Hluboké zásobníkové automaty

Luděk Dolíhal

Fakulta Informačních Technologií

29. prosince 2009

Definice

Hluboký zásobníkový automat ${}_nM$ je sedmice

$$ {}_nM = (Q, \Sigma, \Gamma, R, s, S, F), $$

kde

n - maximální hloubka v níž může dojít k rozvoji
Q - konečná množina stavů, ve kterých se může automat nacházet
Σ - abeceda vstupních symbolů
Γ - abeceda zásobníkových symbolů, kde $\Sigma \subset \Gamma, \# \in (\Gamma - \Sigma)$ je symbol označující dno zásobníku
R - konečná množina pravidel tvaru
$(mqA \rightarrow pv) \in R, A \in \Gamma, p, q \in Q, 0 < m \leq n, v \in \Gamma^*$
s $\in$ Q- počáteční stav
S $\in \Gamma$ - počáteční zásobníkový symbol
F $\subseteq$ Q- konečná množina koncových stavů

Konfigurace a přechod

Nechť ${}_nM = (Q, \Sigma, \Gamma, R, s, S, F)$, je hluboký zásobníkový automat. Konfigurací automatu ${}_nM$ je pak trojice $(q, w, \alpha) \in Q \times \Sigma^* \times (\Gamma - \{\ \#\})^*\{\#\}$, kde

1. q je přítomný stav vnitřního řízení
2. w je dosud nezpracovaná část vstupního řetězce
3. α je obsah zásobníku

Přechod hlubokého zásobníkového automatu je binární relace $\vdash_p$ definovaná na množině konfigurací:

$$(q, aw, Z\gamma) \vdash (q', w, \beta\gamma) \Longleftrightarrow (q', \beta) \in \delta(q, a, Z)$$

Přijímaný jazyk

Jazyk přijímaný koncovým stavem ${}_nM$, $L({}_nM)$ definujeme následovně:

$$ {}_nM, L({}_nM) = \{(w \in \Sigma^* \mid (s, w, S\#) \Longrightarrow^* $$
$$ (f, \varepsilon, S\#) \text{ patří do } {}_nM \text{ a } f \in F\} $$

Lze definovat i jazyk $E({}_nM)$ přijímaný prázdným zásobníkem:

$$ {}_nM, E({}_nM) = \{(w \in \Sigma^* \mid (s, w, S\#) \Longrightarrow^* $$
$$ (q, \varepsilon, \#) \text{ patří do } {}_nM \text{ a } q \in Q\} $$

Lze definovat i jazyk přijímaný prázdným zásobníkem a koncovým stavem. Všechny druhy automatů jsou ekvivalentní.
Pro každé $k \geq 1$ ${}_{deep}PD_k$ označuje množinu jazyků definovaných hlubokým zásobníkovým automatem hloubky i kde $1 \leq i \leq k$.
Analogicky i pro automaty přijímající prázdným zásobníkem.

Příklad

Mějme hluboký zásobníkový automat ${}_2M$ definovaný následovně:

$$ {}_2M = (\{s, p, q, f\}, \{a, b, c, d\}, \{a, b, c, d, A, S, \#\}, R, s, S\{f\}) $$

Množina pravidel je definována následovně:
$R = \{1sS \to qAA,\ 1qA \to paAb,\ 1qA \to fab,\ 2pA \to qcAd,\ 1fA \to fcd\}$

Automat přijme řetězec *aabbccdd* takto:

$$
\begin{aligned}
(s, aabbccdd, S\#)_e &\Longrightarrow (q, aabbccdd, AA\#) && [1sS \to qAA] \\
{}_e &\Longrightarrow (p, aabbccdd, aAbA\#) && [1qA \to paAb] \\
{}_p &\Longrightarrow (p, abbccdd, AbA\#) \\
{}_e &\Longrightarrow (q, abbccdd, AbcAd\#) && [2pA \to qcAd] \\
{}_e &\Longrightarrow (q, abbccdd, abbcAd\#) && [1qA \to fab]
\end{aligned}
$$

Příklad

Mějme hluboký zásobníkový automat ${}_2M$ definovaný následovně:

$$ {}_2M = (\{s, p, q, f\}, \{a, b, c, d\}, \{a, b, c, d, A, S, \#\}, R, s, S\{f\}) $$

Množina pravidel je definována následovně:
$R = \{1sS \to qAA,\ 1qA \to paAb,\ 1qA \to fab,\ 2pA \to qcAd,\ 1fA \to fcd\}$

