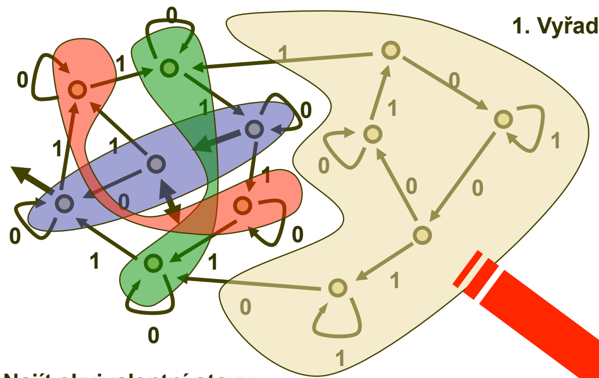
Automaty a gramatiky

Roman Barták, KTIML

bartak@ktiml.mff.cuni.cz
http://ktiml.mff.cuni.cz/~bartak

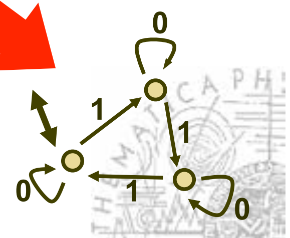
Pro připomenutí

1. Vyřadit nedosažitelné stavy



2. Najít ekvivalentní stavy
$\forall w \in X^*\ \delta^*(p,w) \in F \Leftrightarrow \delta^*(q,w) \in F$

3. Sestrojit podílový automat



Věta o isomorfismu reduktů

Pro libovolné dva redukované konečné automaty jsou následující dvě tvrzení ekvivalentní:

a) automaty jsou ekvivalentní,

b) automaty jsou isomorfní.

Důsledky:

Dva redukty libovolných dvou ekvivalentních konečných automatů se shodují až na isomorfismus.

Pro každý konečný automat je jeho redukt určen až na isomorfismus jednoznačně.

Ve třídě navzájem ekvivalentních konečných automatů existuje „minimální“ automat.

Důkaz věty o isomorfismu reduktů

a) isomorfismus $\Rightarrow$ ekvivalence (víme)

b) ekvivalence reduktů $\Rightarrow$ isomorfismus

hledáme homomorfismus $h: Q_1 \rightarrow Q_2$, který je „prostý a na“ tj.
pro každé $q \in Q_1$ hledáme právě jedno $p \in Q_2$

q je dosažitelný stav, tudíž $\exists u \in X^*\ \delta_1(q_1,u)=q$

položme $h(q) = \delta_2(q_2,u)$

je to skutečně funkce?

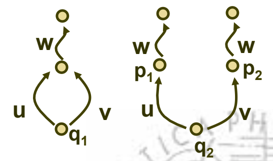
$\delta_1(q_1,u)=\delta_1(q_1,v) \Leftrightarrow \delta_2(q_2,u)=\delta_2(q_2,v)$ (*)

sporem, nechť $\delta_1(q_1,u)=\delta_1(q_1,v)\ \&\ \delta_2(q_2,u)\neq\delta_2(q_2,v)$

z A_1 víme $\forall w \in X^*\ uw \in L \Leftrightarrow vw \in L$

automaty jsou ekvivalentní, tedy $p_1 \sim p_2$

spor - automat A_2 je redukovaný



funkce h je „prostá a na“ (vlastnost (*))
$h(q_1)=q_2$ (pro $u=\lambda$)
$h(\delta_1(q,x)) = \delta_2(h(q),x)$ ($\delta_1(q_1,v)=q$, $u=vx$)
$q \in F_1 \Leftrightarrow h(q) \in F_2$ (pro $u \in L$ + ekvivalentní automaty)

Normalizace automatu

Jak najít isomorfismus automatů?

Normovaný tvar **automatu**

1) fixujme pořadí písmen v abecedě
2) počáteční stav označme 1
3) tabulku (automatu) vyplňujme po řádcích zleva doprava a pokud narazíme na nový stav, přiřadíme mu první volné číslo

Příklad:

		a	b
	A	B	A
↔	B	D	C
	C	A	D
←	D	A	B

		a	b
↔	1(B)	2	3
←	2(D)	4	1
	3(C)	4	2
	4(A)	1	4

Poznámky k redukci a ekvivalenci

Algoritmicky umíme řešit:

– zjištění ekvivalence automatů
  zredukujeme, znormalizujeme a porovnáme
– zjištění zda $L(A)=\emptyset$
  žádný koncový stav není dosažitelný
– zjištění zda $L(A)=X^*$
  po redukci dostaneme jednostavový automat (s koncovým stavem)

