<KIV> KATEDRA INFORMATIKY A VÝPOČETNÍ TECHNIKY

FAKULTA APLIKOVANÝCH VĚD ZÁPADOČESKÉ UNIVERZITY V PLZNI

Předmět KIV/TI - přednáška 5

Regulární jazyky a jejich souvislost s konečnými automaty - III

Ing. Václav Vais, Ph.D.

vais@kiv.zcu.cz

Jak sestrojit ekvivalentní KA k NKA s *e*-hranami?

- Nejprve je třeba vytvořit ekvivalentní NKA bez *e*-hran.
- K NKA bez *e*-hran se poté vytvoří ekvivalentní KA dříve prezentovaným způsobem.

Ekvivalentní odstranění e-hran:

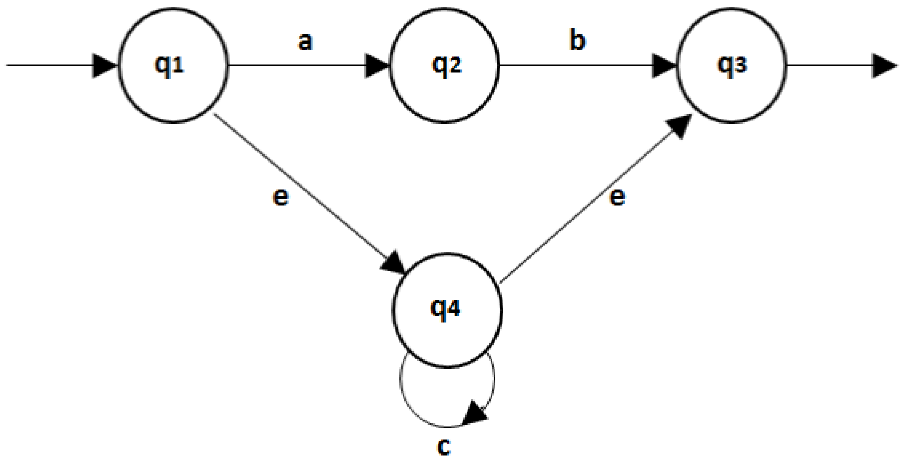
- Pro každý stav X je třeba vytvořit, množinu *e*-následníků, tedy stavů, které jsou ze stavu X dosažitelné cestami složenými z *e*-hran. Každý stav je *e*-následníkem sebe samého !

Jak sestrojit ekvivalentní KA k NKA s *e*-hranami?

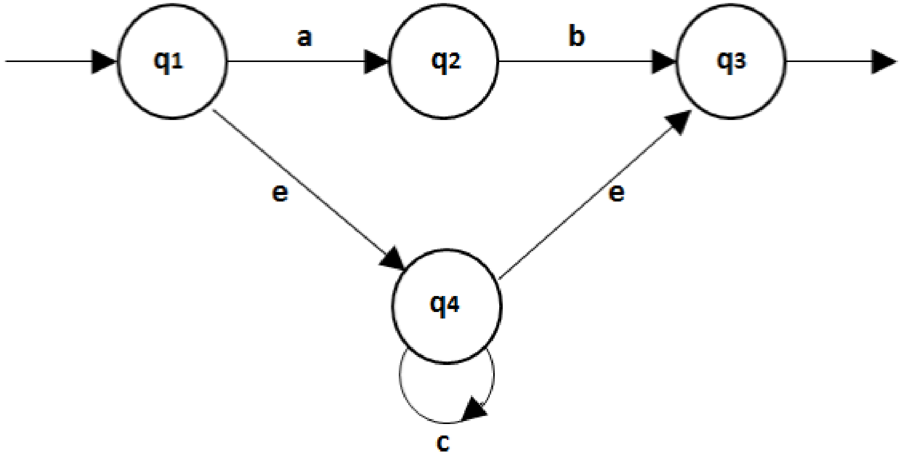
- Množiny *e*-následníků použijeme
  - k vytvoření přechodové tabulky ekvivalentního NKA bez *e*-hran tím, že stavy v množinách určujících hodnoty funkce δ nahradíme všemi jejich *e*-následníky
  - k vytvoření počátečního stavu ekvivalentního KA tím, že každý počáteční stav v množině *S* nahradíme všemi jeho *e*-následníky.

Jak sestrojit ekvivalentní KA k NKA s *e*-hranami - příklad

		a	b	C	e
$\rightarrow$	q_1	$\{q_2\}$	$\emptyset$	$\emptyset$	$\{q_4\}$
	q_2	$\emptyset$	$\{q_3\}$	$\emptyset$	$\emptyset$
$\leftarrow$	q_3	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
	q_4	$\emptyset$	$\emptyset$	$\{q_4\}$	$\{q_3\}$



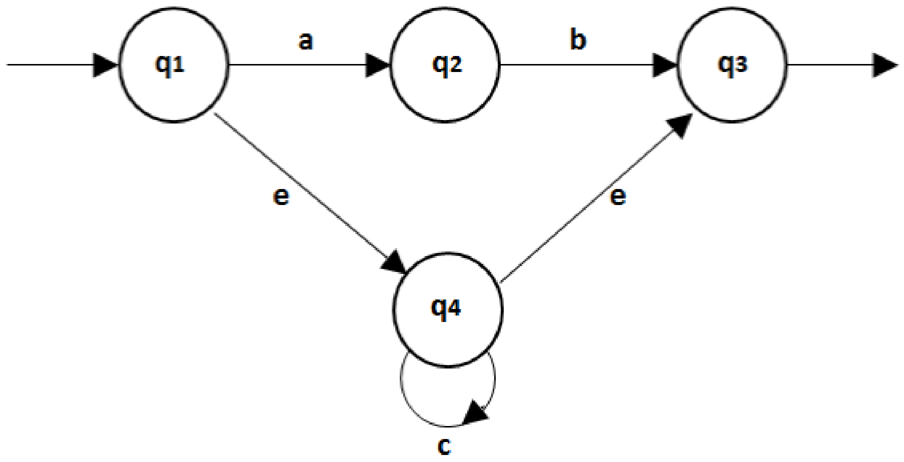
Jak sestrojit ekvivalentní KA k NKA s *e*-hranami - příklad



Jaké řetězce jsou tímto NKA akceptovány:

- prázdný řetězec *e*
- řetězec *ab*
- všechny řetězce c^n kde $n > 0$

Jak sestrojit ekvivalentní KA k NKA s *e*-hranami - příklad



	e-následníci
q_1	$\{q_1,q_3,q_4\}$
q_2	$\{q_2\}$
q_3	$\{q_3\}$
q_4	$\{q_3,q_4\}$

Ekvivalentní přechodová tabulka NKA bez e-hran:

		a	b	c
→	q_1	$\{q_2\}$	$\emptyset$	$\emptyset$
	q_2	$\emptyset$	$\{q_3\}$	$\emptyset$
←	q_3	$\emptyset$	$\emptyset$	$\emptyset$
	q_4	$\emptyset$	$\emptyset$	$\{q_3,q_4\}$

Počáteční stav deterministického ekvivalentu: $\{q_1,q_3,q_4\}$

Jak sestrojit ekvivalentní KA k NKA s *e*-hranami - příklad

Ekvivalentní přechodová tabulka NKA bez e-hran:

		a	b	c
→	q_1	$\{q_2\}$	$\emptyset$	$\emptyset$
	q_2	$\emptyset$	$\{q_3\}$	$\emptyset$
←	q_3	$\emptyset$	$\emptyset$	$\emptyset$
	q_4	$\emptyset$	$\emptyset$	$\{q_3,q_4\}$

Výsledný (D)KA:

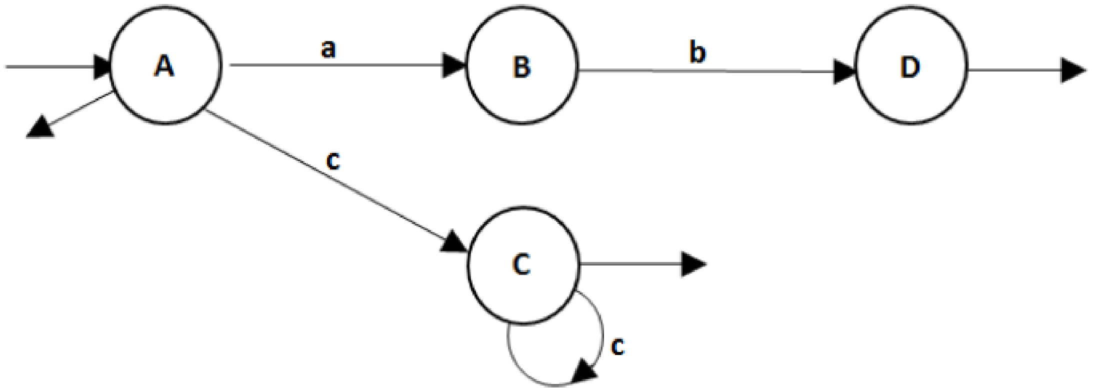
		A	b	c
↔	A $\{q_1, q_3, q_4\}$	B $\{q_2\}$	$\emptyset$	C $\{q_3, q_4\}$
	B $\{q_2\}$	$\emptyset$	D $\{q_3\}$	$\emptyset$
←	C $\{q_3, q_4\}$	$\emptyset$	$\emptyset$	C $\{q_3, q_4\}$
←	D $\{q_3\}$	$\emptyset$	$\emptyset$	$\emptyset$

Počáteční stav deterministického ekvivalentu: $\{q_1,q_3,q_4\}$

Jak sestrojit ekvivalentní KA k NKA s *e*-hranami - příklad

Výsledný (D)KA:

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



Je zřejmé, že tento KA akceptuje právě tyto řetězce:

- prázdný řetězec *e*
- řetězec *ab*
- všechny řetězce c^n kde $n > 0$

tedy stejný jazyk jako výchozí NKA.

