdy je řetězec *w* akceptován NKA?

- Kdy NKA akceptuje prázdný řetězec *e* ?
  - když $S \cap F \neq \emptyset$

    nebo
  - když existuje cesta složená z *e*-hran z některého z počátečních stavů do některého z koncových stavů
- Různé možnosti zpracování (jednoho) řetězce chápeme jako různé *výpočty.*
- U deterministického KA existuje pro každý řetězec právě jeden výpočet.
- U nedeterministického KA toto neplatí.

Příklad NKA bez *e*-hran

		a	b
$\leftrightarrow$	q_1	$\{q_1,q_2\}$	$\{q_3\}$
$\rightarrow$	q_2	$\{q_3\}$	$\{q_4\}$
	q_3	$\{q_1,q_3\}$	$\{q_3\}$
$\leftarrow$	q_4	$\{q_1\}$	$\{q_3\}$



Jak může reagovat na *abaa ?*

$$(q_2,abaa)\mapsto(q_3,baa)\mapsto(q_3,aa)\mapsto(q_1,a)\mapsto(q_3,e) \qquad q_3\notin F$$

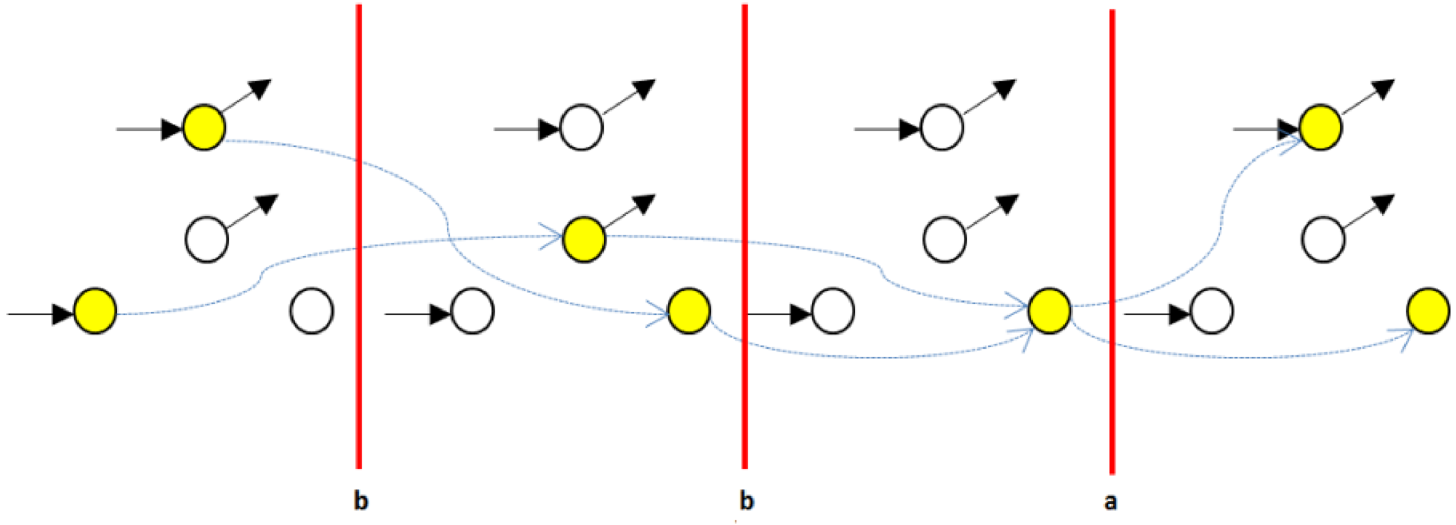
$$(q_2,abaa)\mapsto(q_3,baa)\mapsto(q_3,aa)\mapsto(q_3,a)\mapsto(q_3,e) \qquad q_3\notin F$$

