Automaty a gramatiky

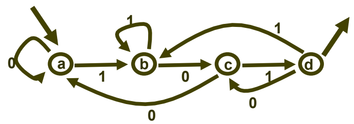
2

Roman Barták, KTIML

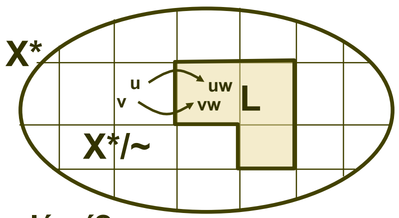
bartak@ktiml.mff.cuni.cz
http://ktiml.mff.cuni.cz/~bartak

Na zopakování

Víme, co je konečný automat
$A = (Q,X,\delta,q_0,F)$



Umíme konečné automaty charakterizovat
(Myhill-)Nerodova věta



A co dnes?

- Proč se jazyky jmenují regulární?
  - iterační (pumping) lemma
- Jak automaty zjednodušit?
  - redukce automatů

„Pravidelnost“ regulárních jazyků

Konečný automat kóduje pouze konečnou informaci.
Přesto můžeme rozpoznávat nekonečné jazyky!
Můžeme přijímat libovolně (ale konečně) dlouhá slova!

Pozorování:

$L = \{w \mid w \in \{0,1\}^* \wedge |w|_1 = 3k+1\}$

010011010 $\in$ L potom také:

- 01001101011010 $\in$ L — řetězec 01101 jsme zdvojili
- 0100110101101011010 $\in$ L — řetězec 01101 jsme ztrojili
- 0100 $\in$ L — řetězec 01101 jsme vypustili

Lze takovouto operaci provést s každým slovem libovolného regulárního jazyka?
ANO (pokud je slovo dostatečně dlouhé)

Iterační (pumping) lemma

Nechť L je regulární jazyk. Potom existuje přirozené číslo n takové, že libovolné slovo $z \in L$, $|z| \geq n$ lze psát ve tvaru uvw, kde:
$|uv| \leq n$, $1 \leq |v|$ a pro všechna $i \geq 0$ $uv^iw \in L$.

Důkaz:

za n vezměme počet stavů příslušného automatu

při zpracování slova délky ≥ n se automat nutně musí dostat do jednoho stavu dvakrát (krabičkový princip)

vezměme takový stav p, který se opakuje jako první

potom $\delta(q_0,u) = p$ poprvé jsme se dostali do p

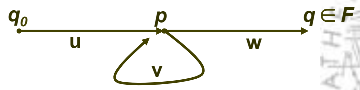
$\delta(q_0,uv) = p$ podruhé jsme se dostali do p

zřejmě $|uv| \leq n$ (pouze p se opakuje tj. maximálně n+1 stavů)

$|v| \geq 1$ (smyčka obsahuje alespoň jedno písmeno)

smyčku mezi prvním a druhým průchodem p (odpovídá jí slovo v) nyní můžeme vypustit nebo projít vícekrát

tj. $\forall i \geq 0$ $uv^iw \in L$



Použití iteračního lemmatu

Důkaz neregulárnosti jazyka

$L = \{0^i 1^i \mid i \geq 0\}$ není regulární jazyk

sporem:

- nechť L je regulární, potom lze pumpovat (∃n tž. …)
- vezměme slovo $0^n 1^n$ (to je určitě delší než n)
- pumpovat můžeme pouze v části 0^n
- potom, ale $0^{n+i}1^n \notin L$ (i > 0 je délka pumpované části), což je spor

POZOR! Iterační lemma představuje nutnou podmínku regulárnosti jazyka, nedává podmínku postačující.

Existuje jazyk, který není regulární a lze pumpovat.

$L = \{u \mid u = a^+ b^i c^i \vee u = b^i c^j\}$

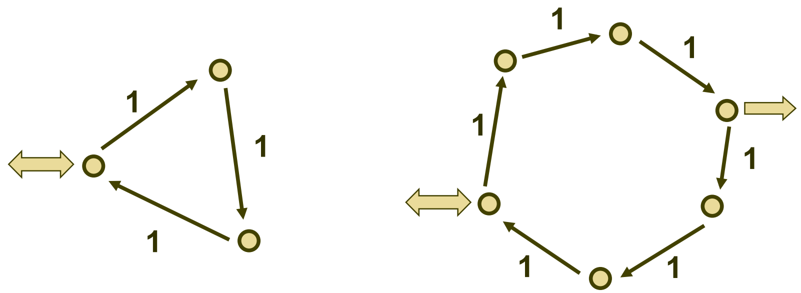
vždy lze pumpovat první písmeno

L není regulární (Nerodova věta)

Jazyk a přijímající automaty

Je automat pro daný jazyk určen jednoznačně?