<KIV> KATEDRA INFORMATIKY A VÝPOČETNÍ TECHNIKY

FAKULTA APLIKOVANÝCH VĚD ZÁPADOČESKÉ UNIVERZITY V PLZNI

Předmět KIV/TI - přednáška 5

Reprezentace regulárních jazyků

Ing. Václav Vais, Ph.D.

vais@kiv.zcu.cz

Regulární množiny

- Uvidíme, že regulární množina není nic jiného než (nám již známý) regulární jazyk (jazyk typu 3), tedy množina řetězců, ke které existuje konečný automat, jenž ji rozpoznává.
- Pojem regulární množina je jen výsledkem jiného přístupu k regulárnímu jazyku než na základě konečného automatu nebo gramatiky typu 3.
- Tento přístup následně umožňuje popis regulárních jazyků pomocí regulárních výrazů, tedy formou podobnou aritmetickým výrazům
- Regulární množiny a regulární výrazy popsal S. C. Kleene [klejný] (1956)

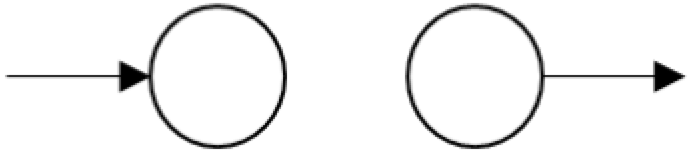
Regulární množiny

Regulární množina nad abecedou Σ je definována rekurzivně takto

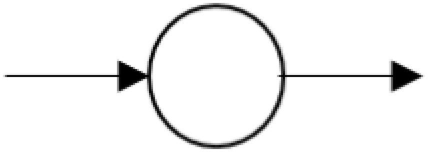
1. $\emptyset$ je regulární množina
2. $\{e\}$ je regulární množina
3. $\{a\}$ je regulární množina $\forall\, a \in \Sigma$
4. Jsou-li P a Q regulární množiny, pak
    a. $P \cup Q$ je regulární množina
    b. $P \cdot Q$ je regulární množina
    c. P^* a Q^* jsou regulární množiny
5. Neexistují žádné jiné regulární množiny.

KA rozpoznávající elementární regulární množiny

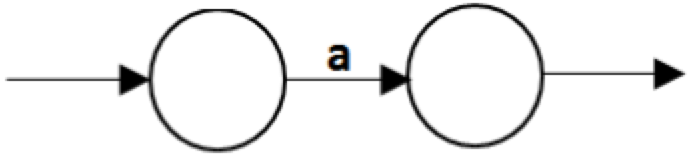
$L(A) = \emptyset$



$L(A) = \{e\}$

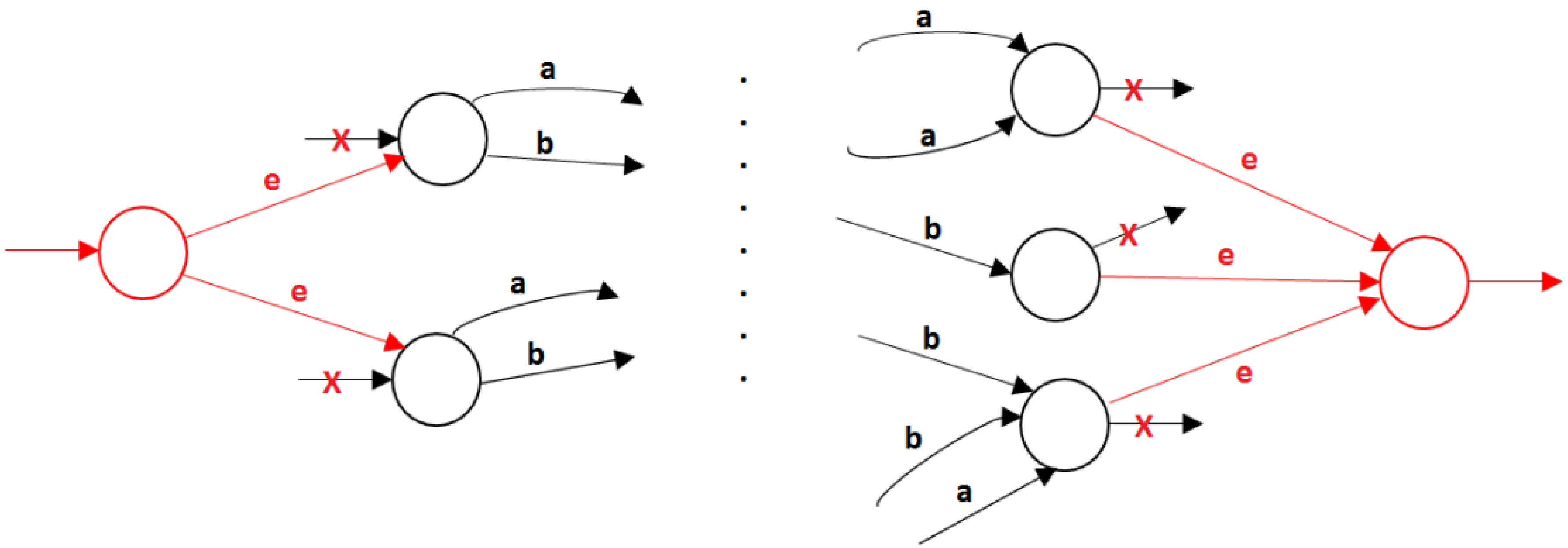


$L(A) = \{a\}$



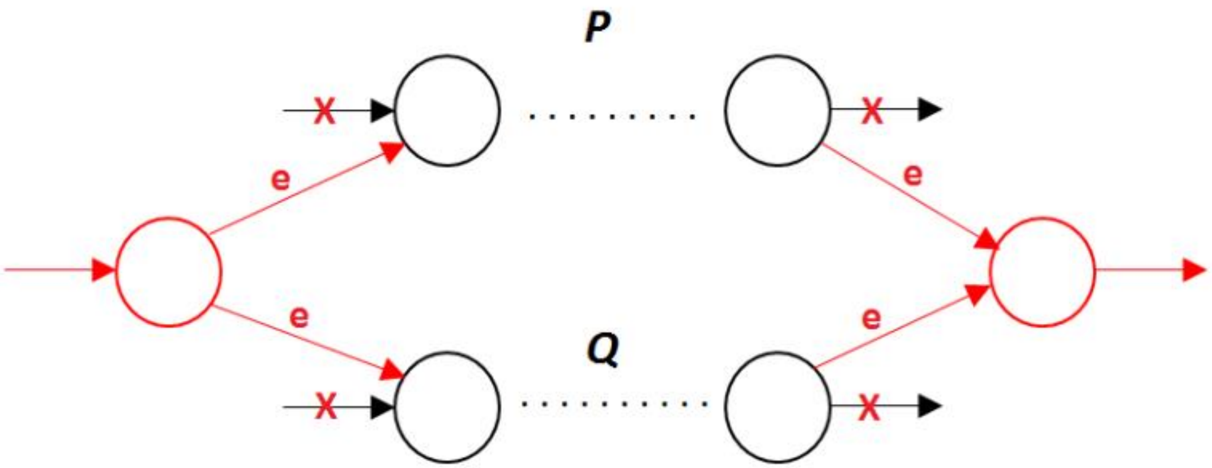
KA rozpoznávající regulární množiny vytvořené regulárními operacemi

- Bez újmy na obecnosti budeme předpokládat, že NKA má právě jeden počáteční a právě jeden koncový stav. Pokud ne, doplníme *e*-hrany:



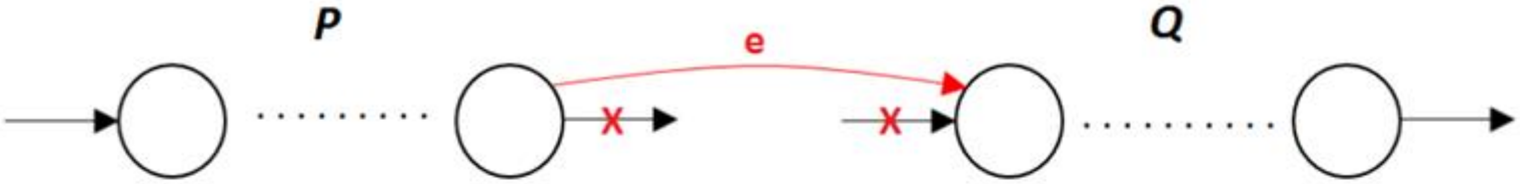
KA rozpoznávající regulární množiny vytvořené regulárními operacemi

Přechodový graf automatu, akceptujícího sjednocení regulárních množin $P \cup Q$:



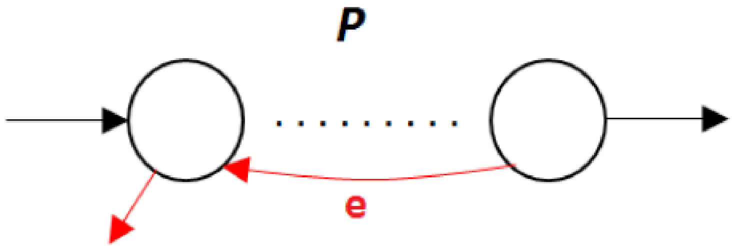
KA rozpoznávající regulární množiny vytvořené regulárními operacemi

Přechodový graf automatu, akceptujícího zřetězení regulárních množin $P \cdot Q$:



KA rozpoznávající regulární množiny vytvořené regulárními operacemi

Přechodový graf automatu, akceptujícího iteraci regulární množiny P^*:



Regulární výrazy (RV)

Regulární výraz nad abecedou Σ je definován rekurzivně takto

1. $\emptyset$ je regulární výraz označující regulární množinu $\emptyset$
2. e je regulární výraz označující regulární množinu $\{e\}$
3. a je regulární výraz je regulární množinu $\{a\} \quad \forall\, a \in \Sigma$
4. Jsou-li p a q regulární výrazy označující regulární množiny P a Q, pak
    a. $p + q$ je regulární výraz označující regulární množinu $P \cup Q$
    b. $p \cdot q$ je regulární výraz označující regulární množinu $P \cdot Q$
    c. p^* a q^* jsou regulární výrazy označující regulární množiny P^* a Q^*
5. Neexistují žádné jiné regulární výrazy.