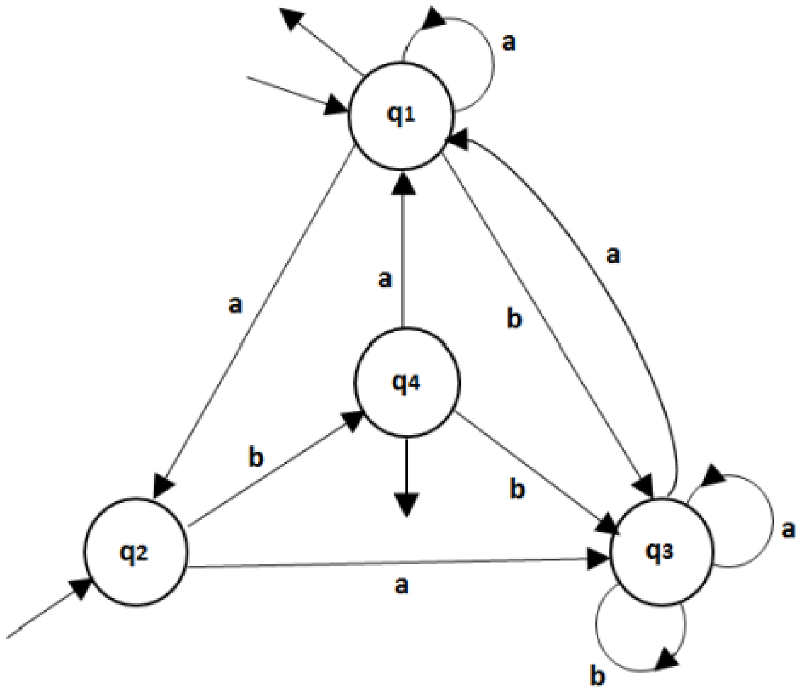
$$(q_1,abaa)\mapsto(q_1,baa)\mapsto(q_3,aa)\mapsto(q_1,a)\mapsto(q_1,e) \qquad q_1\in F$$

Příklad NKA bez *e*-hran

Jak může reagovat na řetězec *bba* ?

Budeme analyzovat „všechny výpočty najednou“.





Žlutě vybarven je stav, v němž v dané fázi zpracování řetězce automat může být.

Žlutě vybarvený koncový stav = řetězec je akceptován.

Závěry k příkladu

- Chování NKA lze popsat sekvencí pozic, z nichž každá jednoznačně definuje, zda je zpracovaný řetězec akceptován či zamítnut. Takových pozic je konečný počet.
- Jednoznačně je určena počáteční pozice.
- Přechody mezi pozicemi jsou jednoznačně určené.
- To vše jsou vlastnosti deterministického konečného automatu (KA).
- Ke každému NKA existuje ekvivalentní KA.
- KA a NKA rozpoznávají tutéž třídu jazyků.

Obecný vztah mezi NKA a ekvivalentním KA

(uvažován NKA bez e-hran)

$A = (Q, \Sigma, \delta, S, F)$ je nedeterministický rozpoznávací konečný automat

$A' = (Q', \Sigma, \delta', S, F')$ je ekvivalentní deterministický rozpoznávací konečný automat

Pak platí: $Q' \subseteq 2^Q$,

$$\delta' : Q' \times \Sigma \to Q', \quad \delta'(K, x) = \bigcup_{q \in K} \delta(q, x) \quad \forall K \in Q', \forall x \in \Sigma$$

$$F' = \{K \mid K \in Q' \wedge K \cap F \neq \emptyset\}$$

Jak sestrojit ekvivalentní KA prakticky?

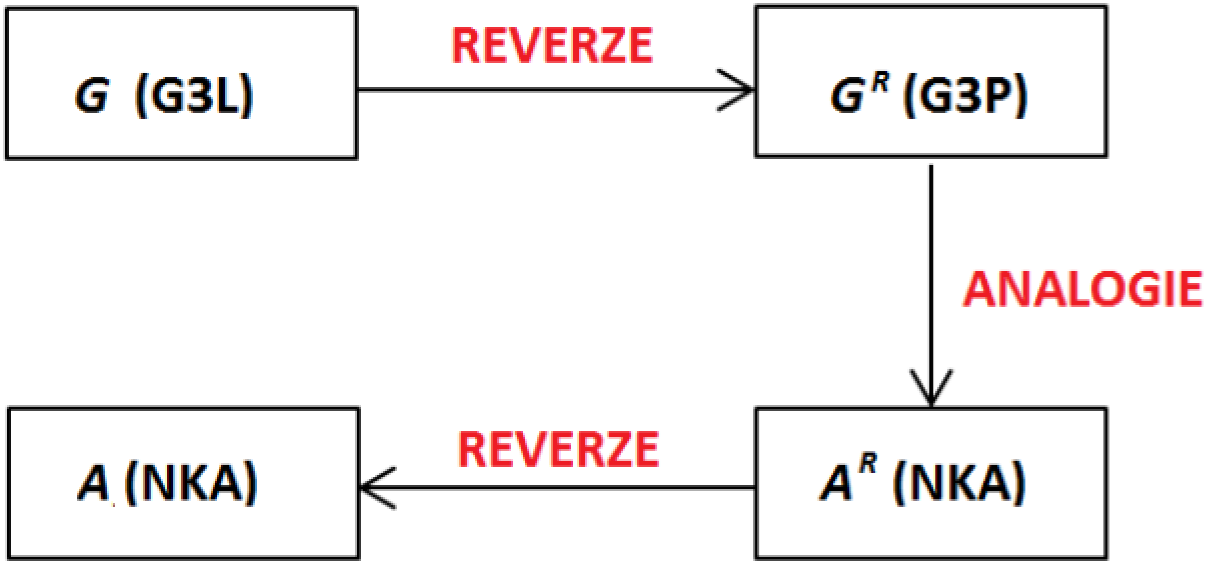
- Předem nelze určit počet stavů ekvivalentního KA.
- Počáteční stav KA odpovídá množině počátečních stavů NKA.
- Stavy KA vytváříme postupně tak, že vyhodnocujeme přechodovou funkci δ' pro „již vypočítané“ stavy KA.
- Hodnota přechodové funkce δ' pro konkrétní podmnožinu $K \subseteq Q$ a konkrétní vstupní písmeno *x* se získá jako sjednocení hodnot funkce δ pro všechny prvky množiny K a vstupní písmeno *x* .
- Koncovými stavy KA (tedy množinou F') budou všechny stavy, které v sobě obsahují některý z koncových stavů výchozího NKA (tedy množiny F).

Jak sestrojit ekvivalentní KA prakticky?

		a	b
$\leftrightarrow$	q_1	$\{q_1,q_2\}$	$\{q_3\}$
$\rightarrow$	q_2	$\{q_3\}$	$\{q_4\}$
	q_3	$\{q_1,q_3\}$	$\{q_3\}$
$\leftarrow$	q_4	$\{q_1\}$	$\{q_3\}$

		a	b
$\leftrightarrow$	$A\ \{q_1,q_2\}$	$B\ \{q_1,q_2,q_3\}$	$C\ \{q_3,q_4\}$
$\leftarrow$	$B\ \{q_1,q_2,q_3\}$	$B\ \{q_1,q_2,q_3\}$	$C\ \{q_3,q_4\}$
$\leftarrow$	$C\ \{q_3,q_4\}$	$D\ \{q_1,q_3\}$	$E\ \{q_3\}$
$\leftarrow$	$D\ \{q_1,q_3\}$	$B\ \{q_1,q_2,q_3\}$	$E\ \{q_3\}$
	$E\ \{q_3\}$	$D\ \{q_1,q_3\}$	$E\ \{q_3\}$

Jak sestrojit NKA k levé lineární gramatice?



Reverze gramatiky

- *Reverzní gramatika G^R* ke gramatice *G* = gramatika, která má na levé i pravé straně převrácené řetězce z gramatiky *G.*

Přesněji:

Nechť $G=(N,T,S,P)$ je gramatika. Reverzní gramatikou $G^R=(N,T,S,P^R)$ ke gramatice G budeme rozumět takovou gramatiku, pro kterou platí $\alpha \to \beta \in P^R \Leftrightarrow \alpha^R \to \beta^R \in P$.

Pak $L(G^R)=\{w \mid w^R \in L(G)\}$.

Platí $(G^R)^R = G$.

Ilustrační příklad k reverzi gramatiky

<u>Příklad:</u>

$$G_1: \quad S \to aS$$
$$aS \to bA \mid bba$$
$$A \to abb$$

$$G^R: \quad S \to Sa$$
$$Sa \to Ab \mid abb$$
$$A \to bba$$

Odvození řetězců z G :

$$S \Rightarrow aS \Rightarrow bA \Rightarrow babb$$
$$S \Rightarrow aS \Rightarrow bba$$

Odvození řetězců z G^R :

$$S \Rightarrow Sa \Rightarrow Ab \Rightarrow bbab$$
$$S \Rightarrow Sa \Rightarrow abb$$

Reverze rozpoznávacího NKA

- *Reverzní automat A^R* k automatu A = automat, v jehož přechodovém grafu jsou obráceny orientace všech šipek přechodového grafu automatu A (obrátí se orientace všech přechodových hran, navzájem se zamění počáteční a koncové stavy).

Přesněji:

Nechť $A = (Q, \Sigma, \delta, S, F)$ je nedeterministický rozpoznávací konečný automat. Reverzním automatem $A^R = (Q, \Sigma, \delta^R, S^R, F^R)$ k automatu A budeme rozumět takový automat, pro který platí $q' \in \delta^R(q, x) \Leftrightarrow q \in \delta\ (q', x)\ \forall q, q' \in Q$, $\forall x \in \Sigma$; $S^R = F$, $F^R = S$.

Pak $L(A^R) = \{w \mid w^R \in L(A)\}$. Platí $(A^R)^R = A$.