

Abstrakcia nad konečnými automatmi v analýze programov

Projektová prax 1

Dávid Bolvanský

Vysoké učení technické v Brně
Fakulta informačních technologií

3. februára 2017

Obsah

Abstrakcia
nad
konečnými
automatmi v
analýze
programov

Dávid
Bolvanský

Úvod

Prienik
jazykov

Aproximácia
testu
prázdnoti

Implementácia
algoritmu

Zhrnutie

- 1 Úvod
- 2 Prienik jazykov
- 3 Aproximácia testu prázdnoti
- 4 Implementácia algoritmu
- 5 Zhrnutie

- Široké využitie automatov v informačných technológiách
- Formálna analýza programov, spracovanie textu
- Rozhodovanie o prázdnoti jazyku automatu, ktorý je výsledkom operácií nad automatmi
- Automatové algoritmy často založené na konštrukcii produktu
- Kvadratická zložitosť produktu
- Snaha o aproximácie
- Naša problematika: aproximácia testu prázdnoti prieniku jazykov rozpoznávaných automatmi

Prienik jazykov automatov

Prázdnosť prieniku jazykov rozpoznávaných automatmi A a B

$$L(A) \cap L(B) = \emptyset$$

Test prázdnosti jazyka

Test prázdnosti jazyka rozpoznávaného automatom analyzuje dostupnosť koncových stavov v produkte automatov od počiatočného stavu.

Aproximácia testu prázdnosti prieniku

Definujme $\Phi(A)$ ako aritmetickú formulu popisujúcu dĺžky slov v jazyku rozpoznávaného automatom A .

$$UNSAT(\Phi(A) \wedge \Phi(B)) \implies L(A) \cap L(B) = \emptyset$$

Abstrakcia
nad
konečnými
automatmi v
analýze
programov

Dávid
Bolvenský

Úvod

Prienik
jazykov

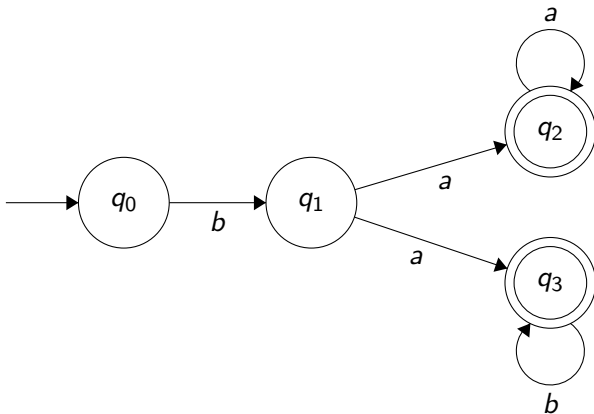
Aproximácia
testu
prázdnosti

Implementácia
algoritmu

Zhrnutie

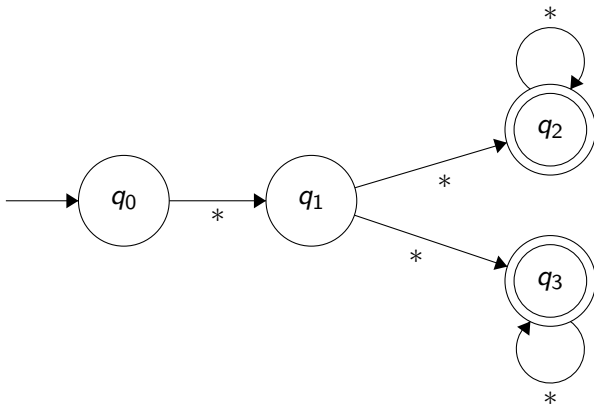
Postup aproximácie testu prázdnoty

- Na začiatku máme nederministický konečný automat A .