Regulární výrazy (RV)

- Regulární výrazy se používají k popisu regulárních množin (zjednodušení zápisu).
- Terminologie používaná pro vztah mezi regulární množinou a regulárním výrazem:
  - regulární výraz *R* *označuje* regulární množinu *A*
  - regulární množina *A* *je hodnotou* regulárního výrazu *R*
- Značení:

$$A = \|R\|$$

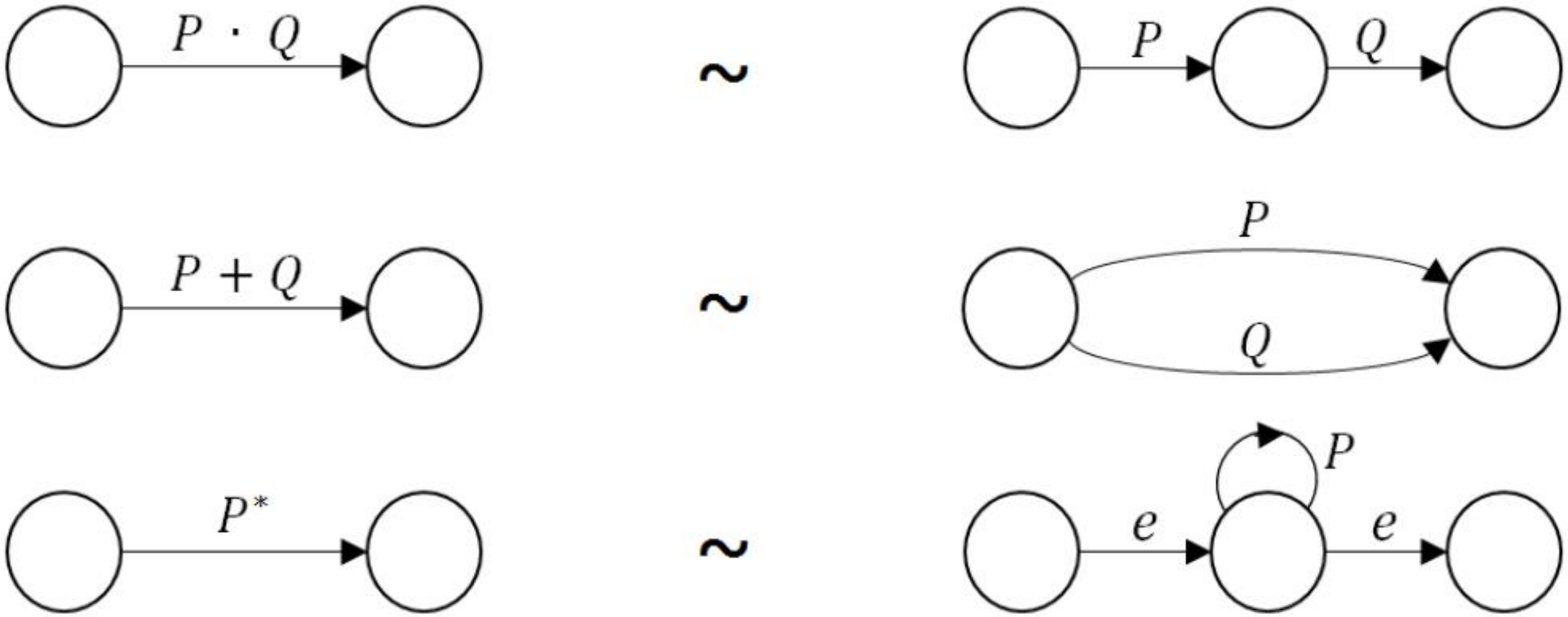
Jak sestrojit NKA akceptující jazyk popsaný RV?

- Postupným rozkladem regulárního výrazu R a odpovídající transformací zobecněného přechodového grafu.
- *Zobecněný přechodový graf* = přechodový graf, u něhož připustíme ohodnocení hran nejen symboly abecedy $a \in \Sigma$ a symbolem e, ale i regulárními výrazy nad abecedou Σ .
- Výchozí zobecněný přechodový graf:
    - počáteční stav S
    - koncový stav K
    - hrana z S do K ohodnocená regulárním výrazem R

Jak sestrojit NKA akceptující jazyk popsaný RV?

- Transformace grafu přidává nové stavy a nové přechodové hrany, čímž se zjednodušují regulární výrazy, kterými jsou hrany ohodnoceny.
- Transformace grafu končí, když je každá jeho hrana ohodnocena písmenem abecedy Σ nebo symbolem *e* .
- Výsledek transformace je přechodovým grafem NKA.
- Tento NKA akceptuje jazyk popsaný výchozím regulárním výrazem.

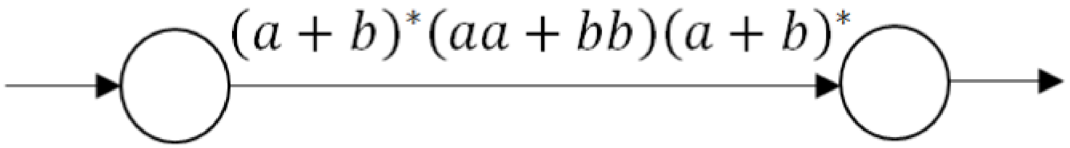
Schémata pro transformaci grafu



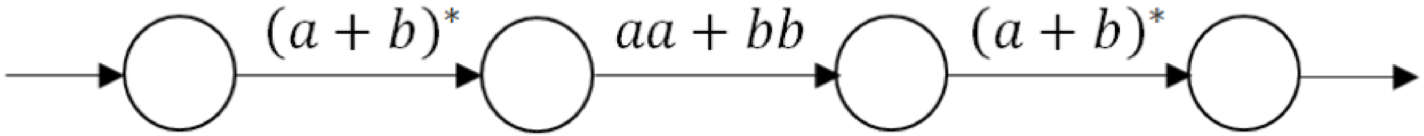
Přechod od RV k NKA - příklad

<u>Příklad</u>: Vytvořte konečný automat rozpoznávající jazyk $(a+b)^*(aa+bb)(a+b)^*$.

1. krok (vytvoření zobecněného přechodového grafu o dvou stavech)

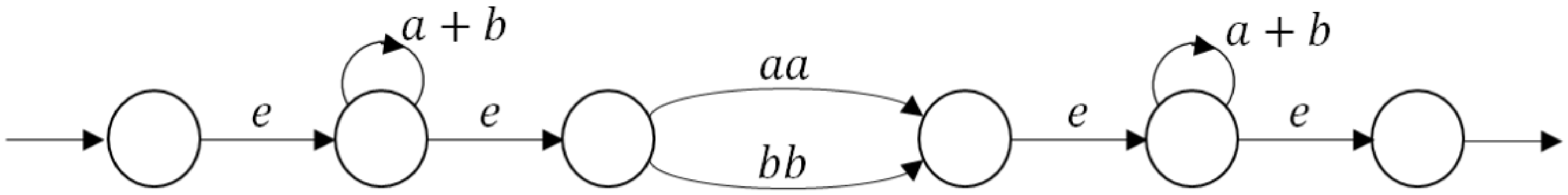


2. krok (transformace dvou zřetězení)

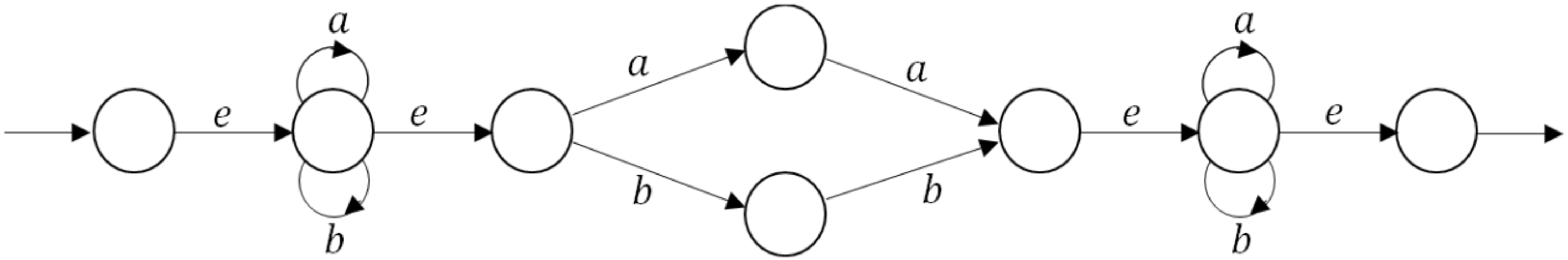


Přechod od RV k NKA - příklad

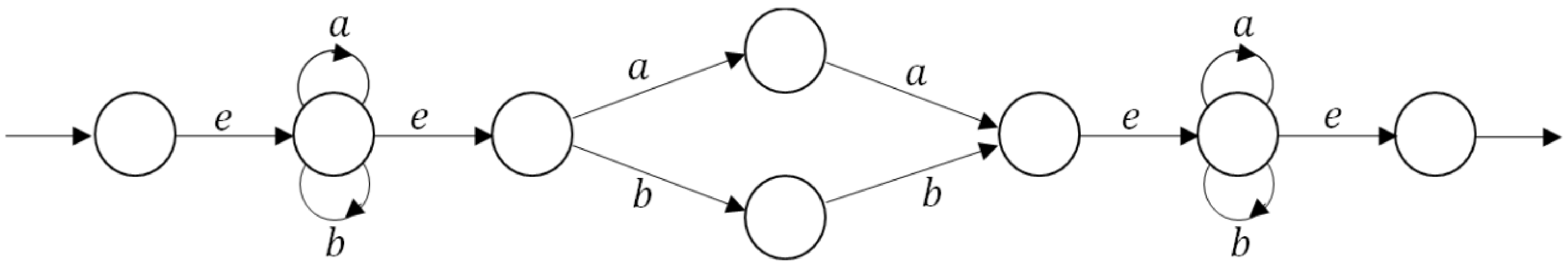
3. krok (transformace dvou iterací a součtu)



