ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY
Volume X, Number X, XXXX, XXX–XXX

Unlimitedly Deep Pushdown Automata and Their Computational Completeness

Lucie Charvát[1] and Alexander Meduna[2]

[1,2]Brno University of Technology
Faculty of Information Technology
Božetěchova 2, 612 66 Brno, Czech Republic
[2]Centre of Excellence IT4Innovations
Email: *icharvatl@fit.vutbr.cz, meduna@fit.vutbr.cz*

Abstract. The present paper defines the notion of an unlimitedly deep pushdown automaton. In essence, this automaton expands the topmost expandable non-input symbol in its pushdown list. This expanded symbol, however, may not occur on the very top of the pushdown; instead, it may appear deeper in the pushdown. The paper demonstrates that this notion represents an automaton-based counter part to the notion of a state grammar. Indeed, both are equally powerful. Therefore, unlimitedly deep pushdown automata are computationally complete–that is, they are as powerful as Turing machines. In fact there are computationally complete with no more than four states.

1 Introduction

Consider the standard transformation that turns any context-free grammar to an equivalent pushdown automaton M that acts as a top-down parser (see [1–3]). During every move, M either pops or extends its pushdown depending on the symbol occurring on the pushdown top. If an input symbol occurs on the pushdown top, M compares the pushdown top symbol with current input symbol, and if they coincide, M pops the topmost symbol from pushdown and proceeds to the next input symbol on the input tape. If a nonterminal occurs on the pushdown top, M expands its pushdown so it replaces the top nonterminal according to an expansion rule with a string.

In this paper, we define the notion of an unlimitedly deep pushdown automaton as a slight generalization of M. The generalized version works exactly as M except that it can make expansions deeper in the pushdown. Whenever automaton is unable to find an expansion rule applicable to the topmost non-input symbol, it proceeds deeper in the pushdown to the second topmost nonterminal, and so on. In this way, M continues descending deeper into the pushdown until it either finds nonterminal to be expanded or reaches the pushdown bottom.

The paper proves that unlimitedly deep pushdown automata are equally powerful as state grammars, which generate the family of recursively enumerable languages (see [4]). Therefore, unlimitedly deep pushdown automata are computationally complete.

2 Preliminaries

We assume that the reader is familiar with formal language theory (see Harrison [5] or Meduna [6,7]). For an alphabet V, V^* represents the free monoid generated by V under the operation thus free semigroup generated by V under the operation of concatenation. For every $w \in V^*$ and $K \subseteq V^*$, **max-suffix**(w, K) denotes the longest suffix of w that is in K; analogously, **max-prefix**(w, K) denotes the longest prefix of w that is in K. Let $\mathbf{alph}(w)$ denote the set of all symbols that occur in w.

A *state grammar* is a quintuple $G = (V, W, T, P, S)$, where V is a alphabet, W is a finite set of states, $T \subseteq V$ is the alphabet of terminals, $N = V - T$, $P \subseteq (W \times N) \times (W \times (N \cup T)^*)$ is a finite set of relation and $S \in N$ is the start symbol. Instead of $(q, A, p, v) \in P$, we write $(q, A) \rightarrow (p, v) \in P$ throughout. If $(q, A) \rightarrow (p, v) \in P$ implies $v \neq \varepsilon$, then G is $\varepsilon-free$. Let $u, v \in V^*$, $(q, A) \rightarrow (p, x) \in P$, and $\mathbf{alph}(u) \cap \{B \mid (q, B) \rightarrow (o, y) \in P, o \in W, y \in V^*\} = \emptyset$. Then, $uAv \Rightarrow uxv$. In the standard manner, we extend $\Rightarrow$ to $\Rightarrow^m$, $m \geq 0$. Based on $\Rightarrow^m$, we define $\Rightarrow^+$ and $\Rightarrow^*$ as usual. The language of G, $L(G)$, is defined as $L(G) = \{w \in T^* \mid (q, S) \Rightarrow^* (p, w), q, p \in W\}$.

