Prezentácia z predmetu Formálne jazyky a automaty

Úloha č. 4/8, Dušan Hamar – 2i2

Zadanie:

A) Dokážte, že bezkontextový jazyk L nad 1-písmenkovou abecedou je regulárny

B) Zistite, či je rozhodnuteľná prázdnosť prieniku dvoch bezkontextových jazykov L1, L2, ak
- **1)** L1 je nad jednopismenovou abecedou
- **2)** L1 aj L2 su nad jednopismenovou abecedou

Riešenie:

A)

Dokážem pomocou pumpovacej lemy pre bezkontextové jazyky. Ukážem, že každé slovo viem odvodiť regulárne za pomoci konečného počtu pravidiel, ale tak že z ich opätovným použitím nevygenerujem slovo ktoré nepatrí do L.

Podľa p-q lemy existuje také p,q, že $\forall w \in L$, $|w|>p$ exist. $u,x,v,y,z \in \Sigma^*$

1. $w=uxvyz$

2. $xy<>\varepsilon$

3. $|xvy|<=q$

4. $\forall i>=0\ ux^iv y^i z \in L$

Teraz:

1. každé slovo $|w| \leq p$, môžem osobitne skontrolovať či je z jazyka L. Týchto slov je konečný počet. Maximálne p. Do gramatiky ich zakomponujem priamo z počiatočného netrminálu.(S->w | S->wX, X je neterminál)

2. uvažujem časti ktoré sa dajú pumpovať. Vždy je dĺžka takéhoto úseku $1 \leq d \leq q$. Gramatiku síce neviem zostrojiť ale

viem ukázať, že existuje. Bude to podmnožina všetkých kombinácii spojení počiatočných slov (bod 1) a "iteračných" pravidiel. ($Y \to a^d Y \mid a^d$).

$\forall w \in \{$všetky slová $|w| \leq p;\ w \in L\}$
$S \to w \mid wX$
$X \to a^d Y \mid a^d\ \forall\ d \in \{1..q\}$; Y je pre $\forall$ d iný
$Y \to a^d Y \mid a^d$

Netvrdím, že táto množina pravidiel je reg. gramatika pre L. Je to len množina regulárnych pravidiel z ktorej viem pre každé slovo vybrať pravidlá ktoré ho vygenerujú.

3. slová $|w|>p$. Tieto slová obsahujú časť ktorá sa dá napumpovať. Teda aj keď i=0. Ak dám i=0 potom vznikne slovo ktoré bude kratšie ako pôvodné w. Zapamätám si dĺžku pumpovaného úseku. Toto budem robiť až kým nedostanem slovo $|w_2| \leq p$. Toto slovo viem urobiť podľa bodu 1. ($S \to w_2X$) Ďalej viem aký úsek som postupne odstraňoval. Podľa toho si vyberiem iteračné pravidlo a použijem ho. (teda najprv dám $S \to w_2X$, potom pridám it. pravidlo podľa toho, ktorý úsek som odstraňoval $X \to a^{d1}Y$; $Y \to a^{d1}Y \mid a^{d1}$)

Teda pre každé slovo $|w|>p$ si viem vybrať nejaké regulárne pravidlá z konečného počtu pravidiel a vygenerovať ho. (t.j. každé slovo je odvoditeľné regulárne). Toto bude kúsok hľadanej gramatiky Z pumpovacej lemy platí, že keby sme tieto pravidlá používali inak (viac krát) vieme, že aj tak vygenerujeme slovo z jazyka L.

Záver: Pre každé slovo $w \in L$ existuje časť gramatiky, čiže hľadaná gramatika existuje aj keď ju neviem týmto spôsobom zostrojiť. Množina možných regulárnych pravidiel je ohraničená teda hľadaná gramatika bude určite konečná a regulárna.

B)

Prípad 2:

L1, L2 $\in L_{cf}$. Obidva sú nad jednopísmenovou abecedou

Riešenie:

Prázdnosť prieniku týchto jazykov je rozhodnuteľný problém

Dôkaz:

1. V časti a) som dokázal, že každý bezkontextový jazyk kde $|\Sigma|=1$ je vlastne regulárny jazyk. Teda máme dva regulárne jazyky L1 a L2.

2. K tým buď existuje automat, alebo gramatika. Ak je niektorý z nich zadaný regulárnou gramatikou potom pomocou štandrardnej konštrukcie (ŠK) vyrobíme ekvivalentný konečný automat.

3. Teraz vieme urobiť ŠK (lepšie povedané konštrukciou z prednášky) automat ktorý akceptuje prienik L1, L2. Ako je uvedené v prílohe

4. Problém prázdnosti jazyka pre R je rozhodnuteľný viacerými spôsobmi.

Napr:

a) urobením gramatiky k nášmu automatu a použitím ŠK na vytvorenie redukovanej gramatiky (ak nám vznikne korektne zadefinovaná gramatika resp. medzi N nám zostane počiatočný neterminál) potom je tento jazyk neprázdny a naopak.

b) prehľadávnaním grafu automatu (ak existuje cesta z počiatočného do akcpetačného stavu potom tento jazyk nie je prázdny a naopak)

Všetko čo som použil buď ŠK alebo jednoduché algoritmy. Teda sa dá zostojiť algoritmus ktorý nám vráti true/false => problém je rozhodnuteľný.

Prípad 1:

L1, L2 $\in L_{cf}$. L1 sú nad jednopísmenovou abecedou

Riešenie:

Prázdnosť prieniku týchto jazykov je rozhodnuteľný problém

Dôkaz:

Budem postupovať takmer identicky ako v predchádzajúcom prípade.

Stačí si uvedomiť čo môže byť v prieniku L1 ∩ L2. Sú to práve slová ktoré sa skladajú len z terminálu ktorý používa gramatika pre L1. Teda stačí z gramatiky pre L2 vyhodiť práve tie pravidlá kde sa používali nejaké iné terminály. (to môžme bez problémov urobiť, lebo slová odvodené pomocou tých pravidiel by v prieniku aj tak neboli). Po tomto kroku budeme ale mať vlastne dva bezkontextové jazyky nad jednopísmenovou abecedou. Čo je prípad 1.

Teda sa dá zostojiť algoritmus ktorý nám vráti true/false => problém je rozhodnuteľný.

Príloha:

Veta: R je uzavretá na prienik (konštrukcia)

Dôkaz: Nech L1, L2 $\in$R a nech A1, A2 sú konečné automaty, pre ktoré L(A1)=L1 a L(A2)=L2. Zostrojme nový automat A3, ktorý bude simulovať paralelnú prácu oboch automatov. Bude si pamätať, a akom stave by bol prvý automat a aj druhý po prečítaní písmena. Toto sa dá ľahko dosiahnúť karteziánksym súčinom množín K1, K2. A3 akceptuje dané slovo, len ak by sa automat A1 aj automat A2 dostali do akceptačných stavov. Konštrukcia A3:

$K3=K1xK2$
$\Sigma 3=\Sigma 1 \cap \Sigma 2$
$F3=F1xF2$
$\delta 3((p,q),a)=(\delta 1(p,a),\delta 2(q,a))$