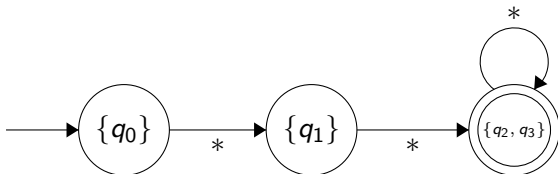
Krok 1: Prevod na jednopísmenkový NKA

- Následne ho prevedieme na jednopísmenkový nedeterministický konečný automat A' , ktorý ma iný jazyk, ale zároveň platí, že $\Phi(A) = \Phi(A')$.



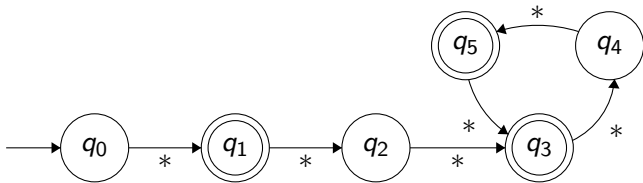
Krok 2: Determinizácia jednopísmenkového NKA

- Zdeterminizujeme jednopísmenkový NKA na jednopísmenkový deterministický konečný automat A'' .



Krok 3: Zistenie dĺžok slov

- Každý jednopísmenkový deterministický konečný automat má dve časti - rukoväť (handle) a laso (loop).



- Pre náš ukážkový automat platí, že stavy q_0, q_1, q_2, q_3 sú v rukoväti a stavy q_3, q_4, q_5 v lase.

Krok 4: Zostavenie aritmetickej formule

Aritmetická formula popisujúca dĺžky slov pre koncové stavy automatu A

$$\Phi_q(A) = \begin{cases} |d| = v & \text{ak } q \text{ v rukoväti, } v > n \\ |d| = v + k * l & \text{ak } q \text{ v lase, } v \geq n \end{cases}$$

- n je dĺžka rukoväte
- v je dĺžka od počiatočného stavu po konečný stav
- k je dĺžka lasa
- l je počet cyklení v lase

Aritmetická formula popisujúca dĺžky slov jazyku rozpoznávaného automatom A

$$\Phi(A) = \bigvee_{q \in F} \Phi_q(A)$$

Krok 5: Overenie splniteľnosti aritmetickej formule

Abstrakcia
nad
konečnými
automatmi v
analýze
programov

Dávid
Bolvanský

Úvod

Prienik
jazykov

Aproximácia
testu
prázdnoti

Implementácia
algoritmu

Zhrnutie

Súvis splniteľnosti aritmeckej formule s prázdnosťou prieniku jazykov

Zostavíme aritmetické formule $\Phi(A)$ a $\Phi(B)$ a následne overíme splniteľnosť aritmetickej formule $\Phi(A) \wedge \Phi(B)$. Táto formula môže byť:

- splniteľná - nevieme rozhodnúť o prázdnoti či neprázdnoti prieniku (\rightarrow použitie pokročilejších postupov aproximácie, Parikhove obrazy)
- nesplniteľná - prienik jazykov rozpoznávaných automatmi je prázdny

Implementácia algoritmu na výpočet dĺžok slov

Abstrakcia
nad
konečnými
automatmi v
analýze
programov

Dávid
Bolvanský

Úvod

Prienik
jazykov

Aproximácia
testu
prázdnoty

Implementácia
algoritmu

Zhrnutie

- Implementácia algoritmu v Java
- Základné operácie s automatmi pomocou knižnice BRICS
- Zvolený Alt-Ergo ako SMT solver na riešenie splniteľnosti aritmetickej formuly $\Phi(A) \wedge \Phi(B)$

- Problematika rozhodovania o prázdnoti prieniku jazykov rozpoznávanými automatmi
- Jednoduchá aproximácia pomocou dĺžok slov v jazyku automatu
- Implementácia algoritmu na výpočet dĺžok slov
- Do budúcnosti... prepojenie konštrukcie produktu s dĺžkami slov a Parikhových obrazov (Parikh images)

Abstrakcia
nad
konečnými
automatmi v
analýze
programov

Dávid
Bolvanský

Úvod

Prienik
jazykov

Aproximácia
testu
prázdnoti

Implementácia
algoritmu

Zhrnutie

Ďakujem za pozornosť.