NENÍ!



$L = \{w \mid w \in \{1\}^* \wedge |w| = 3k\}$

Ekvivalence automatů a homomorfismus

Jak zjistit, zda dva automaty přijímají stejný jazyk?

Říkáme, že konečné automaty A a B jsou *ekvivalentní*, jestliže rozpoznávají stejný jazyk, tj. L(A) = L(B).

Nechť A_1 a A_2 jsou konečné automaty. Řekneme, že zobrazení $h: Q_1 \to Q_2$ je (automatový) *homomorfismem*, jestliže:

1) $h(q_1) = q_2$ „stejné" počáteční stavy
2) $h(\delta_1(q,x)) = \delta_2(h(q),x)$ „stejné" přechodové funkce
3) $q \in F_1 \Leftrightarrow h(q) \in F_2$ „stejné" koncové stavy

Homomorfismus prostý a na nazýváme *isomorfismus*.

Věta o ekvivalenci automatů

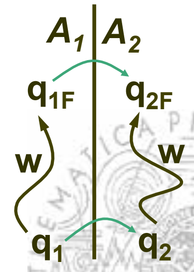
Existuje-li homomorfismus konečných automatů A_1 do A_2, pak A_1 a A_2 jsou ekvivalentní.

Důkaz:

konečnou iterací

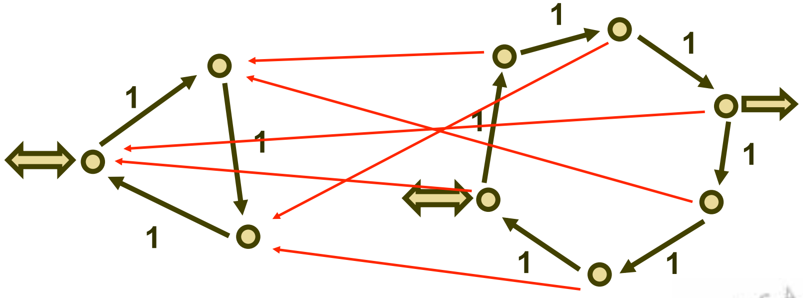
$$h(\delta_1^*(q,w)) = \delta_2^*(h(q),w) \quad w \in X^*$$

$$\begin{aligned} w \in L(A_1) &\Leftrightarrow \delta_1^*(q_1,w) \in F_1 \\ &\Leftrightarrow h(\delta_1^*(q_1,w)) \in F_2 \\ &\Leftrightarrow \delta_2^*(h(q_1),w) \in F_2 \\ &\Leftrightarrow \delta_2^*(q_2,w) \in F_2 \\ &\Leftrightarrow w \in L(A_2) \end{aligned}$$



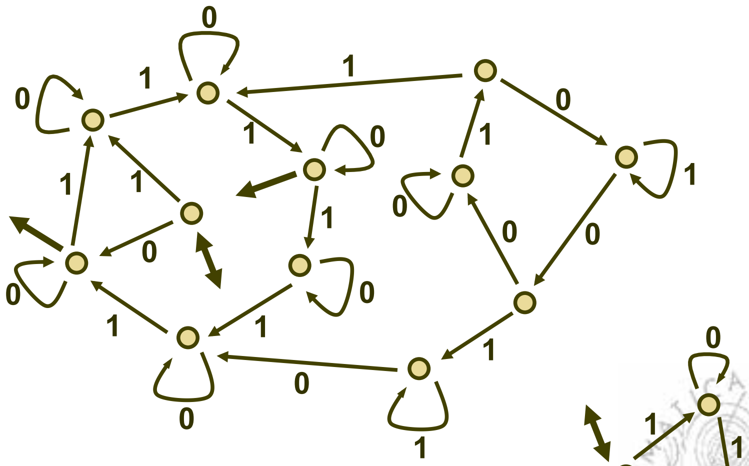
Homomorfismus automatů

$L = \{w \mid w = 1^* \wedge |w| = 3k\}$

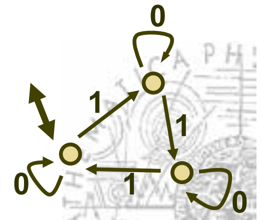


Trochu motivace

Co dělá tento automat?