Families of languages generated by state grammar are denoted by $\mathcal{L}(ST)$ and $\mathcal{L}({}_{\varepsilon-free}ST)$ denote the language families generated by state grammars and $\varepsilon-free$ state grammars, respectively. $\mathcal{L}(RE)$ and $\mathcal{L}(CS)$ denote the families of recursively enumerable and context-sensitive languages, respectively.

3 Definitions

An *unlimitedly deep pushdown automaton*, *UDPDA* for short, is 7-tuple, M=(Q, T, N, R, s, S, F), where Q is a finite set of states, T is a finite alphabet of input symbols, N is a finite alphabet of non-input symbols, N contains a *bottom* symbol denoted by $\#$, $R \subseteq (Q \times (N - \#) \times Q \times ((N \cup T) - \#)^*) \cup (Q \times \# \times Q \times ((N \cup T) - \#)^* \{\#\})$ is a finite relation, $s \in Q$ is the *start state*, $S \in N$ is the *start pushdown* symbol, and $F \subseteq Q$ is a finite set of *final states*. Instead of $(q, A, p, v) \in R$, we write $qA \rightarrow pv \in R$ and call $qA \rightarrow pv$ a rule; R is the *set of rules* in M. If $qA \rightarrow pv \in R$ implies $v \neq \varepsilon$, M is $\varepsilon-free$.

A *configuration of* M is a triple in $Q \times T^* \times ((N \cup T) - \#)^* \{\#\}$. X denotes the set of all configurations of M. Let $x, y \in X$ be two configuration. M *pops* its pushdown from x to y, symbolically written as $x_p \Rightarrow y$, if $x = (q, az, au)$, $y = (q, z, u)$, where $a \in T$, $z \in T^*$, $u \in (N \cup T)^*$. M *expands* its pushdown from x to y, symbolically written as $x_e \Rightarrow y$, if $x = (q, w, uAv)$, $y = (p, w, uvz)$, $qA \rightarrow pv \in R$, $\mathbf{alph}(u) \cap \{B \mid qB \rightarrow p'z', p' \in Q, z' \in (N \cup T)^*\} = \emptyset$, where $A \in N$, $u, v, z \in (N \cup T)^*$, $q, p \in Q$. To express that M makes $x_e \Rightarrow y$ according to $qA \rightarrow pv$, we write $x_e \Rightarrow y[qA \rightarrow pv]$. M makes a *move* from x to y, symbolically written as $x \Rightarrow y$ if M either $x_e \Rightarrow y$ or $x_p \Rightarrow y$. In the standard manner, extend ${}_p\Rightarrow$, ${}_e\Rightarrow$, $\Rightarrow$ to ${}_p\Rightarrow^m$, ${}_e\Rightarrow^m$, $\Rightarrow^m$, respectively, where $m \geq 0$; then, bases on ${}_p\Rightarrow^m$, ${}_e\Rightarrow^m$ $\Rightarrow^m$, define ${}_p\Rightarrow^+$, ${}_p\Rightarrow^*$, ${}_e\Rightarrow^+$, ${}_e\Rightarrow^*$, $\Rightarrow^+$, and $\Rightarrow^*$.

We define $L(M) = \{w \in T^* \mid (s, w, S) \Rightarrow^* (f, \varepsilon, \#)\}$ in M with $f \in F\}$, ${}_fL(M) = \{w \in T^* \mid (s, w, S) \Rightarrow^* (f, \varepsilon, u\#)\}$ in M, where $f \in F$, $u \in (N \cup T)^*$

and ${}_{\varepsilon}L({}_{n}M) = \{w \in T^* \mid (s, w, S) \Rightarrow^+ (q, \varepsilon, \#)\}$, where $q \in Q$.

$\mathcal{L}(UDPDA)$ and $\mathcal{L}({}_{\varepsilon-free}UDPDA)$ denote the families accepted by $UDPDA$s and ${}_{\varepsilon-free}UDPDA$s, respectively.

