# Reducing Deep Pushdown Automata

Zbyněk Křivka & Rudolf Schönecker

Supervised by Alexander Meduna

Department of Information Systems
Faculty of Information Technologies
Brno University of Technology

Electrical Engineering, Information and Communication Technologies, April 27, 2006

# Outline

# Motivation
## inspiration and special characteristics

**Inspiration**:

- A. Meduna's article about deep top-down parser
- general bottom-up context-free parser
- generalization of classical PDA

# Definition
## Reducing Deep Pushdown Automata

**RDPDA** is a 6-tuple $M = (Q, \Sigma, \Gamma, R, s, F)$

- $Q$ is a finite set of states
- $\Sigma$ is an input alphabet
- $\Gamma$ is a pushdown alphabet
- $s \in Q$ is a start state
- $R$ is finite set of rules of the form:

  $qv \vdash mpA \in R$, where $p, q \in Q$, $m \geq 1$, $A \in \Gamma - \Sigma$, $v \in \Gamma^*$
- $F \subseteq Q$ is a set of final states

**Configuration** of RDPDA

- is a pair $Q \times (\Gamma - \{\#\})^*\{\#\}$

# Definition
## Move and Accepted Language

**Move of RDPDA** from $(q, uvz)$ to $(p, uAz)$

- $q, p \in Q$, $u, v, z \in \Gamma^*$, $A \in \Gamma - \Sigma$
- by using $qv \vdash mpA$
- $occur(u, \Gamma - \Sigma) = m - 1$

is $(q, uvz) \models (p, uAz)$ $[qv \vdash mpA]$ in M

**Language accepted** by $M$, $L(M)$ :

- $L(M) = \{w \in \Sigma^* \mid (s, w\#) \models^* (f, \#)\}$

# Definition
## Reducing Deep Pushdown Automata

**Depth of RDPDA**

- every rule $qv \vdash mpA$ satisfies $m \leq n$
- $n$ is minimal positive integer

$\Rightarrow$ denoted by ${}_nM$

**Family of languages** $\mathcal{L}_k(RDPDA)$ (of depth $k$)

- all languages accepted by ${}_iRDPDA$, $1 \leq i \leq k$

# Simple example of RDPDA

## accepting of language $a^n b^n c^n$

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

1. $sab \vdash 1tA$
2. $tc \vdash 2pB$
3. $paAb \vdash 1qA$
4. $qBc \vdash 2pB$
5. $pAB\# \vdash 1f\#$

**Example (acceptance of string *aabbcc*)**

$(s, a\underline{ab}bcc\#) \models (t, aAb\underline{c}c\#)\,[1] \models (p, \underline{aAb}Bc\#)\,[2] \models$
$(q, A\underline{Bc}\#)\,[3] \models (p, \underline{AB\#})\,[4] \models (f, \#)\,[5]$

# Generative power and infinite hierarchy
## of Reducing Deep Pushdown Automata

**infinite hierarchy for RDPDAs**

$\mathcal{L}_n(RDPDA) \subset \mathcal{L}_{n+1}(RDPDA)$, for all $n \geq 1$

1. $\mathcal{L}_n(ST) \subset \mathcal{L}_{n+1}(ST)$, for all $n \geq 1$ (T. Kasai, 1970)
2. $\mathcal{L}_n(ST) = \mathcal{L}_n(RDPDA)$

**Theorem 2**

$\mathcal{L}_n(ST) = \mathcal{L}_n(RDPDA)$, for every $n \geq 1$

Proof: 1) $\mathcal{L}_n(ST) \subseteq \mathcal{L}_n(RDPDA)$ 2) $\mathcal{L}_n(RDPDA) \subseteq \mathcal{L}_n(ST)$

# Interesting determinism notice

## Reducing Deep Pushdown Automata

We give more space for nondeterminism

- analogy with top-down vs. bottom-up parsers

**notice**

*But the generative power remains the same as by Deep PDA.*

# References...

A. Meduna.
Deep Pushdown Automata.
*Acta Informatica*, 2006 (in press)

T. Kasai
A Hierarchy Between Context-Free and Context-Sensitive Languages.
*Journal of Computer and System Sciences*, vol. 4, 1970.