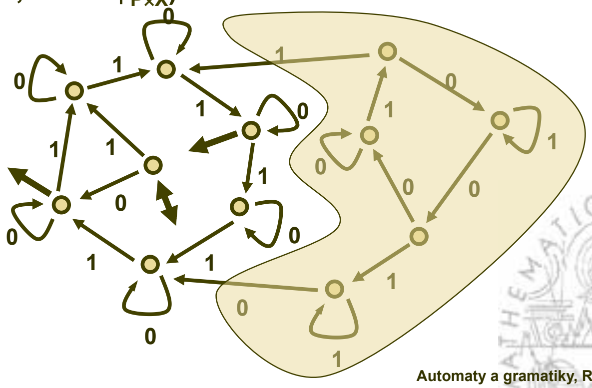
$L = \{w \mid w \in \{0,1\}^* \wedge |w|_1 = 3k\}$



Dosažitelné stavy

$A = (Q,X,\delta,q_0,F)$ je konečný automat a $q \in Q$.
Řekneme, že stav q je *dosažitelný*, jestliže $\exists w \in X^*\ \delta^*(q_0,w)=q$

Věta: Nechť P jsou dosažitelné stavy automatu A. Potom $B = (P,X,\delta',q_0,F')$ je konečný automat ekvivalentní s A ($F' = F \cap P$, $\delta' = \delta\!\uparrow_{P\times X}$).



Hledání dosažitelných stavů

Iterační algoritmus (dosažitelnost po i krocích):

$M_0 = \{q_0\}$
repeat
  $M_{i+1} = M_i \cup \{q \mid q \in Q, \exists p \in M_i\ \exists x \in X\ \delta(p,x) = q\}$
until $M_{i+1} = M_i$

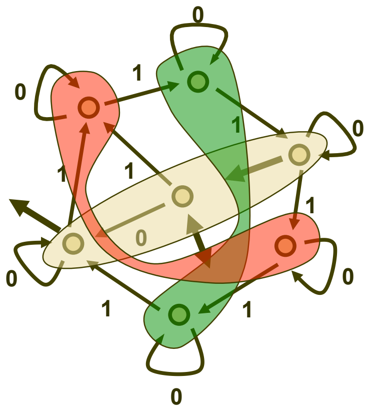
Korektnost

$M_0 \subset M_1 \subset \ldots \subset Q$ (algoritmus je konečný)
každé M_i obsahuje pouze dosažitelné stavy

Úplnost

nechť q je libovolný dosažitelný stav, tj. $\exists\, w \in X^*\ \delta^*(q_0,w) = q$
vezměme nejkratší takové $w = x_1..x_n$ tž. $\delta^*(q_0,x_1..x_n) = q$
zřejmě $\delta^*(q_0,x_1..x_i) \in M_i$ (dokonce $M_i - M_{i-1}$)
tedy $\delta^*(q_0,x_1..x_n) \in M_n$
$q \in M_n$

Ekvivalentní stavy



Výpočty startující z ekvivalentních stavů nelze rozlišit!

Nechť $A = (Q,X,\delta,q_0,F)$ je konečný automat.

Stavy p,q jsou *ekvivalentní*, značíme $p \sim q$, jestliže:
$\forall w \in X^*\ \delta^*(p,w) \in F \Leftrightarrow \delta^*(q,w) \in F$

Ekvivalence po krocích

Ekvivalence po i krocích

$p \sim^i q \qquad \forall w \in X^* \text{ tž. } |w| \leq i\ \ \delta^*(p,w) \in F \Leftrightarrow \delta^*(q,w) \in F$

$p \sim q \qquad \Leftrightarrow\ \forall i\ \ p \sim^i q$

Iterativní konstrukce $\sim^i$

$p \sim^0 q \quad \dots \quad p \in F \Leftrightarrow q \in F$

$p \sim^{i+1} q \quad \dots \quad p \sim^i q\ \wedge\ \forall\, x \in X\ \delta(p,x) \sim^i \delta(q,x)$

Je to v pořádku?

