Automaty a gramatiky

4

Roman Barták, KTIML

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

Co bylo minule

Nedeterministické automaty
- nedeterministický výpočet
- ekvivalence s konečným automatem (algoritmus převodu)
- zobecněný nedeterministický automat

Množinové operace s jazyky
- sjednocení, průnik, rozdíl, doplněk
- uzavřenost operací (algoritmus převodu)

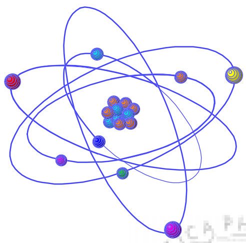
Řetězcové operace nad jazyky
- zřetězení, mocniny, pozitivní/obecná iterace, zrcadlový obraz
- kvocienty a derivace (zleva, zprava)
- uzavřenost operací (algoritmus převodu)

Substituce a homomorfismus jazyků

Trochu motivace

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

$L = L_1 \cup L_2 \cup L_3$, kde
- $L_1 = \{ w \mid w=babau,\ u \in \{a,b\}^* \}$,
- $L_2 = \{ w \mid w=uabbv,\ u,v \in \{a,b\}^* \}$
- $L_3 = \{ w \mid w=ubaa,\ u \in \{a,b\}^* \}$



Můžeme jít ještě dál!
- $L_1 = \{baba\} . \{a,b\}^*$
- $L_2 = \{a,b\}^* . \{abb\} . \{a,b\}^*$
- $L_3 = \{a,b\}^* . \{baa\}$

Pojďme ještě dál
- $L_3 = (\{a\} \cup \{b\})^* . \{b\}.\{a\}.\{a\}$

Nešlo by všechny regulární jazyky „poskládat“ z nějakých triviálních jazyků?

Regulární jazyky

***Třída regulárních jazyků* RJ(X) nad konečnou neprázdnou abecedou X je nejmenší třída jazyků, která:**

- obsahuje prázdný jazyk $\emptyset$
- pro každé písmeno $x \in X$ obsahuje jazyk $\{x\}$
- $A,B \in RJ(X) \Rightarrow A \cup B \in RJ(X)$ uzavřená na sjednocení
- $A,B \in RJ(X) \Rightarrow A.B \in RJ(X)$ uzavřená na zřetězení
- $A \in RJ(X) \Rightarrow A^* \in RJ(X)$ uzavřená na iteraci

Vlastně algebraický popis jazyků!

Speciálně:

$\{\lambda\} \in RJ(X)$ protože $\{\lambda\}' = \emptyset^*$
$X \in RJ(X)$ protože $X = \cup_{x \in X} \{x\}$ (pozor! je to konečné sjednocení)
$\{x_{i_1},\ldots,x_{i_k}\} \in RJ(X)$
$X^* \in RJ(X)$

Kleeneova věta

Libovolný jazyk je regulární právě když je rozpoznatelný konečným automatem.

Konečnými automaty lze rozpoznávat jen triviální jazyky (prázdný a jednopísmenné) a jazyky, které z nich lze složit operacemi sjednocení, zřetězení a iterace.

Důkaz $RJ \Rightarrow \mathcal{F}$

regulární jazyky jsou rozpoznatelné konečnými automaty

- triviální jazyky jsou rozpoznatelné konečným automatem
- operace sjednocení, zřetězení a iteraci dávají opět jazyk rozpoznatelný konečným automatem

Důkaz Kleeneovy věty

jazyky rozpoznatelné konečnými automaty jsou regulární

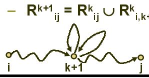
máme automat $A=(Q,X,\delta,q_1,F)$, který definuje jazyk $L(A)$
chceme ukázat, že $L(A)$ dostaneme z elementárních jazyků a operací

definujme $R_{ij} = \{w \in X^* \mid \delta^*(q_i,w)=q_j \}$ slova převádějící stav q_i na q_j
potom $L(A) = \cup_{q_i \in F} R_{1i}$ slova převádějící počáteční stav q_1 na nějaký koncový stav q_i

