

Zásobníkové automaty

ZPRACUJE: Mystik

Obsah

- 1 Definice zásobníkového automatu (ZA)
- 2 Varianty zásobníkového automatu
 - 2.1 Rozšířený ZA
 - 2.2 Deterministický ZA
 - 2.3 Deterministický rozšířený ZA
 - 2.4 ZA přijímající vyprázdněním zásobníku
- 3 Algoritmus vytvoření ZA pro bezkontextovou gramatiku
 - 3.1 Metoda shora-dolů
 - 3.2 Metoda zdola-nahoru

Definice zásobníkového automatu (ZA)

ZA je 7-ice $(Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$

Q - konečná **množina stavů**

Σ - konečná **vstupní abeceda**

Γ - konečná **zásobníková abeceda**

δ - **funkce přechodu** (zobrazení $Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \rightarrow 2^{(Q \times \Gamma^*)}$)

q_0 - **počáteční stav** ($q_0 \in Q$)

Z_0 - **startovací symbol zásobníku** ($Z_0 \in \Gamma$)

F - **množina koncových stavů** ($F \subset Q$)

- ϵ -přechody nečtou žádný znak ze vstupu a mohou nastat i když je vstup už prázdný

Konfigurace ZA

trojice $(q, w, \alpha) \in Q \times \Sigma^* \times \Gamma^*$ tj. stav, zbývající vstup a obsah zásobníku

Počáteční konfigurace

trojice (q_0, w, Z_0) tj. počáteční stav, celý vstup a startovací symbol zásobníku

Koncová konfigurace

trojice $(q \in F, \epsilon, \alpha)$ tj. některý koncový stav a prázdný vstup a libovolný obsah zásobníku

Přechod (\vdash)

$(q, aw, Z\alpha) \vdash (q', w, \gamma\alpha)$

změna jedné konfigurace na jinou jednou aplikací funkce přechodu (přečtení max. jednoho znaku ze vstupu, vyjmutí symbolu ze zásobníku - změna stavu a umístění nových symbolů na zásobník)

- při grafickém znázornění přechod zobrazujeme jako hranu grafu s popiskem $a / Z / \gamma$ (znak, odebráno ze zásobníku, přidáno na zásobník)

Jazyk přijímaný ZA

$L(M) = \{w \mid (q_0, w, Z_0) \vdash^* (q \in F, \epsilon, \alpha) \wedge q \in F\}$

tj. množina všech řetězců pro které ZA přejde z počáteční do koncové konfigurace

Jazyky přijímané ZA jsou jazyky třídy 2 Chomského hierarchie (bezkontextové jazyky)

Varianty zásobníkového automatu

Rozšířený ZA

δ - funkce přechodu (zobrazení $Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma^* \rightarrow 2^{(Q \times \Gamma^*)}$)

- může číst ze zásobníku celé řetězce nejen symboly
- může provádět přechody i s prázdným zásobníkem (nemusí číst symbol ze zásobníku)
- ekvivalentní ZA

Deterministický ZA

δ - funkce přechodu (zobrazení $Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \rightarrow Q \times \Gamma^*$)

- u každého přechodu pouze jedna možnost
- přijímá deterministické bezkontextové jazyky
- striktně menší vyjadřovací síla než nedeterministický ZA

Deterministický rozšířený ZA

Rozšířený zásobníkový automat je deterministický pokud platí:

- je-li $\delta(q, a, \alpha) \neq \emptyset, \delta(q, a, \beta), \alpha \neq \beta$ pak α není prefixem β ani β není prefixem α
- je-li $\delta(q, a, \alpha) \neq \emptyset, \delta(q, \epsilon, \beta), \alpha \neq \beta$ pak α není prefixem β ani β není prefixem α
(tj. musí být vždy jasné jaký přechod se použije - není možné váhat mezi dvěma řetězci zásobníkových symbolů, které oba mohou pasovat na aktuální obsah zásobníku)
- ekvivalentní deterministickému ZA

ZA přijímající vyprázdněním zásobníku

Automat přijímá řetězec přechodem do koncového stavu ($q \in F, \epsilon, \epsilon$)

- obvykle se zavádí speciální symbol označující dno zásobníku, a aby automat mohl provádět přechody i v případě, že nepotřebuje během výpočtu mít na zásobníku žádný znak
- ekvivalentní ZA

Algoritmus vytvoření ZA pro bezkontextovou gramatiku

Metoda shora-dolů

Pro gramatiku (N, Σ, P, S) zkonstruujeme ZA $(\{q\}, \Sigma, N \cup \Sigma, \delta, q, S, \emptyset)$

1) Je-li $A \rightarrow \alpha$ pravidlo z P pak $\delta(q, \epsilon, A)$ obsahuje (q, α)

2) $\delta(q, a, a) = \{(q, \epsilon)\}$

- vytváříme levou derivaci vstupního řetězce

Metoda zdola-nahoru

Pro gramatiku (N, Σ, P, S) zkonstruujeme rozšířený ZA $(\{q, r\}, \Sigma, N \cup \Sigma \cup \{\$, \}, \delta, q, \$, \{r\})$

1) $\delta(q, a, \epsilon) = \{(q, a)\}$ pro všechna $a \in \Sigma$ (tj. skládáme znaky na zásobník)

2) Je-li $A \rightarrow a$ pravidlo z P pak $\delta(q, \epsilon, a)$ obsahuje (a, A) (tj. redukujeme vrchol zásobníku dle přepisovacích pravidel)

3) $\delta(q, \epsilon, \$) = \{(r, \epsilon)\}$ (tj. po zredukování všeho na startovací nonterminál jdeme do koncového stavu)

- syntaktický analyzátor vytvářející pravou derivaci vstupu postupnými redukcemi l-fráze
- v podstatě automat postupně přesouvá prefix vstupu na zásobník, pokud je na vrcholu zásobníku řetězec odpovídající pravé straně nějakého pravidla redukujeme jej na levou stranu tohoto pravidla, kterou umístíme na zásobník, musíme skončit se startovacím nonterminálem gramatiky na zásobníku

Kategorie: Státnice 2011 | Teoretická informatika

Stránka byla naposledy editována 29. 5. 2011 v 07:08.