$p \sim^0 q \quad \dots \quad \delta^*(p,\lambda) = p \in F \ \Leftrightarrow\ \delta^*(q,\lambda) = q \in F$

$p \sim^{i+1} q \quad \dots \quad p \sim^i q$ tj. platí pro slova w tž. $|w| \leq i$

slova délky i+1, $w = xu$, $|u| = i$

$\delta(p,x) \sim^i \delta(q,x)$ tj. $\delta^*(\delta(p,x),u) \in F \Leftrightarrow \delta^*(\delta(q,x),u) \in F$

dohromady $\delta^*(p,xu) \in F \Leftrightarrow \delta^*(q,xu) \in F$

Vlastnosti ekvivalence po krocích

1) $\forall i \geq 0$ je $\sim^i$ ekvivalence na Q, označme $\mathcal{R}_i = Q/\sim^i$
2) $\mathcal{R}_{i+1}$ zjemňuje $\mathcal{R}_i$
3) $\mathcal{R}_{i+1} = \mathcal{R}_i \Rightarrow \forall t>0\ \mathcal{R}_{i+t} = \mathcal{R}_i$
4) nechť $|Q|=n$, potom $\exists k \leq n-1\ \mathcal{R}_{k+1} = \mathcal{R}_k$
5) $\mathcal{R}_{k+1} = \mathcal{R}_k \Rightarrow (p \sim q \Leftrightarrow p \sim^k q)$

Důkaz:

1) a 2) přímo z definice

3) $p \sim^{i+1} q \Leftrightarrow\ p \sim^i q \wedge \forall x \in X\ \delta(p,x) \sim^i \delta(q,x)$

$p \sim^{i+2} q \Leftrightarrow\ p \sim^{i+1} q \wedge \forall x \in X\ \delta(p,x) \sim^{i+1} \delta(q,x)$

4) přímo z max. počtu tříd rozkladu (n)

5) $p \sim q \quad \Leftrightarrow \forall i \geq 0\ p \sim^i q$

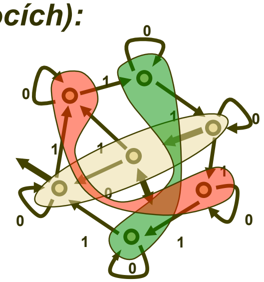
$\Leftrightarrow \forall 0 \leq i \leq k\ p \sim^i q \wedge \forall i>k\ p \sim^i q$

$\Leftrightarrow p \sim^k q$

Hledání ekvivalence stavů

Iterační algoritmus (ekvivalence po i krocích):

sestroj $\mathcal{R}_0$
repeat
  sestroj $\mathcal{R}_{i+1}$ z $\mathcal{R}_i$
until $\mathcal{R}_{i+1} = \mathcal{R}_i$

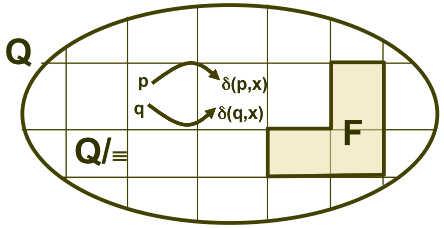


Příklad:

	0	1	$\mathcal{R}_0$	0	1	$\mathcal{R}_1$	0	1	$\mathcal{R}_2$
a	b	c	A	A	C	A	A	C	A
b	b	c	A	A	C	A	A	C	A
c	c	d	C	C	C	C	C	D	C
d	d	e	C	C	A	D	D	A	D
e	e	f	A	A	C	A	A	C	A
f	f	g	C	C	C	C	C	D	C
g	g	b	C	C	A	D	D	A	D

Automatová kongruence

Nechť ≡ je relace ekvivalence na Q. Říkáme, že ≡ je *automatovou kongruencí*, jestliže:
$\forall p,q\in Q\ p \equiv q \Rightarrow (p \in F \Leftrightarrow q \in F) \wedge \forall x \in X\ (\delta(p,x) \equiv \delta(q,x))$



Tvrzení: Ekvivalence stavů ~ je automatovou kongruencí.

Důkaz: nechť p~q, potom: $p \in F \Leftrightarrow q \in F$ (položme $w = \lambda$, $\delta^*(p,\lambda) = p$)

nechť p~q, potom: $\forall x\in X\ \forall u\in X^*\ (\delta^*(p,xu)\in F \Leftrightarrow \delta^*(q,xu) \in F)$

$\Leftrightarrow \forall x\in X\ \forall u\in X^*\ (\delta^*(\delta(p,x),u) \in F \Leftrightarrow \delta^*(\delta(q,x),u) \in F)$

$\Leftrightarrow \forall x\in X\ \delta(p,x) \sim \delta(q,x)$

Podílový automat

$A = (Q,X,\delta,q_0,F)$ je konečný automat a ≡ je automatová kongruence.
Potom $A/\equiv\ = (Q/\equiv, X, \delta_\equiv, [q_0]_\equiv, \{[q]_\equiv \mid q\in F\})$, kde $\delta_\equiv([q],x) = [\delta(q,x)]_\equiv$ je konečný automat (nazvěme ho *podílovým automatem*) ekvivalentní s A.

1) Je A/≡ skutečně konečný automat?
Množiny Q/≡ a X jsou neprázdné a konečné.
$\delta_\equiv$ je definována korektně (vlastnosti automatové kongruence)

2) Jsou oba automaty ekvivalentní?
Stačí najít homomorfismus A do A/≡ (věta o ekvivalenci automatů)!
Definujme $h: Q \to Q/\equiv$ takto $h(q) = [q]_\equiv$
$h(q_0) = [q_0]_\equiv$
$h(\delta(q,x)) = [\delta(q,x)]_\equiv = \delta_\equiv([q]_\equiv,x) = \delta_\equiv(h(q),x)$
$q \in F \Leftrightarrow h(q) = [q]_\equiv \in F_\equiv$ ($F_\equiv$ jsou koncové stavy automatu A/≡)

Podílový automat a ekvivalence stavů

A je konečný automat a ~ je ekvivalence stavů.
Potom A/~ je konečný automat ekvivalentní s A a žádné stavy A/~ nejsou ekvivalentní.

1) Ekvivalence stavů ~ je automatová kongruence, a tedy víme, že A/~ je konečný automat ekvivalentní s A.

2) V A/~ nejsou ekvivalentní stavy.
Sporem: nechť $[p]_\sim$ a $[q]_\sim$ jsou různé ekvivalentní stavy (tj. $\neg$ p~q)
vezměme libovolné $w\in X^*$:

$\delta_\sim([p]_\sim,w)\in F_\sim \Leftrightarrow \delta_\sim([q]_\sim,w)\in F_\sim$	($[p]_\sim$ a $[q]_\sim$ jsou ekvivalentní)
$\delta_\sim(h(p),w)\in F_\sim \Leftrightarrow \delta_\sim(h(q),w)\in F_\sim$	($h(p)= [p]_\sim$)
$h(\delta(p,w))\in F_\sim \Leftrightarrow h(\delta(q,w))\in F_\sim$	(h je homomorfismus)
$\delta(p,w)\in F \Leftrightarrow \delta(q,w)\in F$	
p~q	spor

Redukce automatu

Konečný automat je *redukovaný*, jestliže:
- nemá nedosažitelné stavy,
- žádné dva stavy nejsou ekvivalentní.

Konečný automat B je *reduktem* automatu A, jestliže:
- B je redukovaný,
- A a B jsou ekvivalentní.

Věta: Ke každému konečnému automatu existuje nějaký jeho redukt.

Konstruktivní důkaz *(dvě možné metody)*:
1) vyřazení nedosažitelných stavů
faktorizace podle ekvivalence stavů (nezpůsobí nedosažitelnost)
nebo
2) faktorizace podle ekvivalence stavů
vyřazení nedosažitelných stavů (nezpůsobí změnu ekvivalence)

Příklad redukce automatů

Co dělá tento automat (jaký jazyk přijímá)?

		a	b	$\mathcal{R}_0$	a	b	$\mathcal{R}_1$	a	b	$\mathcal{R}_2$
→	1	2	3	I	I	III	I	II	III	I
	2	2	4	I	I	I	II	II	II	II
←	3	3	5	III	III	III	III	III	III	III
	4	2	7	I	I	I	II	II	II	II
←	5	6	3	III	III	III	III	III	III	III
←	6	6	6	III	III	III	III	III	III	III
	7	7	4	I	I	I	II	II	II	II
	~~8~~	~~2~~	~~3~~	~~I~~	~~I~~	~~III~~	~~I~~	~~II~~	~~III~~	~~I~~
	~~9~~	~~9~~	~~4~~	~~I~~	~~I~~	~~I~~	~~II~~	~~II~~	~~II~~	~~II~~

Nedosažitelné stavy

Redukovaný automat

		a	b
→	I	II	III
	II	II	II
←	III	III	III

L = {w | w=bu, u∈{a,b}*}

Redukce automatu v kostce