4. krok (transformace dvou součtů a dvou zřetězení)



Přechod od RV k NKA - příklad



- Každá hrana je označena písmenem vstupní abecedy nebo symbolem *e*, rozklad je ukončen.
- Automat je obecně nedeterministický s *e*-hranami. Lze nalézt ekvivalentní deterministický automat. (Později)

Jak k NKA sestrojit ekvivalentní RV?

- Bez újmy na obecnosti předpokládáme NKA s jedním počátečním a jedním koncovým stavem.
- Postupně budeme eliminovat vrcholy a hrany přechodového grafu.
- Během procesu eliminace budou modifikována ohodnocení hran regulárními výrazy.
- Tímto procesem vytváříme sekvenci zobecněných přechodových grafů, které akceptují stejné množiny řetězců.
- Proces končí, když v grafu zbydou jen počáteční stav a koncový stav a jedna hrana mezi nimi.   Její ohodnocení = hledaný regulární výraz.

Schémata pro eliminaci hran

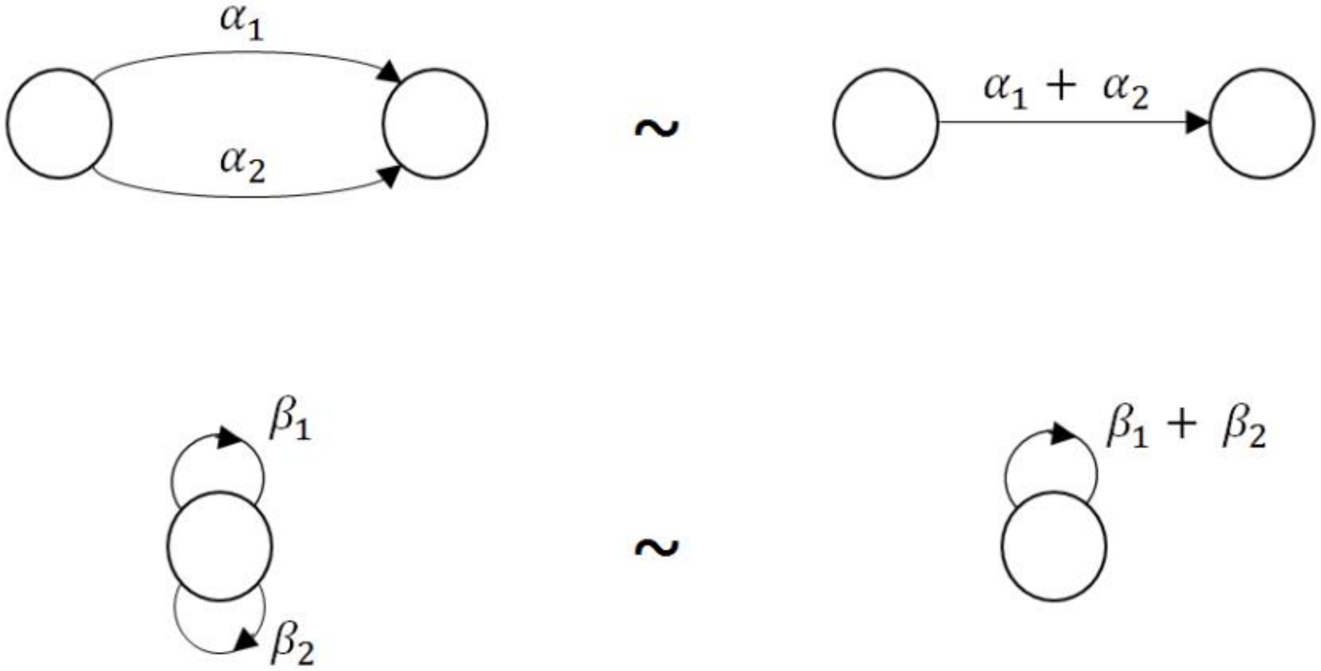
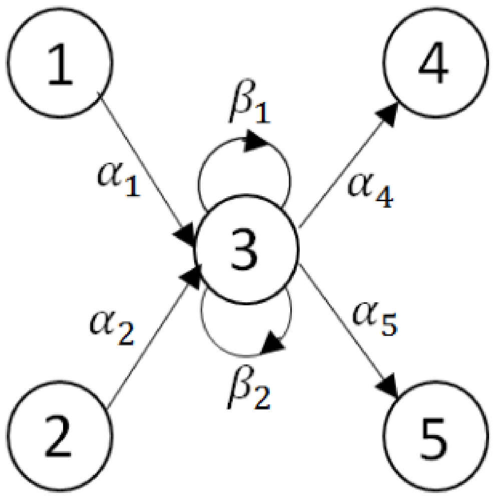
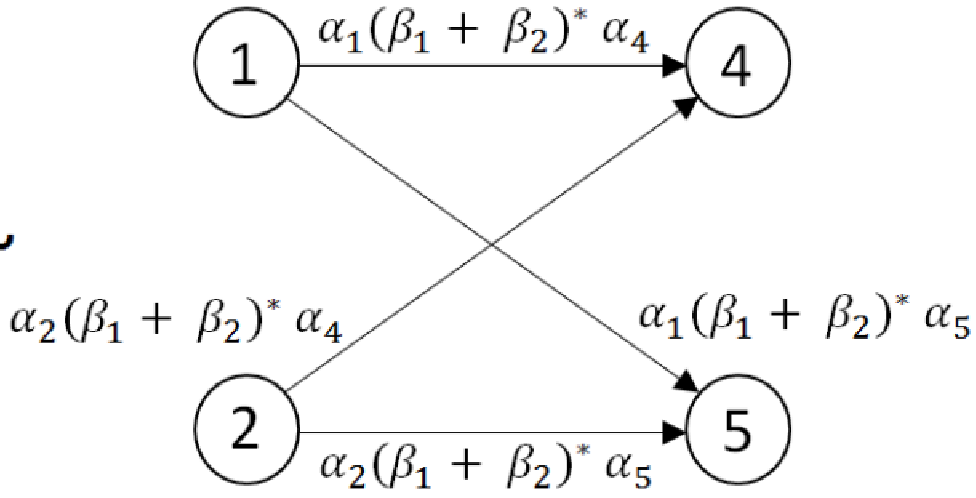


Schéma pro eliminaci vrcholů

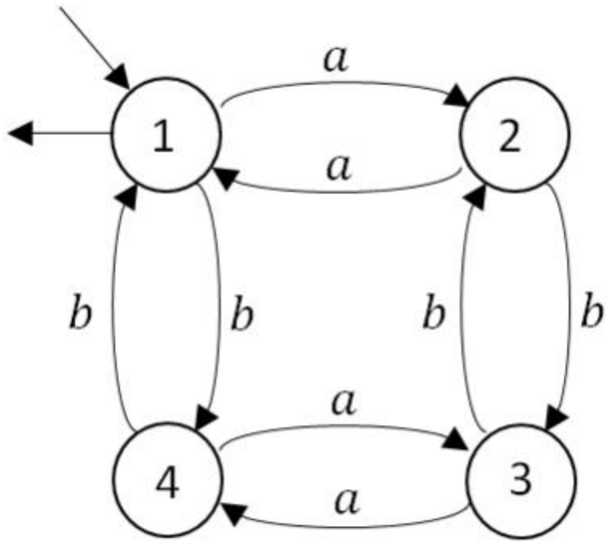


$\sim$



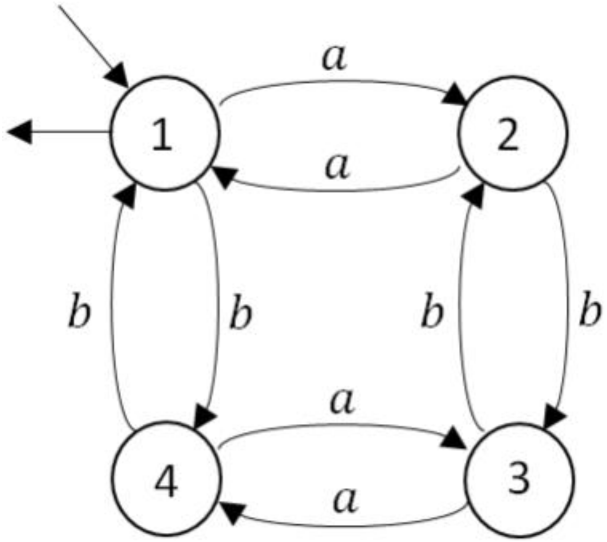
Přechod od NKA k RV - příklad

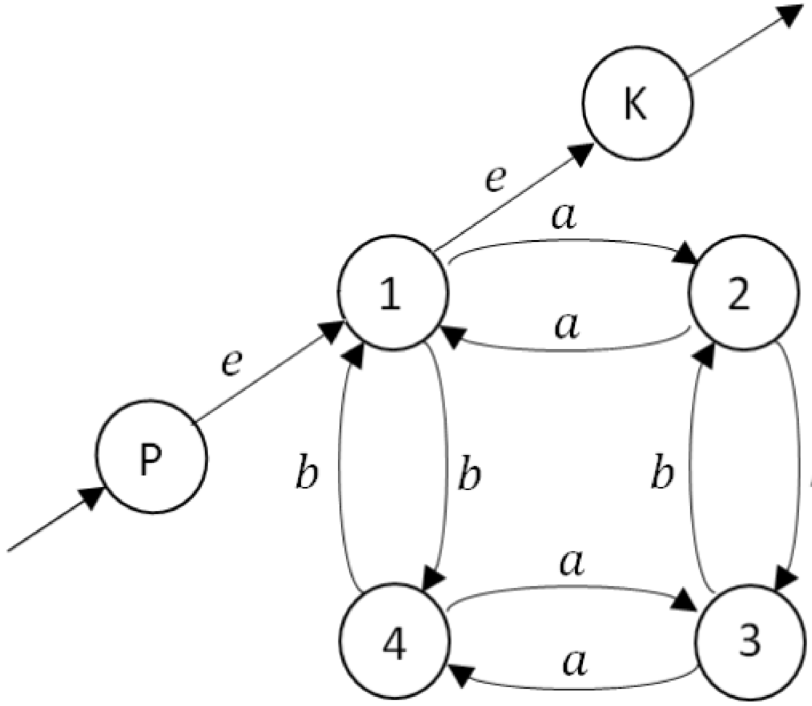
<u>Příklad</u>: Konečný automat A nad vstupní abecedou $\{a, b\}$ akceptuje řetězce obsahující sudý počet znaků a a sudý počet znaků b (nulu považujeme za sudé číslo). Popište jazyk $L(A)$ regulárním výrazem.



Přechod od NKA k RV - příklad

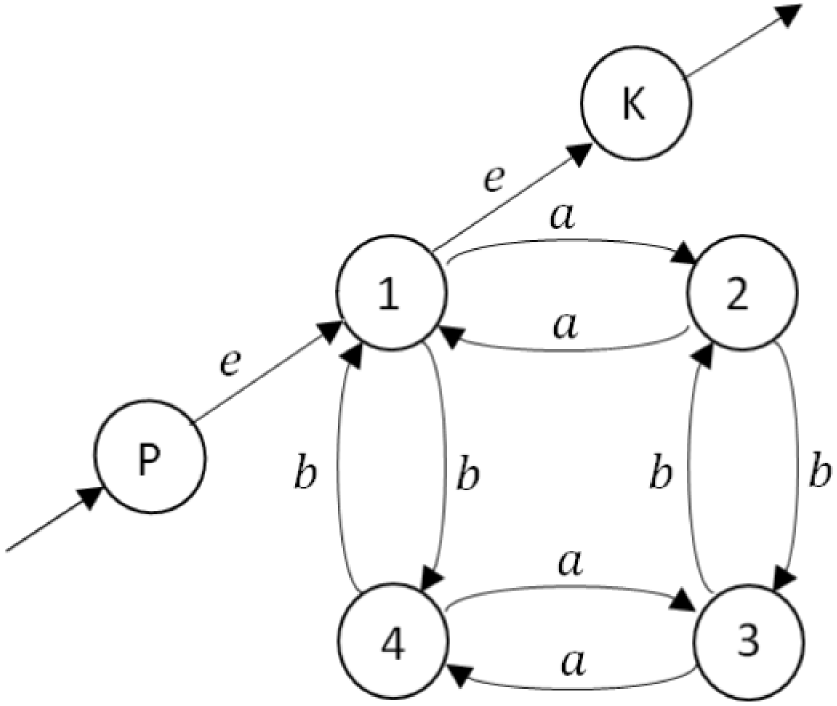
Krok 1 – vytvoření samostatného počátečního a samostatného koncového stavu :





Přechod od NKA k RV - příklad

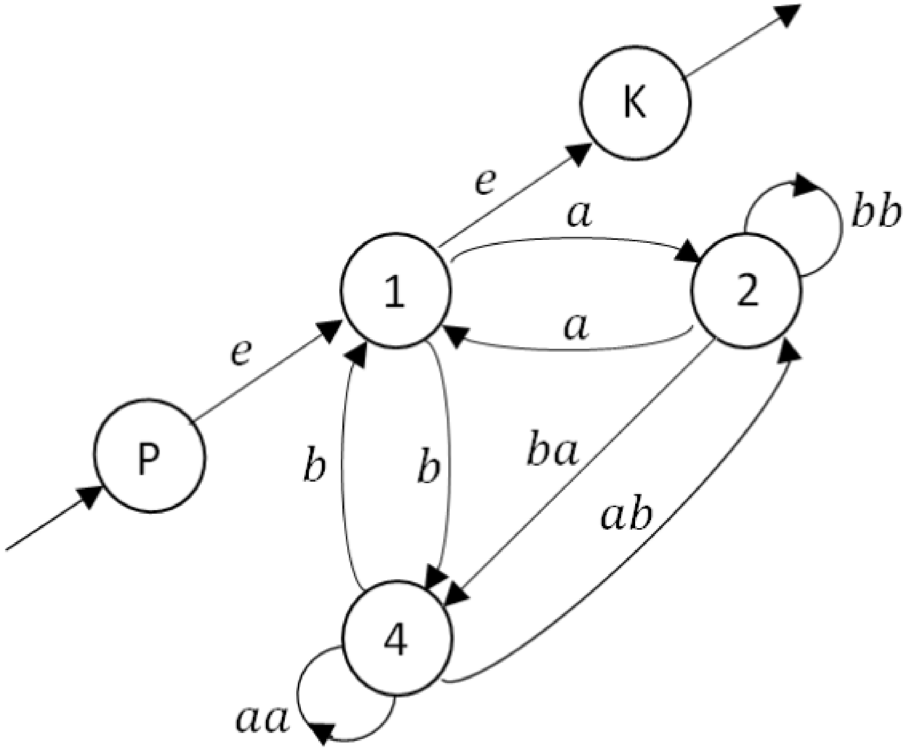
Krok 2 – eliminace vrcholu 3:



Cesty přes vrchol 3:

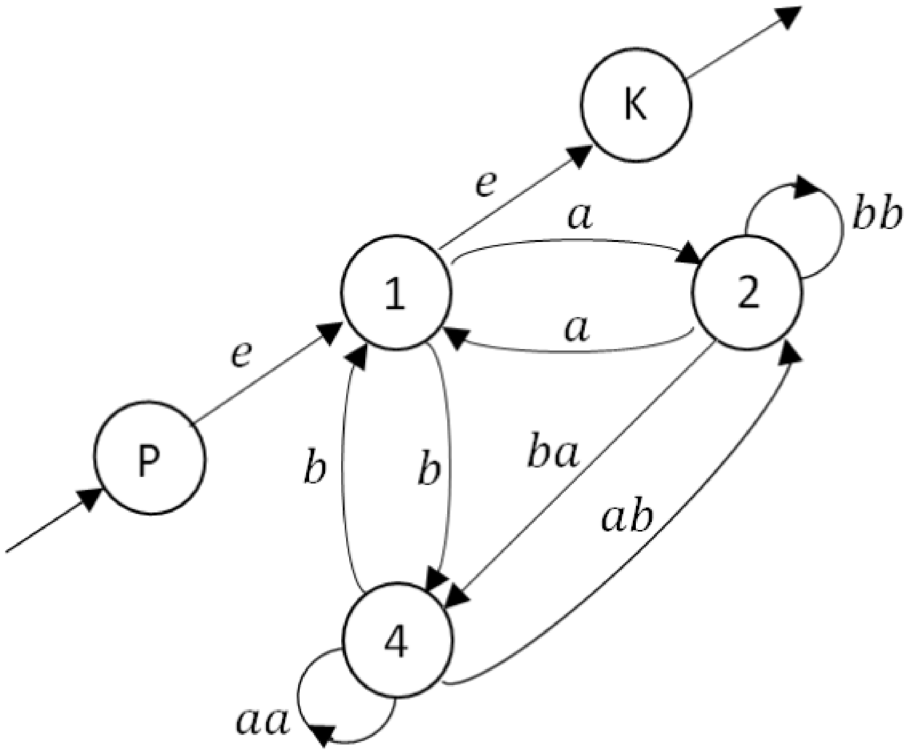
2 → 2: bb , 2 → 4: ba

4 → 2: ab , 4 → 4: aa



Přechod od NKA k RV - příklad

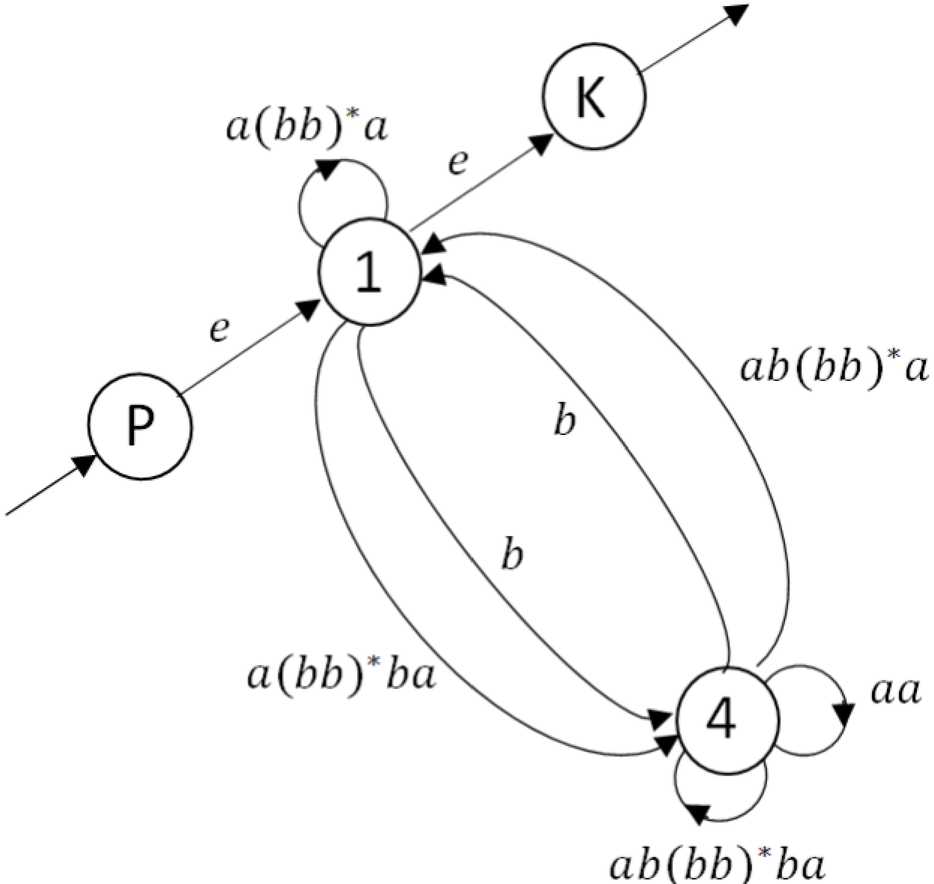
Krok 3 – eliminace vrcholu 2:



Cesty přes vrchol 2:

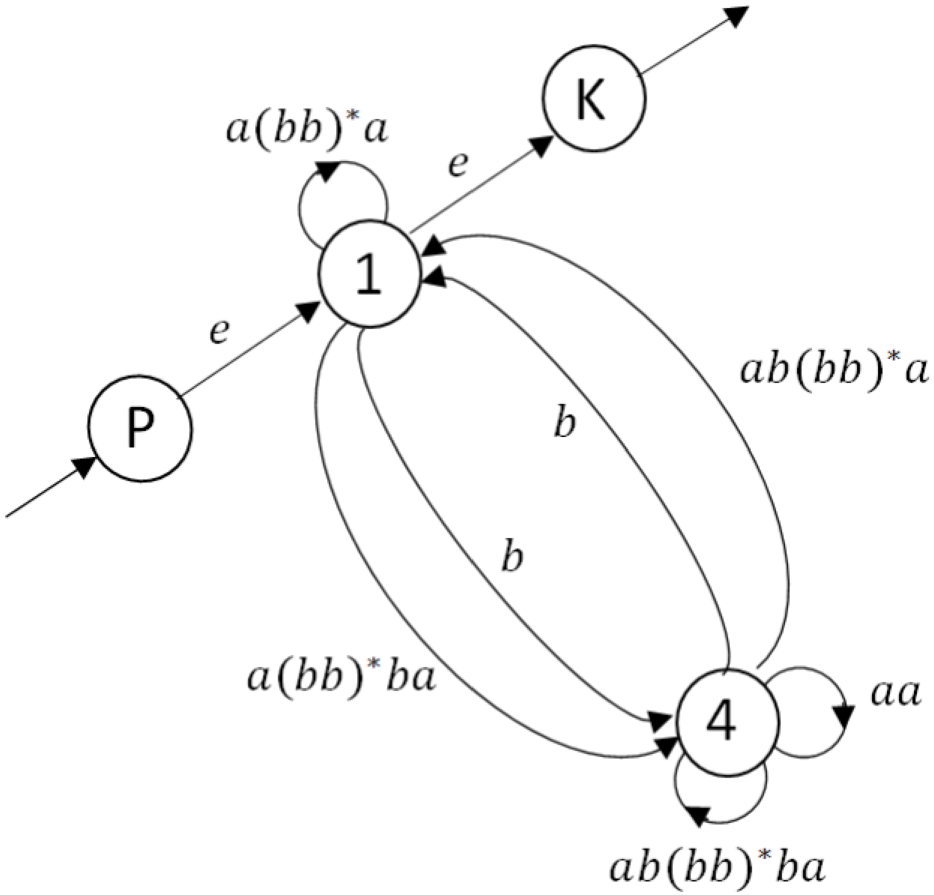
1 → 1: $a(bb)^*a$ 1 → 4: $a(bb)^*ba$

4 → 1: $ab(bb)^*a$ 4 → 4: $ab(bb)^*ba$



Přechod od NKA k RV - příklad

Krok 4 – eliminace paralelních hran :

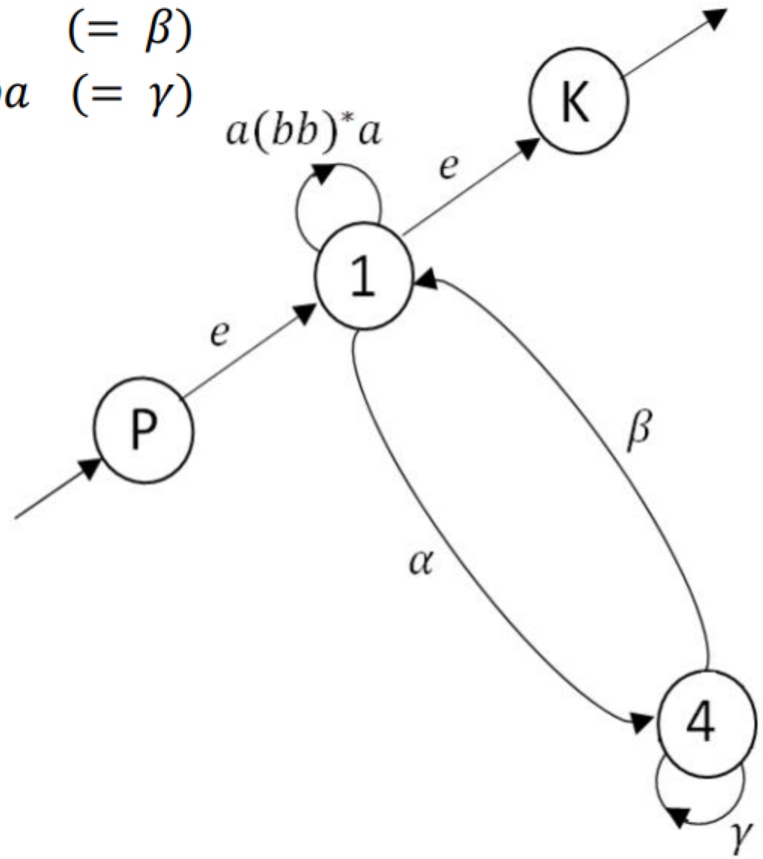


Hrany mezi vrcholy 4 a 1:

1 → 4: $b + a(bb)^*ba$ $(= \alpha)$

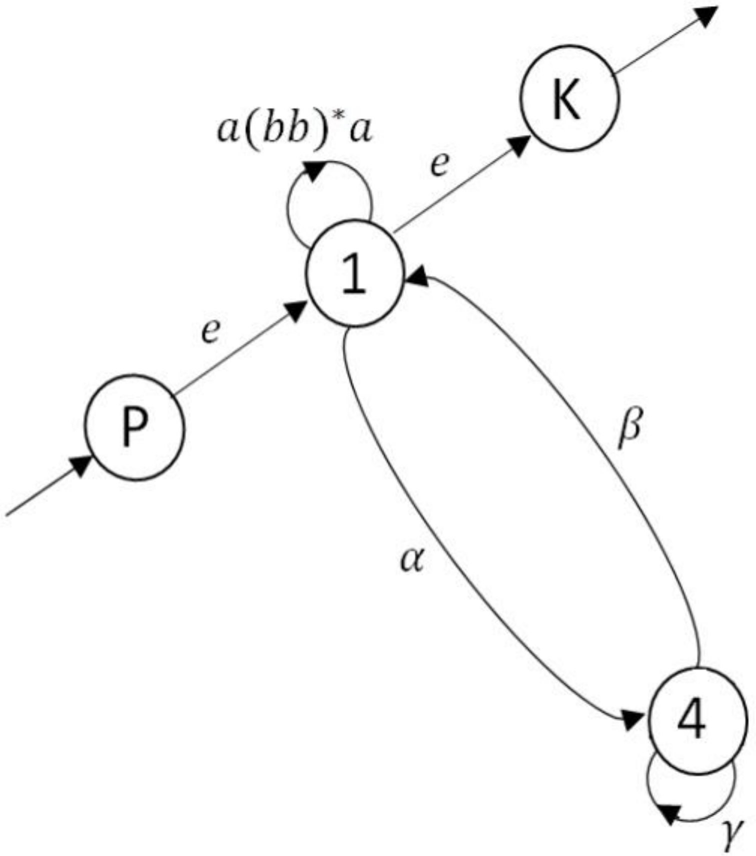
4 → 1: $b + ab(bb)^*a$ $(= \beta)$

4 → 4: $aa + ab(bb)^*ba$ $(= \gamma)$



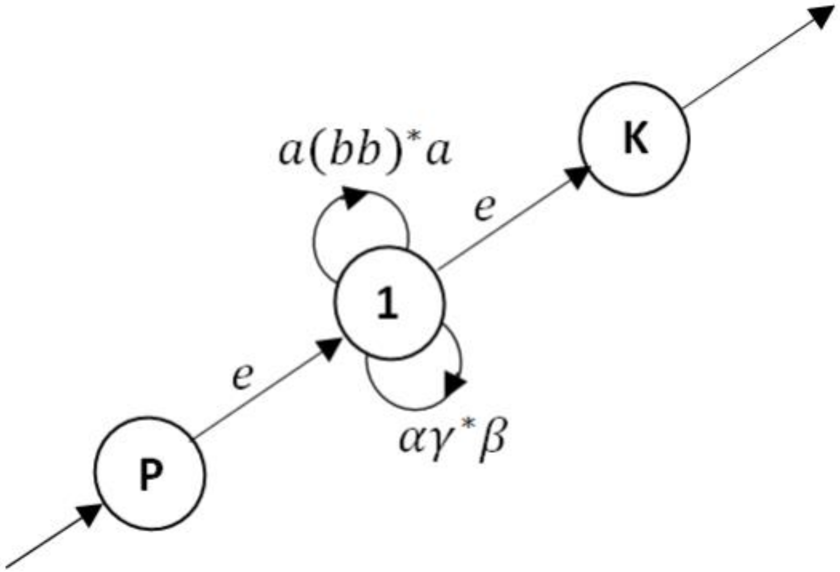
Přechod od NKA k RV - příklad

Krok 5 – eliminace vrcholu 4:



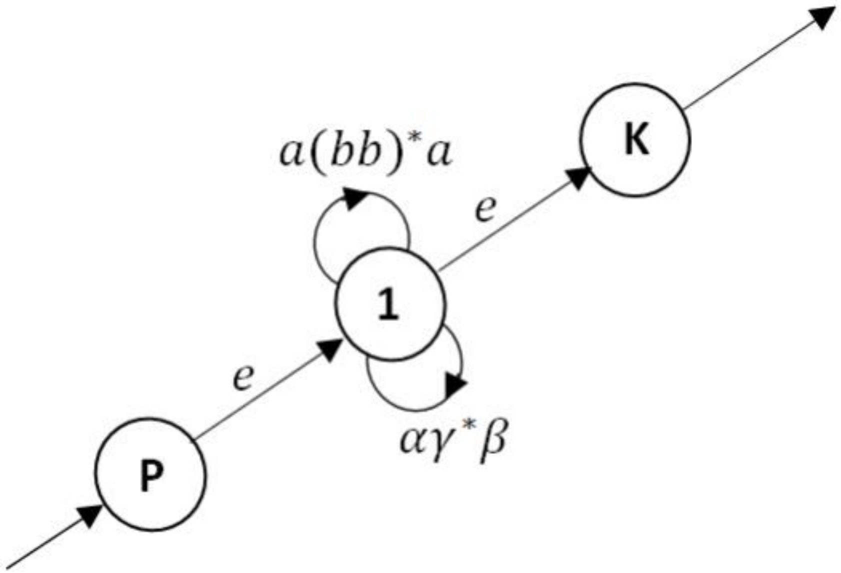
Cesty přes vrchol 4:

1 → 1: $\alpha\gamma^*\beta$



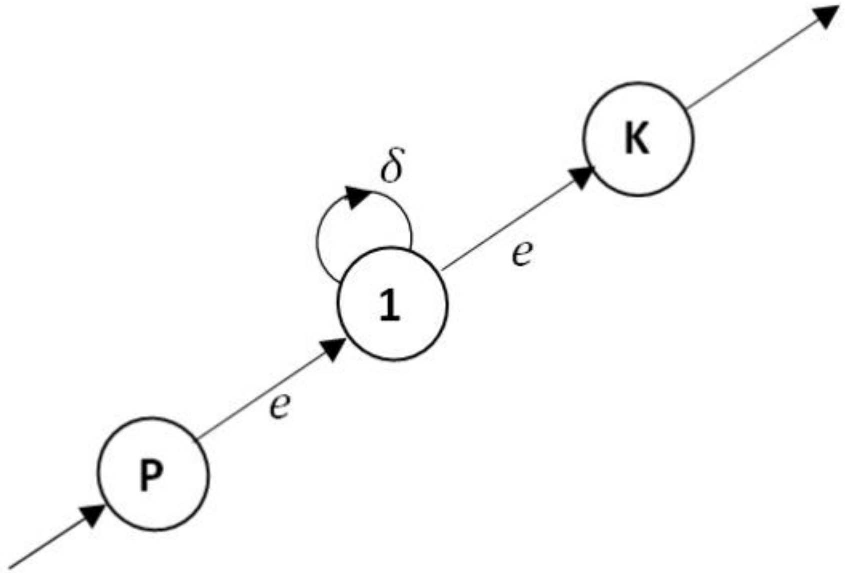
Přechod od NKA k RV - příklad

Krok 6 - eliminace paralelní smyčky v 1:



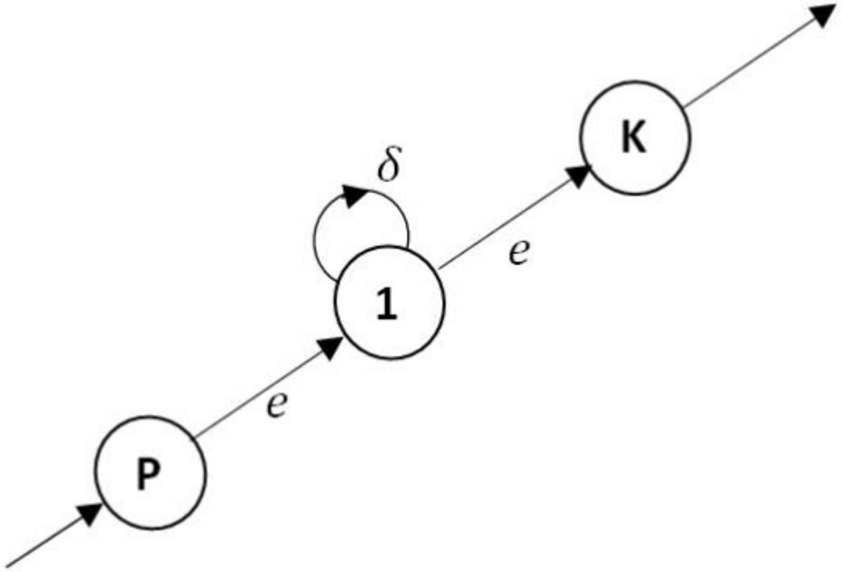
Smyčka ve vrcholu 1:

1 → 1: $a(bb)^*a + \alpha\gamma^*\beta = \delta$



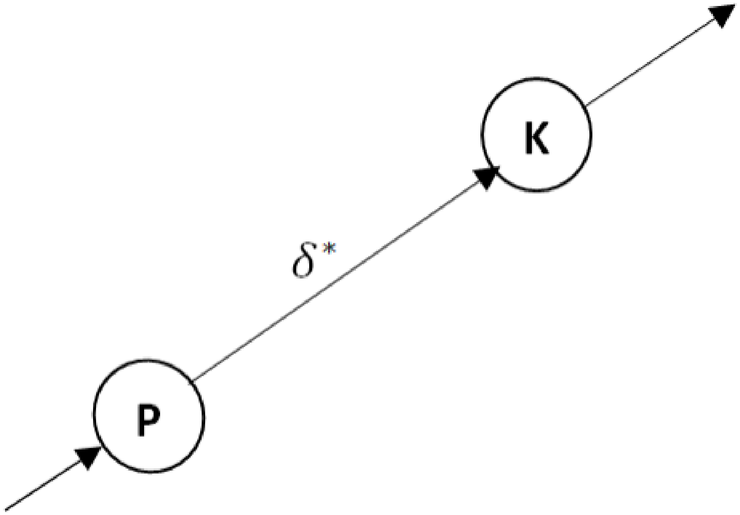
Přechod od NKA k RV - příklad

Krok 7 - eliminace vrcholu 1:

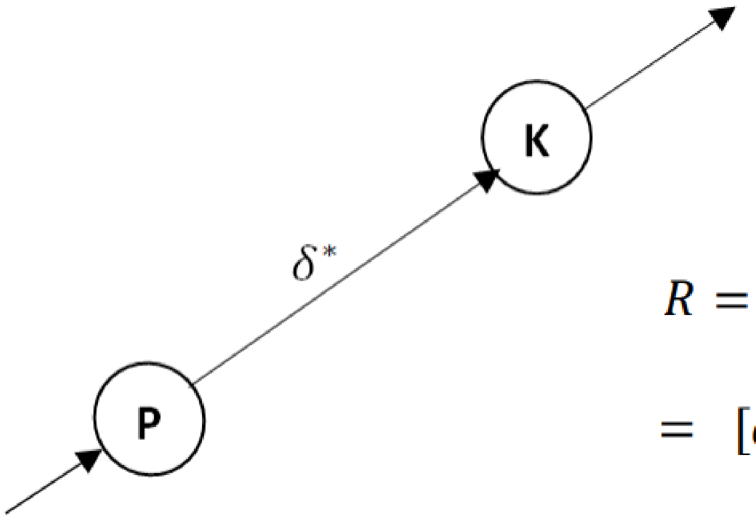


Cesta přes vrchol 1:

P → K: δ^*



Přechod od NKA k RV - příklad



Hledaný regulární výraz má tedy podobu

$$R = \delta^* = [a(bb)^*a + \alpha\gamma^*\beta)]^* =$$

$$= [a(bb)^*a + (b + a(bb)^*ba)(aa + ab(bb)^*ba)^*(b + ab(bb)^*a)]^*$$

Přechod od NKA k RV – závěr k příkladu

- Obecně platí, že tvar výsledného RV závisí na pořadí eliminace vrcholů.
- Při jiném pořadí eliminace vrcholů získáme RV jiný, ale ekvivalentní v tom smyslu, že bude popisovat stejný jazyk (pokud neuděláme při transformaci chybu).
- S RV lze provádět symbolické úpravy podobně jako s aritmetickými výrazy, např. vytýkání (ale POZOR, vytýkání zleva či zprava), „roznásobování“ zleva či zprava, apod.
- Regulární výrazy jsou velice variabilní a není vždy jednoduché dokázat, že dva různé RV jsou ekvivalentní.

Regulární výrazy ve skriptovacích jazycích

- Jsou rozšířením Kleeneho konceptu regulárních výrazů, také popisují množiny řetězců.
- Zavádí další operátory a metajazykové symboly.
- Lze je chápat jako vzory (masky), kterým zpracovávaný řetězec buď vyhovuje nebo nevyhovuje.
- Standardně se používají např. k syntaktické kontrole dat zadávaných ve formulářích (e-mailová adresa, PSČ, bankovní spojení, …..).
- Práci s těmito regulárními výrazy umožňuje řada programovacích jazyků (C#, Java, Visual Basic, .NET, Perl, PHP, Javascript, ….).

RV ve skriptovacích jazycích - ilustrace

Metaznaky pro zástupné symboly a opakování				
Metaznak	**Význam metaznaku**	**Příklad regulárního výrazu**	**Příklady vyhovujících vstupních řetězců**	**Příklady nevyhovujících vstupních řetězců**
.	Libovolný znak (právě jeden)	t.p	step tipovat utopí	tis lep terpentýn
*	Předchozí znak se vyskytne 0x až ∞x	r*m	zarmoutit grrrrm! ataman	figura
+	Předchozí znak se vyskytne 1x až ∞x	s+t	pssst! kostival	blatník
?	Předchozí znak se vyskytne 0x až 1x	k?t	pakt port vektor	Vekktor

RV ve skriptovacích jazycích - ilustrace

Metaznaky pro ukotvení pozice v řetězci				
Metaznak	**Význam metaznaku**	**Příklad regulárního výrazu**	**Příklady vyhovujících vstupních řetězců**	**Příklady nevyhovujících vstupních řetězců**
^	Pevný začátek řetězce	^Pos	Poslouchám Poseidon	Neposlouchej Porta
$	Pevný konec řetězce	rok$	nový rok Snížili úrok	Rok na vsi Dva roky poté Hrajte taroky

RV ve skriptovacích jazycích - ilustrace

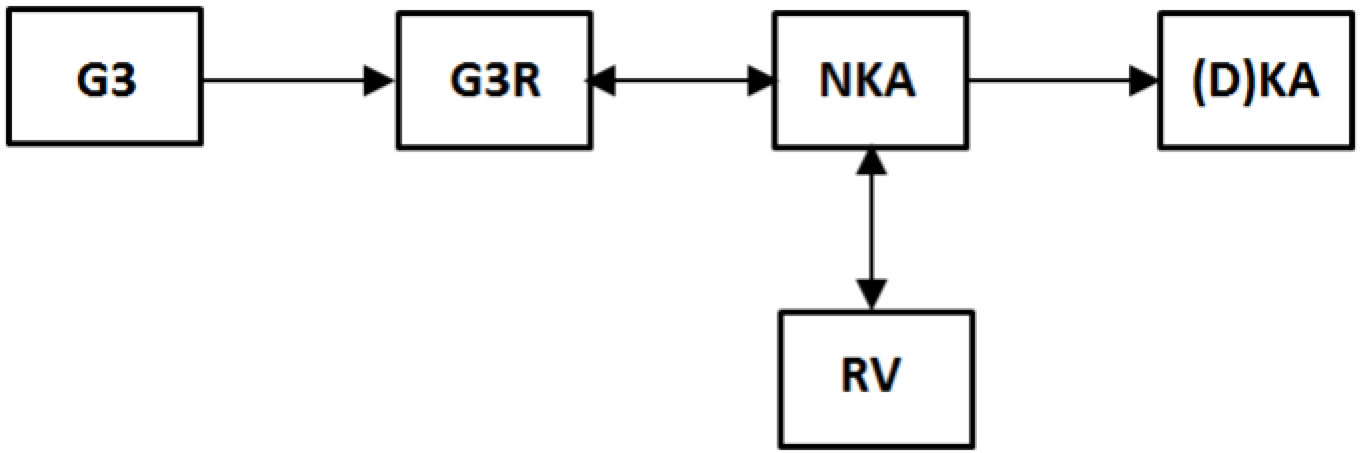
Metaznaky pro výčtové množiny a řetězce				
Metaznak	**Význam metaznaku**	**Příklad regulárního výrazu**	**Příklady vyhovujících vstupních řetězců**	**Příklady nevyhovujících vstupních řetězců**
[]	Libovolný znak z výčtu znaků v závorce	[0123456789]	-16 1 ks máslo Peugeot 308	jedno máslo
[*d*-*h*]	Libovolný znak z množiny znaků definované rozsahem od znaku d až po znak h (rozsahy lze řetězit)	[A-M]	Dovolená Malaga 156A23 B007	-254 agent 007 okresní přebor Pražský výběr
[^d-h]	Libovolný znak **mimo** množinu znaků definovanou rozsahem od znaku d až po znak h (rozsahy lze řetězit)	[^a-z0-9]	ALFA C&K VOCAL	ALFa C&K F.J.1.
()	Celý řetězec v závorce	(slov)	oslovený slovenský máslový	složený úlovek solventní

Ekvivalentní tvrzení o regulárních jazycích nad množinou Σ^*

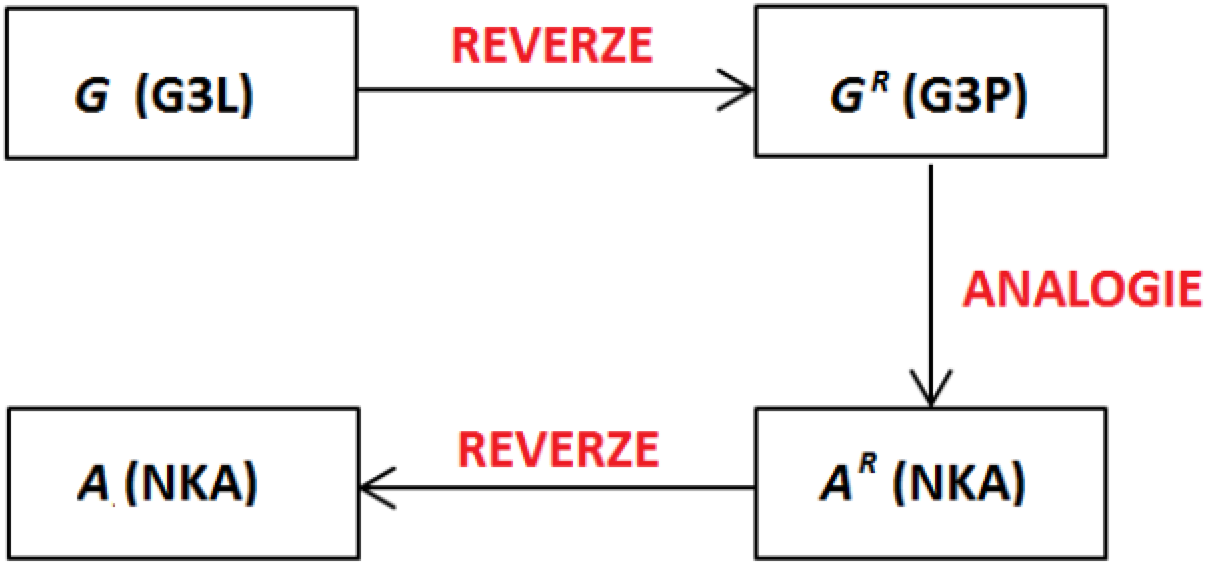
- Jazyk *L* je regulární.
- Jazyk *L* lze popsat regulárním výrazem.
- Existuje gramatika *G* typu 3 taková, že *L* = *L*(*G*).
- Existuje nedeterministický konečný automat *A* takový, že *L* = *L*(*A*).
- Existuje deterministický konečný automat *A* takový, že *L* = *L*(*A*).
- Na množině Σ^* existuje pravá kongruence konečného indexu taková, že *L* je sjednocením některých tříd této kongruence (Nerodova věta).

  (pravá kongruence je relace ekvivalence s konečným počtem tříd a s vlastností $(u \approx v) \Rightarrow (uw \approx vw)\ \forall u, v, w \in \Sigma^*$; počet tříd = počet stavů KA)

Převody mezi reprezentacemi jazyků typu 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

Příklad:

$$G: \quad \begin{aligned} S &\to aS \\ aS &\to bA \mid bba \\ A &\to abb \end{aligned}$$

$$G^R: \quad \begin{aligned} S &\to Sa \\ Sa &\to Ab \mid abb \\ A &\to bba \end{aligned}$$

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 $(A^R)^R = A$.