4 Results

We will show that $\mathcal{L}(RE)$ = $\mathcal{L}(UDPDA)$ and $\mathcal{L}(CS)$ = $\mathcal{L}({}_{\varepsilon-free}UDPDA)$. To do so, we first prove Lemmas 1 and 2.

Lemma 1. *For every state grammar G, there exists an* $UDPDA$ M *such that* $L(G) = L(M)$.

Proof. Construction. Let

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

be a state grammar. Set $N = V - T$. Next, we construct an $UDPDA$

$$M = (Q, T, N, R, s, S, W).$$

Set $Q = W \cup \{s\}$, where $s \notin W$. The rules are constructed as follows.

1. for every $(p, S) \rightarrow (q, x) \in P$, $p, q \in W$, add $s\# \rightarrow pS\#$ to R;
2. for every $(p, A) \rightarrow (q, x) \in P$, $p, q \in W$, $A \in N$, add $pA \rightarrow qx$ to R.

To establish $L(G) = L(M)$, we prove the following following claim.

Claim 1. *Let* $(p, S) \Rightarrow^j (q, xz)$ *in G, where* $p, q \in W$, $x \in T^*$, *and* $z \in (NV^*)^*$. *Then,* $(p, xw, S\#) \Rightarrow^* (q, w, z\#)$ *in M, where* $p, q \in Q$ *and* $w \in T^*$.

Proof. This claim is proved by induction on $j \geq 0$.
Basis. Let $j = 0$, so $(p, S) \Rightarrow^0 (p, S)$ in G, where $p \in W$ and $S \in N$. Then, from 2 in the construction, we obtain

$$(p, w, S\#) \Rightarrow^0 (p, w, S\#)$$

in M, so the basis holds.
Induction Hypothesis. Assume there is $i \geq 0$ such that Claim 1 holds true for all $0 \leq j \leq i$.
Induction Step. Let $(p, S) \Rightarrow^{i+1} (q, xu\alpha v)$ in G, where $x \in T^*$, $u \in (NV^*)^*$, $\alpha, v \in V^*$ and $p, q \in W$. Since $i + 1 \geq 1$, we can express $(p, S) \Rightarrow^{i+1} (q, uxv)$ as

$$(p, S) \Rightarrow^i (h, xuAv) \Rightarrow (q, xu\alpha v)$$

$$[(h, A) \rightarrow (q, \alpha)]$$

where $A \in N$ and $h \in W$. By the induction hypothesis, we have

$$(p, xyw, S\#) \Rightarrow^* (h, yw, uAv\#)$$

where y is **max-prefix**$(u\alpha v, T^*)$. Since $(h, A) \rightarrow (q, \alpha) \in P$, according to 2 in the construction, we also have $hA \rightarrow q\alpha \in R$. Thus,

$$(h, yw, uAv\#) \Rightarrow (q, w, z\#)$$

$$[hA \rightarrow q\alpha]$$

where z is **max-suffix**$(u\alpha v, NV^*)$. Therefore, Claim 1 holds true. □

Claim 2. *Let* $(p, xw, S\#) \Rightarrow^j (q, w, z\#)$ *in* M*, where* $p, q \in Q$*,* $x, w \in T^*$ *and* $z \in (NV^*)^*$*. Then,* $(p, S) \Rightarrow^* (q, xz)$ *in G, where* $p, q \in W$*.*

Proof. This claim is proved by induction on $j \geq 0$.
Basis. Let $j = 0$, so $(p, w, S\#) \Rightarrow^0 (p, w, S\#)$ in M, where $p \in Q$ and $S \in N$. Then, from 2 in the construction, we obtain

$$(p, S) \Rightarrow^0 (p, S)$$

in G, so the basis holds.
Induction Hypothesis. Assume there is $i \geq 0$ such that Claim 2 holds true for all $0 \leq j \leq i$.
Induction Step. Let $(p, xyw, S\#) \Rightarrow^{i+1} (q, w, z\#)$ in M, where $x, y, w \in T^*$, $z \in (NV^*)^*$ and $p, q \in Q$. Since $i+1 \geq 1$, we can express $(p, xyw, S\#) \Rightarrow^{i+1} (q, w, z\#)$ as

$$(p, xyw, S\#) \Rightarrow^i (h, yw, uAv\#) \Rightarrow (q, w, z\#)$$

$$[hA \rightarrow q\alpha]$$

where $A \in N$, $\alpha \in V^*$, z is **max-suffix**$(u\alpha v, NV^*)$, y is **max-prefix**$(u\alpha v, T^*)$ and $h \in Q$. By the induction hypothesis, we have

$$(p, S) \Rightarrow^* (h, xuAv)$$

Since $hA \rightarrow q\alpha \in R$, according to 2 in construction, we also have $(h, A) \rightarrow (q, \alpha) \in P$. Thus,

$$(h, xuAv) \Rightarrow (q, xu\alpha v)$$

$$[(h, A) \rightarrow (q, \alpha)]$$

Therefore, Claim 2 holds true. □

We have shown that Claim 1 and Claim 2 hold. Thus, Lemma 1 must hold as well. □

Lemma 2. *For every UDPDA* M*, there exists a state grammar G such that* $L(M) = L(G)$*.*

Proof. Construction. Let

$$M = (Q, T, N, R, s, S, F)$$

be an *UDPDA*. Set $V = T \cup N$. Next, we construct a state grammar

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

Set $W = Q \cup \{s'\}$, where $s' \notin Q$. The rules are constructed as follows.

1. for every $sA \rightarrow qx \in R$, $q \in Q$, add $(s', S) \rightarrow (s, A)$ to P;
2. for every $pA \rightarrow qx \in R$, $p, q \in Q$, $A \in N$, add $(p, A) \rightarrow (q, x)$ to P.

To establish $L(M) = L(G)$, we prove the following following claim.

Claim 3. *Let* $(p, xw, S\#) \Rightarrow^j (q, w, z\#)$ *in* M*, where* $p, q \in Q$*,* $x, w \in T^*$ *and* $z \in (NV^*)^*$*. Then,* $(p, S) \Rightarrow^* (q, xz)$ *in G, where* $p, q \in W$*.*

Proof. This claim is proved by induction on $j \geq 0$.
Basis. Let $j = 0$, so $(p, w, S\#) \Rightarrow^0 (p, w, S\#)$ in M, where $p \in Q$ and $S \in N$. Then, from 2 in the construction, we obtain

$$(p, S) \Rightarrow^0 (p, S)$$

in G, so the basis holds.
Induction Hypothesis. Assume there is $i \geq 0$ such that Claim 2 holds true for all $0 \leq j \leq i$.
Induction Step. Let $(p, xyw, S\#) \Rightarrow^{i+1} (q, w, z\#)$ in M, where $x, y, w \in T^*$, $z \in (NV^*)^*$ and $p, q \in Q$. Since $i+1 \geq 1$, we can express $(p, xyw, S\#) \Rightarrow^{i+1} (q, w, z\#)$ as

$$(p, xyw, S\#) \Rightarrow^i (h, yw, uAv\#) \Rightarrow (q, w, z\#)$$
$$[hA \rightarrow q\alpha]$$

where $A \in N$, $\alpha \in V^*$, z is **max-suffix**$(u\alpha v, NV^*)$, y is **max-prefix**$(u\alpha v, T^*)$ and $h \in Q$. By the induction hypothesis, we have

$$(p, S) \Rightarrow^* (h, xuAv)$$

Since $hA \rightarrow q\alpha \in R$, according to 2 in construction, we also have $(h, A) \rightarrow (q, \alpha) \in P$. Thus,

$$(h, xuAv) \Rightarrow (q, xu\alpha v)$$
$$[(h, A) \rightarrow (q, \alpha)]$$

Therefore, Claim 3 holds true. □

Claim 4. *Let $(p, S) \Rightarrow^j (q, xz)$ in G, where $p, q \in W$, $x \in T^*$, and $z \in (NV^*)^*$. Then, $(p, xw, S\#) \Rightarrow^* (q, w, z\#)$ in M, where $p, q \in Q$ and $w \in T^*$.*