jsou jazyky R_{ij} regulární?
pokud ano, potom $L(A)$ je také regulární, protože $\cup$ zachovává regulárnost

definujme R^k_{ij} =slova převádějící stav q_i na q_j bez meziprůchodu stavy q_m m>k
zřejmě $R_{ij} = R^n_{ij}$ (n je počet stavů automatu)

jsou jazyky R^k_{ij} regulární?
- R^0_{ij} je regulární (žádné mezistavy, tj. jen jednopísmenná slova)
- $R^{k+1}_{ij} = R^k_{ij} \cup R^k_{i,k+1}.(R^k_{k+1,k+1})^*. R^k_{k+1,j}$ je regulární (sjednocení a iterace regulárních jazyků)



Regulární výrazy

Množina regulárních výrazů RV(X) nad konečnou neprázdnou abecedou $X=\{x_1,\dots,x_n\}$ je nejmenší množina slov v abecedě $\{x_1,\dots,x_n,\emptyset,\lambda,+,.,^*,(,)\}$, která:

- obsahuje výraz $\emptyset$ a výraz λ — $\emptyset\in RV(X)$, $\lambda\in RV(X)$
- pro každé písmeno $x\in X$ obsahuje výraz x — $x\in RV(X)$
- $\alpha,\beta\in RV(X) \Rightarrow (\alpha+\beta)\in RV(X)$
- $\alpha,\beta\in RV(X) \Rightarrow (\alpha.\beta)\in RV(X)$
- $\alpha\in RV(X) \Rightarrow \alpha^*\in RV(X)$

Příklad: ((a+((b.c)+d)*)+e)

Konvence:

- vnější závorky lze vynechat — (a+((b.c)+d)*)+e
- závorky lze vynechat u . a + díky asociativitě — a+((b.c)+d)*+e
- tečku lze vynechat — a+((bc)+d)*+e
- priorita operací (nejvyšší) *, . , + (nejnižší) — a+(bc+d)*+e

Hodnota regulárního výrazu

Hodnotou regulárního výrazu $\alpha\in RV(X)$ je množina slov $[\alpha]$ (jazyk) definovaná následovně:

- $[\emptyset]=\emptyset$, $[\lambda]=\{\lambda\}$, $[x]=\{x\}$
- $[(\alpha+\beta)]=[\alpha]\cup[\beta]$
- $[(\alpha.\beta)]=[\alpha].[\beta]$
- $[\alpha^*]=[\alpha]^*$

Regulární výrazy odpovídají regulárním jazykům

- hodnotou regulárního výrazu je regulární jazyk
- každý regulární jazyk lze reprezentovat pomocí regulárního výrazu (jazyk je hodnotou tohoto výrazu)

Příklady:

[baba(a+b)* + (a+b)*abb(a+b)* + (a+b)*baa] =

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

[(0*10*10*1)*0*] =

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

Použití regulárních výrazů

Praktický

přehledný zápis jazyka

Teoretický

zjednodušení některých důkazů

Věta: $L\in\mathcal{F}$, $\forall x\in X\ \sigma(x)\in\mathcal{F} \Rightarrow \sigma(L)\in\mathcal{F}$

L a $\sigma(x)$ jsou regulární jazyky, lze je tedy reprezentovat regulárními výrazy

každý výskyt x ve výrazu pro L stačí nahradit výrazem pro $\sigma(x)$

Rozšířené regulární výrazy

máme i další „regulární" operace, např. průnik $(\alpha \,\&\, \beta)$

Ekvivalence regulárních výrazů

$\alpha\equiv\beta$ **jestliže** $[\alpha]=[\beta]$ (tj. výrazy reprezentují stejné jazyky)

Příklad: $(0^*1)^*\equiv\lambda+(0+1)^*1$

Jak to zjistíme?