Umíme najít nejkratší slovo rozlišující dva stavy

$p \sim^i q \ \& \ \neg\, p \sim^{i+1} q$

$\exists a_1 \in X \quad \delta(p,a_1) \sim^{i-1} \delta(q,a_1) \ \ \& \ \ \neg\, \delta(p,a_1) \sim^i \delta(q,a_1)$

…

iterací najdeme slovo $a_1 \dots a_{i+1}$

Slovo odlišující dva stavy

		0	1	$\mathfrak{R}_0$	0	1	$\mathfrak{R}_1$	0	1	$\mathfrak{R}_2$
	a	a	b	A	A	A	A	A	B	A
	b	c	a	A	C	A	B	C	A	B
←	c	c	e	C	C	C	C	C	E	C
	d	e	d	A	C	A	B	E	B	D
←	e	e	d	C	C	A	E	E	B	E

Jak je dlouhé nejkratší slovo rozlišující stavy „b“ a „d“?

2 znaky

A jaké je to slovo?

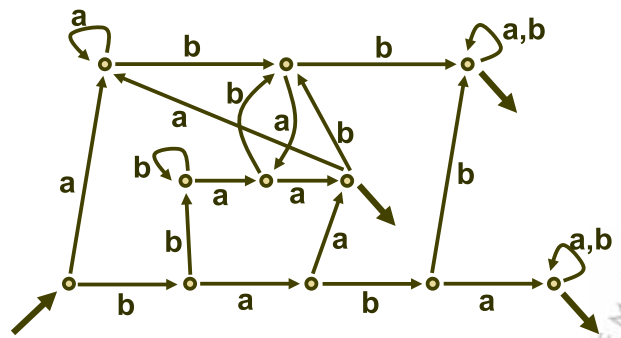
01 nebo 10

Trochu motivace

Dosud: Stav a písmeno jednoznačně určuje další stav!

Příklad:

$L = \{ w \mid w=babau \lor w=uabbv \lor w=ubaa,\ u,v \in \{a,b\}^* \}$

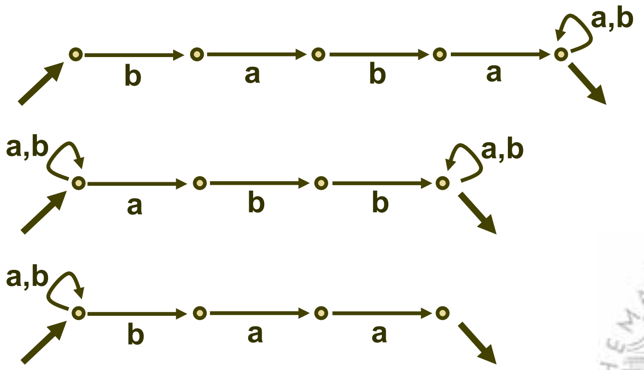


Úvod do nedeterminismu

Stav a písmeno určuje množinu možných dalších stavů!

Příklad:

$L = \{ w \mid w=babau \vee w=uabbv \vee w=ubaa,\ u,v \in \{a,b\}^* \}$



Nedeterministický konečný automat

Nedeterministickým konečným automatem nazýváme pětici A = (Q,X,δ,S,F), kde:

- Q - konečná neprázdná množina stavů (*stavový prostor*)
- X - konečná neprázdná množina symbolů (*vstupní abeceda*)
- δ - zobrazení $Q \times X \rightarrow P(Q)$ (*přechodová funkce*)
- $S \subseteq Q$ (*množina počátečních stavů*)
- $F \subseteq Q$ (*množina přijímacích stavů*)

Reprezentace:

stavový diagram, tabulka, stavový strom

Jak se počítá s nedeterminismem?

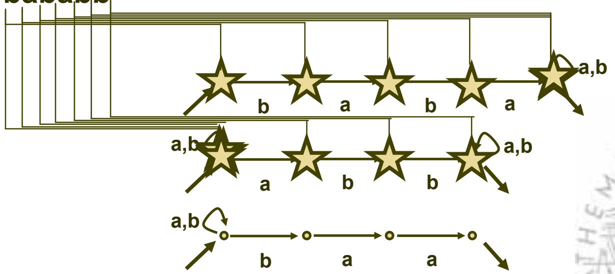
Slovo $w = x_1 \ldots x_n$ je *přijímáno* nedeterministickým konečným automatem, jestliže existuje posloupnost stavů $q_1,\ldots,q_{n+1}$ taková, že:

$q_1 \in S$, $q_{i+1} \in \delta(q_i, x_i)$ pro $i=1\ldots n$, $q_{n+1} \in F$.

Prázdné slovo je přijímáno právě když $S \cap F \neq \emptyset$

Přijímajících výpočtů pro dané slovo může být více!

Př. bababb



Determinismus vs. nedeterminismus

Konečný automat je speciálním případem nedeterministického konečného automatu!

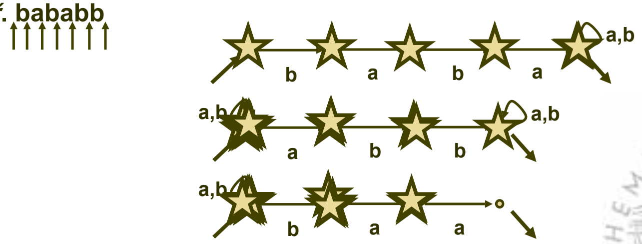
Důsledek: Jazyky rozpoznávané konečnými automaty jsou rozpoznávané nedeterministickými konečnými automaty.

Platí i obrácené tvrzení?

Zkusme to!

- potřebujeme postupovat systematicky a s konečnou pamětí
- pomocí značek na stavech simulujeme všechny možné výpočty
  - tzv. podmnožinová konstrukce

Př. bababb



Převod nedeterminismu na determinismus

Věta: Je-li A nedeterministický konečný automat, potom lze sestrojit konečný automat B takový, že L(A)=L(B).

Důkaz: (podmnožinová konstrukce)

nechť $A = (Q,X,\delta,S,F)$

potom definujme $B = (P(Q),X,\delta',S,F')$, kde

$F' = \{ K \mid K \in P(Q), K \cap F \neq \emptyset \}$

$\delta'(K,x) = \bigcup_{q \in K} \delta(q,x)$

1) B je definován korektně

2) L(A)=L(B)?

$\lambda \in L(A) \Leftrightarrow S \cap F \neq \emptyset \Leftrightarrow S \in F' \Leftrightarrow \lambda \in L(B)$

$L(A) \subseteq L(B)$

$w= x_1 \dots x_n \in L(A) \Leftrightarrow \exists q_1,\dots,q_{n+1} \in Q \;\; q_1 \in S,\ q_{i+1} \in \delta(q_i, x_i),\ q_{n+1} \in F$

položme $K_1=S \;\; (q_1 \in K_1),\ K_{i+1}= \delta'(K_i,x_i) \;\; (q_{i+1} \in K_{i+1})$, potom $K_{n+1} \in F'$

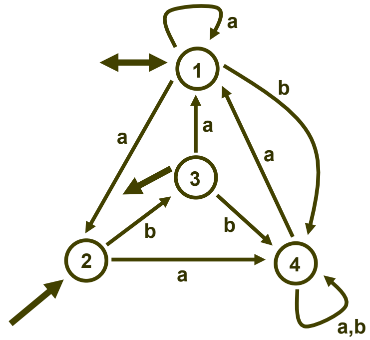
$L(B) \subseteq L(A)$

$w= x_1 \dots x_n \in L(B) \Leftrightarrow \exists K_1,\dots,K_{n+1} \in P(Q) \;\; K_1=S,\ K_{i+1}=\delta'(K_i, x_i),\ K_{n+1} \in F'$

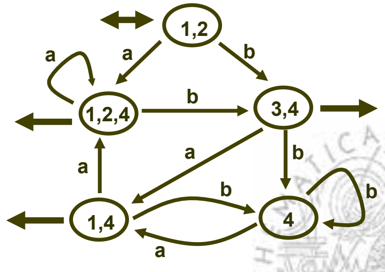
vezměme $q_{n+1} \in F \cap K_{n+1},\ q_i \in K_i$ tž. $q_{i+1} \in \delta(q_i,x_i), \dots,\ q_1 \in K_1=S$

Ukázka převodu

		a	b
↔	1	1,2	4
→	2	4	3
←	3	1	4
	4	1,4	4



		a	b
↔	{1,2}	{1,2,4}	{3,4}
←	{1,2,4}	{1,2,4}	{3,4}
←	{3,4}	{1,4}	{4}
←	{1,4}	{1,2,4}	{4}
	{4}	{1,4}	{4}



Poznámky k nedeterminismu

Význam:

- teoretický (např. při převodu gramatik na automaty)
- praktický (zjednodušení návrhu automatu)

U konečných automatů vede nedeterminismus ke stejné třídě jazyků jako determinismus.

- neplatí obecně (zásobníkové automaty)!

Převod na determinismus znamená (až) exponenciální nárůst počtu stavů ($Q \rightarrow P(Q)$).

- obecně je tento nárůst nezbytný!
  - $L_n = \{ w \mid w \in \{0,1\}^*, w=u1v, |v|=n-1 \}$
- není potřeba explicitně převádět.

Existují také *zobecněné nedeterministické automaty*

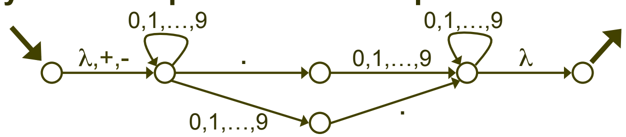
λ-přechod: změna stavu bez čtení vstupu

λ-přechody

Automat může změnit stav bez čtení symbolu.

Hodí se pro zjednodušení popisu automatu.

Příklad: rozpoznání čísla s desetinou tečkou s možností vynechání 0 před/za tečkou a prefixu +/-

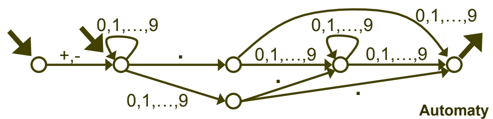


Odstranění λ-přechodů převodem na NKA

λ-uzávěr(q) = stav q a stavy, do kterých se z q dostaneme λ-přechody

nové počáteční stavy: λ-uzávěr(S)

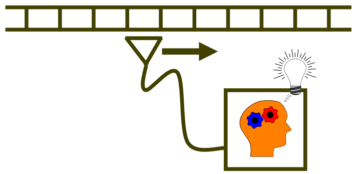
nové hrany: $\delta'(q, x) = \lambda$-uzávěr$(\delta(q, x))$



Můžeme konečné automaty ještě zobecnit?

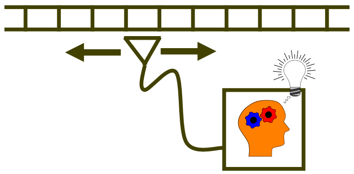
Konečný automat provádí následující činnosti:

- přečte písmeno
- změní stav vnitřní jednotky
- posune čtecí hlavu doprava



Čtecí hlava se nesmí vracet!

Co když automatu povolíme ovládání hlavy?



Pozor! Automat na pásku nic nepíše!

Dvousměrné (dvoucestné) konečné automaty

Dvousměrným (dvoucestným) konečným automatem nazýváme pětici $A = (Q,X,\delta,q_0,F)$, kde:

Q - konečná neprázdná množina stavů (*stavový prostor*)

X - konečná neprázdná množina symbolů (*vstupní abeceda*)

δ - zobrazení $Q \times X \rightarrow Q \times \{-1,0,+1\}$ (*přechodová funkce*)

přechodová funkce určuje i pohyb čtecí hlavy

$q_0 \in Q$ (*počáteční stav*)

$F \subseteq Q$ (*množina přijímacích stavů*)

Reprezentace:

stavový diagram, tabulka, stavový strom

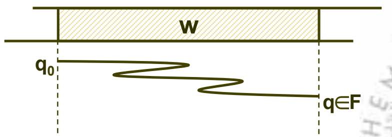
Počítání s dvousměrnými automaty

Kdy dvousměrný automat přijímá slovo?

Co se děje, je-li hlava mimo čtené slovo?

Slovo w je *přijato dvousměrným konečným automatem*, pokud:

- výpočet začal na prvním písmenu slova w vlevo v počátečním stavu
- čtecí hlava poprvé opustila slovo w vpravo v některém přijímacím stavu
- mimo čtené slovo není výpočet definován (výpočet zde končí a slovo není přijato)



Příklad dvousměrného automatu

Nejprve poznámka:

ke slovům si můžeme přidat speciální koncové znaky $\# \notin X$

je-li $L(A) = \{\#w\# \mid w \in L \subseteq X^*\}$ regulární, potom i L je regulární

$L = \partial_{\#} \partial^R_{\#} (L(A) \cap \#X^*\#)$

Příklad:

$L(B) = \{\#u\# \mid uu \in L(A)\}$ Pozor! Toto není levý ani pravý kvocient!

Nechť $A = (Q,X,\delta,q_1,F)$, definujme dvousměrný konečný automat $B = (Q \cup Q' \cup Q'' \cup \{q_0, q_N, q_F\}), X, \delta', q_0, \{q_F\})$ takto:

δ'	$x \in X$	#	poznámka
q_0	$q_N,-1$	$q_1,+1$	q_1 je počátek A
q	$p,+1$	$q',-1$	$p = \delta(q,x)$
q'	$q',-1$	$q'',+1$	
q''	$p'',+1$	$q_F,+1$	$q \in F$, $p = \delta(q,x)$
q''	$p'',+1$	$q_N,+1$	$q \notin F$, $p = \delta(q,x)$
q_N	$q_N,+1$	$q_N,+1$	
q_F	$q_N,+1$	$q_N,+1$