Proof. This claim is proved by induction on $j \geq 0$.
Basis. Let $j = 0$, so $(p, S) \Rightarrow^0 (p, S)$ in G, where $p \in W$ and $S \in N$. Then, from 2 in the construction, we obtain

$$(p, w, S\#) \Rightarrow^0 (p, w, S\#)$$

in M, so the basis holds.
Induction Hypothesis. Assume there is $i \geq 0$ such that Claim 1 holds true for all $0 \leq j \leq i$.
Induction Step. Let $(p, S) \Rightarrow^{i+1} (q, xu\alpha v)$ in G, where $x \in T^*$, $u \in (NV^*)^*$, $\alpha, v \in V^*$ and $p, q \in W$. Since $i + 1 \geq 1$, we can express $(p, S) \Rightarrow^{i+1} (q, uxv)$ as

$$(p, S) \Rightarrow^i (h, xuAv) \Rightarrow (q, xu\alpha v)$$
$$[(h, A) \rightarrow (q, \alpha)]$$

where $A \in N$ and $h \in W$. By the induction hypothesis, we have

$$(p, xyw, S\#) \Rightarrow^* (h, yw, uAv\#)$$

where y is **max-prefix**$(u\alpha v, T^*)$. Since $(h, A) \rightarrow (q, \alpha) \in P$, according to 2 in construction, we also have $hA \rightarrow q\alpha \in R$. Thus,

$$(h, yw, uAv\#) \Rightarrow (q, w, z\#)$$
$$[hA \rightarrow q\alpha]$$

where z is **max-suffix**$(u\alpha v, NV^*)$. Therefore, Claim 4 holds true. □

We have shown that Claim 3 and Claim 4 holds. Thus, Lemma 2 must hold as well. □

Theorem 1. $\mathcal{L}(ST) = \mathcal{L}(UDPDA) = \mathcal{L}(RE)$

Proof. This theorem follows from Lemma 1 and Lemma 2. □

Corollary 1. *Let $L \in \mathcal{L}(RE)$. Then there exists an UDPDA $M = (Q, T, N, R, s, S, F)$ such that $L = L(M)$ and Q has no more than four states.*

Proof. This corollary follows from Theorem 1 in this paper and Theorem 2 in [4]. □

Theorem 2. $\mathcal{L}({}_{\varepsilon-free}ST) = \mathcal{L}({}_{\varepsilon-free}UDPDA) = \mathcal{L}(CS)$

Proof. This theorem follows from Theorem 1 in this paper and Theorem 2 in [3].

Can Theorem 2 be established in terms of ${}_{\varepsilon-free}UDPDA$s with a limited number of states? □

Aknowledgement

This work was supported by The Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability (NPUII); project IT4Innovations excellence in science - LQ1602; the TAČR grant TE01020415; and the BUT grant FIT-S-14-2299.

References

[1] A. Meduna, Formal Languages and Computation: Models and Their Applications, Auerbach Publications, 2014.

[2] A. V. Aho, J. D. Ullman, The Theory of Parsing, Translation, and Compiling, Prentice Hall, 1972.

[3] T. Kasai, An hierarchy between context-free and context-sensitive languages, Journal of Computer and System Sciences 4 (5) (1970) 492–508.

[4] G. Horvat, A. Meduna, On state grammars, Acta Cybernetica 1988 (8) (1988) 237–245.

[5] M. A. Harrison, Introduction to Formal Language Theory, Addison-Wesley, 1978.

[6] A. Meduna, Automata and Languages: Theory and Applications, Springer, 2000.

[7] A. Meduna, P. Zemek, Regulated Grammars and Automata, Springer, 2014.