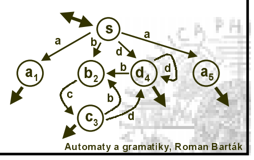
Převod regulárního výrazu na konečný automat

Metoda 1 (inkrementální):

- převeď elementární jazyky (prázdný, jednopísmenné)
- spoj použitím regulárních operací podle výrazu

Metoda 2 (přímá) — a+(bc+d)*+a

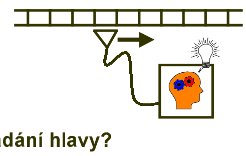
- očísluj symboly ve výrazu (zleva do doprava) — $a_1+(b_2c_3+d_4)^*+a_5$
- zjisti všechny možné páry symbolů, které se mohou vyskytovat za sebou — b_2c_3, c_3d_4, c_3b_2, d_4d_4, d_4b_2
- zjisti symboly, které mohou být první ve slově — a_1, b_2, d_4, a_5
- zjisti symboly, které mohou být poslední ve slově — a_1, c_3, d_4, a_5
- zjisti, zda jazyk obsahuje prázdné slovo — ANO
- vytvoř nedeterministický automat
  - stavy: s + očíslované symboly
  - počátek = s
  - konec = poslední symboly (+s pro λ)
  - přechody: s→první symbol
  - $x_i \to x_j$, pokud je pár x_ix_j



Můžeme konečné automaty ještě zobecnit?

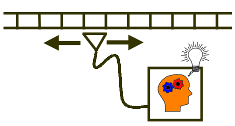
Konečný automat provádí následující činnosti:

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



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

přechodová funkce určuje i pohyb čtecí hlavy

$q_0\in Q$ (*počáteční stav*)

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

Reprezentace:

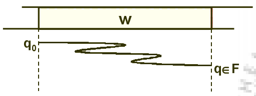
stavový diagram, tabulka, stavový strom

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 koncovém 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

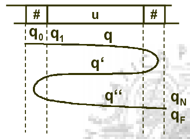
$L = \partial_\# \partial^R_\# (L(A) \cap \#X^*\#)$

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	#	poznámka
q_0	q_0,-1	q_1,+1	
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	



Věta o dvousměrných automatech

Jazyky přijímané dvousměrnými konečnými automaty jsou právě jazyky přijímané konečnými automaty.

Možnost pohybovat čtecí hlavou po pásce nezvětšila sílu konečného automatu!

Pozor, na pásku nic nepíšeme!

Pokud můžeme na pásku psát, dostaneme Turingův stroj.

***Zřejmé:* konečný automat → dvousměrný konečný automat**

dvousměrný automat vždy posouvá hlavu doprava

KA $A=(Q,X,\delta,q_0,F) \rightarrow$ 2KA $B=(Q,X,\delta',q_0,F)$, $\delta'(q,x)=(\delta(q,x),+1)$

***Zbývá:* dvousměrný konečný automat → konečný automat**

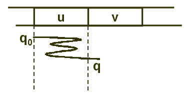
Důkaz věty o dvousměrných automatech (1)

1) Formální popis vlivu slova *u* na výpočet nad slovem *v*

(i) kdy poprvé opustíme slovo *u* vpravo (v jakém stavu poprvé vstoupíme nad *v*)

$f(q'_0) = q$ poprvé přejdeme na *v* ve stavu q

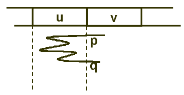
$f(q'_0) = 0$ nikdy neopustíme *u* vpravo



(ii) pokud opustíme slovo *v* vlevo, kdy se nad *v* opět vrátíme

$f(p) = q$ vrátíme se nad *v* ve stavu q

$f(p) = 0$ nikdy už se nevrátíme



2) Výpočet nad *u* máme popsaný funkcí f_u

$$f_u: Q \cup \{q'_0\} \rightarrow Q \cup \{0\}$$

$f_u(q'_0)$ popisuje situaci (i), když začneme výpočet vlevo v počátečním stavu q_0, v jakém stavu poprvé odejdeme vpravo

