

PARSING BASED ON PUSHDOWN TRANSDUCERS

Marek KUNC, Master Degree Programme (5)
Dept. of Information Systems, FIT, BUT
E-mail: xkuncm00@stud.fit.vutbr.cz

Supervised by: Ing. Luboš Lorenc

ABSTRACT

This paper discusses parser construction based on pushdown transducer. As a practical result of this work, there will be constructed a pushdown transducer based on definitions introduced in this paper.

1 ÚVOD

Tento dokument popisuje tvorbu syntaktického analyzátoru založeného na zásobníkovém převodníku. Celá práce vychází z formální definice zásobníkového převodníku a následného praktického využití jeho možností. Zásobníkový automat umí rozhodnout o syntaktické příslušnosti vstupního řetězce do daného jazyka. Oproti tomu zásobníkový převodník například navíc umožní vydávat dodatečné informace o procesu překladu, je schopen zpracovat vstupní řetězec a dodat ho na výstup v předem předepsané formě, či vytvořit rozbor vstupní věty.

2 PROCES PŘEKLADU

Zásobníkový automat je schopen přijímat celou třídu bezkontextových jazyků. Jeho výstup je ale omezen pouze na rozhodnutí příslušnosti vstupního řetězce do daného jazyka. To ale nemusí vždy dostačovat našim záměrům.

Převodník obecně je automat s možností zapisovat na výstupní pásku. Rozšíříme-li zásobníkový automat o možnost zapisovat na výstupní pásku, získáme zásobníkový převodník. Zásobníkový převodník startuje ze svého počátečního stavu (kdy se na vrcholu zásobníku nachází počáteční zásobníkový symbol a hlavy obou pásek, jak vstupní, tak výstupní se nachází na začátku). Převodník postupně čte symboly ze vstupní pásky, mění svůj stav a vrchol zásobníku. Souběžně zapisuje symboly na výstupní pásku. Po přečtení celé vstupní pásky a normálním ukončení činnosti převodníku (přechodem do koncového stavu, vyprázdněním zásobníku, či kombinací obou) se na výstupní pásce nalézá překlad vstupního řetězce. Jelikož zásobníkový převodník vychází ze zásobníkového automatu, jsou všechny tři způsoby ukončení činnosti převodníku ekvivalentní, co se týče přijímací síly převodníku.

3 FORMÁLNÍ DEFINICE ZÁSOBNÍKOVÉHO PŘEVODNÍKU

Zásobníkový převodník je osmice $M = (Q, \Sigma, \Gamma, \Delta, R, S, q_0, F)$, kde

- Q je konečná množina stavů,
- Σ je konečná vstupní abeceda,
- Γ je konečná zásobníková abeceda,
- Δ je konečná výstupní abeceda,
- S je počáteční symbol zásobníku,
- $q_0 \in Q$ je počáteční stav,
- $F \subseteq Q$ je množina koncových stavů,
- R je konečná množina pravidel tvaru $upa \rightarrow vqb$ kde $u \in \Gamma$, $v \in \Gamma^*$, $p, q \in Q$,
 $a \in (\Sigma \cup \{\varepsilon\})$ a $b \in \Delta^*$.

3.1 KONFIGURACE ZÁSOBNÍKOVÉHO PŘEVODNÍKU

Nechť $M = (Q, \Sigma, \Gamma, \Delta, R, S, q_0, F)$ je zásobníkový převodník a $\$$ je speciální oddělovací symbol, pro který platí $\$ \notin (\Sigma \cup \Gamma \cup \Delta \cup Q)$. Potom konfigurace M je řetězec $zq_0\$x$, kde $z \in \Gamma^*$, $q \in Q$, $a \in \Sigma^*$ a $x \in \Delta^*$.

3.2 RELACE PŘECHODU ZÁSOBNÍKOVÉHO PŘEVODNÍKU

Nechť $M = (Q, \Sigma, \Gamma, \Delta, R, S, q_0, F)$ je zásobníkový převodník a řetězce $uvpax\$z$ a $uwqxs\$zb$ jsou dvě konfigurace M , kde $v \in \Gamma$, $u, w \in \Gamma^*$, $p, q \in Q$, $a \in (\Sigma \cup \{\varepsilon\})$, $x \in \Sigma^*$, $b, z \in \Delta^*$ a $r = vpa \rightarrow wqb$ je pravidlo z R . Potom M provede přechod z konfigurace $uvpax\$z$ do konfigurace $uwqxs\$zb$ podle pravidla r , což zapíšeme $uvpax\$z \vdash uwqxs\$zb [r]$.

Zjednodušeně $uvpax\$z \vdash uwqxs\zb .

3.3 VSTUPNÍ A VÝSTUPNÍ JAZYK ZÁSOBNÍKOVÉHO PŘEVODNÍKU

Mějme zásobníkový převodník M , jeho vstupní jazyk označme jako $L_I(M)$ a výstupní jako $L_O(M)$. Vstupní jazyk M poté definujeme jako

$$L_I(M) = \{x \in \Sigma^* \mid Sq_0x\$ \vdash^* vq_F\$y; v \in \Gamma^*; q_F \in F; y \in \Delta^*\}.$$

A výstupní jazyk M nadefinujeme jako

$$L_O(M) = \{y \in \Delta^* \mid Sq_0x\$ \vdash^* vq_F\$y; v \in \Gamma^*; q_F \in F; x \in \Sigma^*\}.$$

3.4 PŘEKLAD DEFINOVANÝ ZÁSOBNÍKOVÝM PŘEVODNÍKEM

Překlad definovaný zásobníkovým převodníkem M označíme jako $\tau(M)$ a je definován $\tau(M) = \{(x, y) \mid x \in \Sigma^*; y \in \Delta^*; Sq_0x\$ \vdash^* vq_F\$y; v \in \Gamma^*; q_F \in F\}$.

4 BUDOUCÍ VÝVOJ PROJEKTU

Dnes je běžnou praxí vytvářet LALR analýzu s využitím generátorů typu bison a yacc. V mé práci se hodlám zaměřit na ruční návrh této analýzy, aby bylo názorně vidět, jak tento složitý proces překladu funguje. Během implementace bude použito několik výstupních zásobníků, přepokládám dva (jeden pro uchování pravého rozboru a druhý pro uchování údajů pro následnou vizualizaci samotného procesu překladu).

Pro navrženou gramatiku typu LALR vytvořím vlastní lexikální analyzátor (na principu konečného automatu), který na vstup zásobníkového převodníku dodá řetězec tokenů získaný ze vstupního řetězce. Výstupem zásobníkového převodníku bude rozhodnutí o příslušnosti vstupního řetězce do daného jazyka. Bude-li vstup syntakticky korektní, zásobníkový převodník navíc vypíše pravý rozbor vstupního řetězce. Ten získá během procesu syntaktické analýzy postupným ukládáním čísel použitých pravidel.

Dalším výstupem zásobníkového převodníku, v případě syntaktické korektnosti vstupu, bude názorná vizualizace procesu syntaktické analýzy metodou zdola-nahoru. Zde se zaměřím na srozumitelnost a lehkou pochopitelnost demonstrace procesu překladu.

LITERATURA

- [1] Meduna, A.: Automata and Languages, Springer, London, 2000
- [2] Meduna, A., Lorenc, L.: Self-reproducing pushdown transducers, 2004
- [3] Češka, M., Hruška, T., Beneš M.: Překladače, skripta VUT Brno FEI