Automat přijme řetězec *aabbccdd* takto:

$$
\begin{aligned}
(s, aabbccdd, S\#)_e &\Longrightarrow (q, aabbccdd, AA\#) && [1sS \to qAA] \\
{}_e &\Longrightarrow (p, aabbccdd, aAbA\#) && [1qA \to paAb] \\
{}_p &\Longrightarrow (p, abbccdd, AbA\#) \\
{}_e &\Longrightarrow (q, abbccdd, AbcAd\#) && [2pA \to qcAd] \\
{}_e &\Longrightarrow (q, abbccdd, abbcAd\#) && [1qA \to fab]
\end{aligned}
$$

Příklad

Mějme hluboký zásobníkový automat ${}_2M$ definovaný následovně:

$$ {}_2M = (\{s, p, q, f\}, \{a, b, c, d\}, \{a, b, c, d, A, S, \#\}, R, s, S\{f\}) $$

Množina pravidel je definována následovně:
$R = \{1sS \to qAA,\ 1qA \to paAb,\ 1qA \to fab,\ 2pA \to qcAd,\ 1fA \to fcd\}$

Automat přijme řetězec *aabbccdd* takto:

$$
\begin{aligned}
(s, aabbccdd, S\#)_e &\Longrightarrow (q, aabbccdd, AA\#) && [1sS \to qAA] \\
{}_e &\Longrightarrow (p, aabbccdd, aAbA\#) && [1qA \to paAb] \\
{}_p &\Longrightarrow (p, abbccdd, AbA\#) \\
{}_e &\Longrightarrow (q, abbccdd, AbcAd\#) && [2pA \to qcAd] \\
{}_e &\Longrightarrow (q, abbccdd, abbcAd\#) && [1qA \to fab]
\end{aligned}
$$

Příklad

Mějme hluboký zásobníkový automat ${}_2M$ definovaný následovně:

$$ {}_2M = (\{s, p, q, f\}, \{a, b, c, d\}, \{a, b, c, d, A, S, \#\}, R, s, S\{f\}) $$

Množina pravidel je definována následovně:
$R = \{1sS \to qAA,\ 1qA \to paAb,\ 1qA \to fab,\ 2pA \to qcAd,\ 1fA \to fcd\}$

Automat přijme řetězec *aabbccdd* takto:

$$
\begin{aligned}
(s, aabbccdd, S\#)_e &\Longrightarrow (q, aabbccdd, AA\#) && [1sS \to qAA] \\
{}_e &\Longrightarrow (p, aabbccdd, aAbA\#) && [1qA \to paAb] \\
{}_p &\Longrightarrow (p, abbccdd, AbA\#) \\
{}_e &\Longrightarrow (q, abbccdd, AbcAd\#) && [2pA \to qcAd] \\
{}_e &\Longrightarrow (q, abbccdd, abbcAd\#) && [1qA \to fab]
\end{aligned}
$$

Příklad

Mějme hluboký zásobníkový automat ${}_2M$ definovaný následovně:

$$ {}_2M = (\{s, p, q, f\}, \{a, b, c, d\}, \{a, b, c, d, A, S, \#\}, R, s, S\{f\}) $$

Množina pravidel je definována následovně:
$R = \{1sS \to qAA,\ 1qA \to paAb,\ 1qA \to fab,\ 2pA \to qcAd,\ 1fA \to fcd\}$

Automat přijme řetězec *aabbccdd* takto:

$$
\begin{aligned}
(s, aabbccdd, S\#)_e &\Longrightarrow (q, aabbccdd, AA\#) && [1sS \to qAA] \\
{}_e &\Longrightarrow (p, aabbccdd, aAbA\#) && [1qA \to paAb] \\
{}_p &\Longrightarrow (p, abbccdd, AbA\#) \\
{}_e &\Longrightarrow (q, abbccdd, AbcAd\#) && [2pA \to qcAd] \\
{}_e &\Longrightarrow (q, abbccdd, abbcAd\#) && [1qA \to fab]
\end{aligned}
$$

Příklad

Mějme hluboký zásobníkový automat ${}_2M$ definovaný následovně:

$$ {}_2M = (\{s, p, q, f\}, \{a, b, c, d\}, \{a, b, c, d, A, S, \#\}, R, s, S\{f\}) $$

Množina pravidel je definována následovně:
$R = \{1sS \to qAA,\ 1qA \to paAb,\ 1qA \to fab,\ 2pA \to qcAd,\ 1fA \to fcd\}$

Automat přijme řetězec *aabbccdd* takto:

$$
\begin{aligned}
(s, aabbccdd, S\#)_e &\Longrightarrow (q, aabbccdd, AA\#) && [1sS \to qAA] \\
{}_e &\Longrightarrow (p, aabbccdd, aAbA\#) && [1qA \to paAb] \\
{}_p &\Longrightarrow (p, abbccdd, AbA\#) \\
{}_e &\Longrightarrow (q, abbccdd, AbcAd\#) && [2pA \to qcAd] \\
{}_e &\Longrightarrow (q, abbccdd, abbcAd\#) && [1qA \to fab]
\end{aligned}
$$

Příklad pokračování

$$
\begin{aligned}
&(s, aabbccdd, S\#) \\
&\qquad\vdots \\
&_p\Longrightarrow (f, bccdd, bcAd\#) \\
&_p\Longrightarrow (f, ccdd, cAd\#) \\
&_e\Longrightarrow (f, cdd, Ad\#) \qquad [1fA \to fcd] \\
&_p\Longrightarrow (f, cdd, cdd\#) \\
&_p\Longrightarrow (f, dd, dd\#) \\
&_p\Longrightarrow (f, d, d\#) \\
&_p\Longrightarrow (f, \varepsilon, \#)
\end{aligned}
$$

Automat se z počáteční konfigurace $(s, aabbccdd, S\#)$ dostal až do konfigurace $(f, \varepsilon, \#)$.

Jazyk automatu je tedy $L({}_2M) = E({}_2M) = \{a^n b^n c^n d^n | n \geq 1\}$

$L({}_2M) \notin CF$

Příklad pokračování

$$
\begin{aligned}
&(s, aabbccdd, S\#) \\
&\qquad\vdots \\
&_p \Longrightarrow (f, bccdd, bcAd\#) \\
&_p \Longrightarrow (f, ccdd, cAd\#) \\
&_e \Longrightarrow (f, cdd, Ad\#) \qquad [1fA \rightarrow fcd] \\
&_p \Longrightarrow (f, cdd, cdd\#) \\
&_p \Longrightarrow (f, dd, dd\#) \\
&_p \Longrightarrow (f, d, d\#) \\
&_p \Longrightarrow (f, \varepsilon, \#)
\end{aligned}
$$

Automat se z počáteční konfigurace $(s, aabbccdd, S\#)$ dostal až do konfigurace $(f, \varepsilon, \#)$.

Jazyk automatu je tedy $L({}_2M) = E({}_2M) = \{a^n b^n c^n d^n | n \geq 1\}$

$L({}_2M) \notin CF$

Příklad pokračování

$$
\begin{aligned}
&(s, aabbccdd, S\#) \\
&\qquad\vdots \\
&_p \Longrightarrow (f, bccdd, bcAd\#) \\
&_p \Longrightarrow (f, ccdd, cAd\#) \\
&_e \Longrightarrow (f, cdd, Ad\#) && [1fA \rightarrow fcd] \\
&_p \Longrightarrow (f, cdd, cdd\#) \\
&_p \Longrightarrow (f, dd, dd\#) \\
&_p \Longrightarrow (f, d, d\#) \\
&_p \Longrightarrow (f, \varepsilon, \#)
\end{aligned}
$$

Automat se z počáteční konfigurace $(s, aabbccdd, S\#)$ dostal až do konfigurace $(f, \varepsilon, \#)$.

Jazyk automatu je tedy $L({}_2M) = E({}_2M) = \{a^n b^n c^n d^n | n \geq 1\}$

$L({}_2M) \notin CF$

Příklad pokračování

$$
\begin{aligned}
&(s, aabbccdd, S\#) \\
&\qquad\vdots \\
&_p \Longrightarrow (f, bccdd, bcAd\#) \\
&_p \Longrightarrow (f, ccdd, cAd\#) \\
&_e \Longrightarrow (f, cdd, Ad\#) && [1fA \rightarrow fcd] \\
&_p \Longrightarrow (f, cdd, cdd\#) \\
&_p \Longrightarrow (f, dd, dd\#) \\
&_p \Longrightarrow (f, d, d\#) \\
&_p \Longrightarrow (f, \varepsilon, \#)
\end{aligned}
$$

Automat se z počáteční konfigurace $(s, aabbccdd, S\#)$ dostal až do konfigurace $(f, \varepsilon, \#)$.

Jazyk automatu je tedy $L({}_2M) = E({}_2M) = \{a^n b^n c^n d^n | n \geq 1\}$

$L({}_2M) \notin CF$

Příklad pokračování

$$
\begin{aligned}
&(s, aabbccdd, S\#) \\
&\qquad\vdots \\
&_p \Longrightarrow (f, bccdd, bcAd\#) \\
&_p \Longrightarrow (f, ccdd, cAd\#) \\
&_e \Longrightarrow (f, cdd, Ad\#) && [1fA \rightarrow fcd] \\
&_p \Longrightarrow (f, cdd, cdd\#) \\
&_p \Longrightarrow (f, dd, dd\#) \\
&_p \Longrightarrow (f, d, d\#) \\
&_p \Longrightarrow (f, \varepsilon, \#)
\end{aligned}
$$

Automat se z počáteční konfigurace $(s, aabbccdd, S\#)$ dostal až do konfigurace $(f, \varepsilon, \#)$.

Jazyk automatu je tedy $L({}_2M) = E({}_2M) = \{a^n b^n c^n d^n | n \geq 1\}$

$L({}_2M) \notin CF$

Příklad pokračování

$$
\begin{aligned}
&(s, aabbccdd, S\#) \\
&\qquad\vdots \\
&_p \Longrightarrow (f, bccdd, bcAd\#) \\
&_p \Longrightarrow (f, ccdd, cAd\#) \\
&_e \Longrightarrow (f, cdd, Ad\#) && [1fA \to fcd] \\
&_p \Longrightarrow (f, cdd, cdd\#) \\
&_p \Longrightarrow (f, dd, dd\#) \\
&_p \Longrightarrow (f, d, d\#) \\
&_p \Longrightarrow (f, \varepsilon, \#)
\end{aligned}
$$

Automat se z počáteční konfigurace $(s, aabbccdd, S\#)$ dostal až do konfigurace $(f, \varepsilon, \#)$.

Jazyk automatu je tedy $L({}_2M) = E({}_2M) = \{a^n b^n c^n d^n | n \geq 1\}$

$L({}_2M) \notin CF$

Příklad pokračování

$$
\begin{aligned}
&(s, aabbccdd, S\#) \\
&\qquad\vdots \\
&_p \Longrightarrow (f, bccdd, bcAd\#) \\
&_p \Longrightarrow (f, ccdd, cAd\#) \\
&_e \Longrightarrow (f, cdd, Ad\#) && [1fA \to fcd] \\
&_p \Longrightarrow (f, cdd, cdd\#) \\
&_p \Longrightarrow (f, dd, dd\#) \\
&_p \Longrightarrow (f, d, d\#) \\
&_p \Longrightarrow (f, \varepsilon, \#)
\end{aligned}
$$

Automat se z počáteční konfigurace $(s, aabbccdd, S\#)$ dostal až do konfigurace $(f, \varepsilon, \#)$.

Jazyk automatu je tedy $L({}_2M) = E({}_2M) = \{a^n b^n c^n d^n | n \geq 1\}$

$L({}_2M) \notin CF$

Příklad pokračování

$$
\begin{aligned}
&(s, aabbccdd, S\#) \\
&\qquad\vdots \\
&_p \Longrightarrow (f, bccdd, bcAd\#) \\
&_p \Longrightarrow (f, ccdd, cAd\#) \\
&_e \Longrightarrow (f, cdd, Ad\#) && [1fA \to fcd] \\
&_p \Longrightarrow (f, cdd, cdd\#) \\
&_p \Longrightarrow (f, dd, dd\#) \\
&_p \Longrightarrow (f, d, d\#) \\
&_p \Longrightarrow (f, \varepsilon, \#)
\end{aligned}
$$

Automat se z počáteční konfigurace $(s, aabbccdd, S\#)$ dostal až do konfigurace $(f, \varepsilon, \#)$.

Jazyk automatu je tedy $L({}_2M) = E({}_2M) = \{a^n b^n c^n d^n | n \geq 1\}$

$L({}_2M) \notin CF$

n-omezené stavové gramatiky

Stavová gramatika G je pětice

$$G = (V, W, T, P, S)$$

kde

V - celková abeceda
W - konečná množina stavů
T - abeceda terminálních symbolů, $T \subset V$
S - počáteční symbol $S \in (V - T)$
P - konečná množina pravidel tvaru
$(qA \rightarrow pv) \in P, A \in (V - T), p, q \in W, v \in V^+$

Dále řekneme, že gramatika je n omezená, pokud existuje kladné celé číslo n takové, že $occur(xA, V - T) \leq n$.
Symbolicky tento fakt zapisujeme takto
$(q, xAy)_n \Longrightarrow (p, xvy)[(q, A) \Longrightarrow (p, v)]$

Vlastnosti

- hluboké zásobníkové automaty a n-omezené stavové gramatiky jsou ekvivalení modely
- hluboké zásobníkové automaty tvoří také nekonečnou hierarchii mezi třídou kontextových a bezkontextových jazyků
- položíme-li hloubku rovnu jedné, získáme klasický zásobníkový automat
- dvě formy determinismu
  - striktní determinismus - pouze jedno pravidlo pro každý stav, platí pro všechny hloubky
  - není známo zda je takovýto automat stejně silný jako nedeterministická varianta
  - determinismus s ohledem na hloubku - maximálně jedno přípustné pravidlo na každou hloubku a stav
  - ke každému hlubokému zásobníkovému automatu existuje ekvivalentní hluboký zásobníkový automat, který je deterministický vzhledem k hloubce expanze

Seznam Literatury

- Meduna, Alexander: Deep pushdown automata
- Pluháček, David: Syntaktická analýza založená na modifikovaných zásobníkových automatech
- Viktorin, Jiří: Aplikace hlubokých zásobníkových automatů v kompilátorech