$f_u(p)$ ($p \in Q$) popisuje situaci (ii), když začneme výpočet vpravo v p, v jakém stavu opět odejdeme vpravo

symbol 0 značí, že daná situace nenastane (odejdeme vlevo nebo cyklus)

Důkaz věty o dvousměrných automatech (2)

Pro každé slovo *u* máme funkci f_u popisující výpočet dvousměrného automatu A nad *u*

Definujme ekvivalenci slov takto: $u \sim w \Leftrightarrow_{def} f_u=f_w$

tj. slova jsou ekvivalentní, pokud mají stejné „výpočtové" funkce

Vlastnosti ~:

- je to ekvivalence (zřejmé, definováno pomocí =)
- má konečný index (maximální počet různých funkcí je $(n+1)^{n+1}$ pro n-stavový dvousměrný automat)
- je to pravá kongruence (zřejmě $u \sim w \Rightarrow uv \sim wv$, protože rozhraní u|v a w|v je stejné a nad v se automat chová stejně)
- L(A) je sjednocením jistých tříd rozkladu $X^*/\sim$

  stačí si uvědomit, že $w \in L(A) \Leftrightarrow f_w(q'_0) \in F$

  $u \sim w \Rightarrow f_u(q'_0)=f_w(q'_0) \Rightarrow (u \in L(A) \Leftrightarrow w \in L(A))$

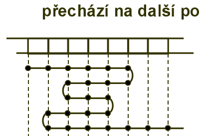
Podle Nerodovy věty je L(A) regulární jazyk.

Převod 2KA na KA

Konstruktivní důkaz věty u dvousměrných automatech.

Jak výpočet s návraty převést na lineární výpočet?

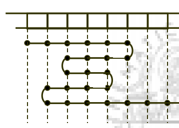
- zajímají nás jen přijímací výpočty
- díváme se na přechody mezi symboly (v jakém stavu se přechází na další políčko)



Pozorování:

- stavy se v přechodu (řezu) střídají (doprava/doleva)
- první stav jde doprava, poslední také doprava
- v deterministických přijímajících výpočtech nejsou cykly
- první a poslední řez obsahují jediný stav

1. Najdeme všechny možné řezy - posloupnosti stavů (je jich konečně mnoho).
2. Mezi řezy definujeme (nedeterministické) přechody podle čteného symbolu.
3. Rekonstruujeme výpočet skládáním řezů (jako puzzle).

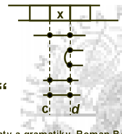


Formální převod 2KA na KA

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

Definujme ekvivalentní nedeterministický konečný automat $B=(Q',X,\delta',(q_0),F')$, kde:

- Q' = všechny korektní přechodové posloupnosti
  - posloupnosti stavů $(q^1,\dots,q^k)$ z Q takové, že
    - délka posloupnosti je lichá ($k=2m+1$)
    - žádný stav se neopakuje na liché ani na sudé pozici ($\forall i\neq j\ q^{2i}\neq q^{2j}$) & ($\forall i\neq j\ q^{2i+1}\neq q^{2j+1}$)
- $F' = \{(q) \mid q\in F\}$ přechodové posloupnosti délky 1 obsahující koncový stav
- $\delta'(c,x) = \{ d \mid d\in Q' \ \&\ c\rightarrow d$ je lokálně konzistentní přechod pro x$\}$



L(A)=L(B)?

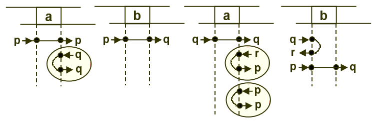
„trajektorie 2KA A odpovídá řezům KA B“

Příklad převodu 2KA na KA

Mějme následující dvousměrný konečný automat:

	a	b
→ p	p,+1	q,+1
← q	q,+1	r,-1
r	p,+1	r,-1

Možné řezy a jejich konzistentní přechody:



Ukázka výpočtu:

```
aabaabaabb
pppqqq
     r
      pqqq
         r
          pq
         rr
          pq
         rr
          ..
```

Výsledný nedeterministický KA: