# A Parallel Approach to Syllabification

Anca Dinu[1] and Liviu P. Dinu[2]

[1] University of Bucharest, Faculty of Foreign Languages,
5-7 Edgar Quinet, 70106, Bucharest, Romania
anca_radulescu@yahoo.com

[2] University of Bucharest, Faculty of Mathematics and Computer Science,
14 Academiei, 70109, Bucharest, Romania
ldinu@funinf.cs.unibuc.ro

**Abstract.** In this paper we propose a parallel manner of syllabification introducing some parallel extensions of insertion grammars. We use this grammars in an application to Romanian language syllabification.

## 1 Introduction

In formal language theory, most of the generative mechanisms investigated are based on the *rewriting* operation. Several other classes of mechanisms, whose main ingredient is the *adjoining* operation, were introduced along the time. The most important of them are *the contextual grammars* (Marcus, 1969), *the tree adjoining grammars* (TAG) (Joshi et al., 1975) and *the insertion grammars* (Galiukschov, 1981), all three of them introduced with linguistic motivations. Contextual grammars were introduced by Marcus (1969) and have their origin in the attempt to bridge the gap between the structuralism and generativism. The insertion grammars (or semi-contextual grammars) are somewhat intermediate between Chomsky context-sensitive grammars (where the non-terminal are rewritten according to specified contexts) and contextual grammars (where contexts are adjoined to specified strings associated with contexts).

In this paper we introduce some parallel extensions of insertion grammars and we use them to propose a parallel manner of word syllabification. Up to now, from our knowledge, most of the formal models of syllabification were treated in a sequential manner (Vennemann (1978), Koskenniemi (1983), Bird and Ellison (1994), Kaplan and Kay (1994), Muller (2002), Dinu (2003)).

This paper is structured as follows: in Section 2 we present *the insertion grammars* and introduce two new variants of them: *parallel insertion grammars* and *maximum parallel insertion grammars*. The syllabification of words, the definition of syllable and an application (Romanian words syllabification) of this approach of syllabification is given in Section 3.

## 2 Parallel Extensions of Insertion Grammars

For elementary notions of formal language theory, such as *alphabet, concatenation, language, free monoid, lengths of words,* etc. we refer to (Păun, 1997).

A. Gelbukh (Ed.): CICLing 2005, LNCS 3406, pp. 83–87, 2005.

The basic operation in insertion grammars is the adjoining of strings, as in contextual grammars, not rewriting, as in Chomsky grammars, but the operation is controlled by a context, as in context-sensitive grammars.

**Definition 1 (Păun, 1997).** *An insertion grammar is a triple $G = (V, A, P)$, where $V$ is an alphabet, $A$ is a finite language over $V$, and $P$ is a finite set of triples of strings over $V$.*

*The elements in $A$ are called axioms and those in $P$ are called insertion rules.*

*The meaning of a triple $(u, x, v) \in P$ is: $x$ can be inserted in the context $(u, v)$. Specifically, for $w, z \in V^*$ we write $w \Rightarrow z$ if $w = w_1uvw_2,\ z = w_1uxvw_2$, for $(u, x, v) \in P$ and $w_1, w_2 \in V^*$.*

The language generated by G is defined by: $L(G) = \{z \in V^* \mid w \overset{*}{\Rightarrow} z, \text{ for } w \in A\}$.

Here we introduce two parallel extensions of insertion grammars.

**Definition 2.** *Let $G = (V, A, P)$ be an insertion grammar. We define the parallel derivation denoted $\Rightarrow_p$, by:*

$$\begin{aligned} w \Rightarrow_p z \textit{ iff } w = w_1w_2\ldots w_r, \textit{for some } r \geq 2,\ z = w_1x_1w_2x_2w_3\ldots x_{r-1}w_r \textit{ and,}\\ \textit{for all } 1 \leq i \leq r-1, \textit{ there is } (u_i, x_i, v_i) \in P \textit{ and } \alpha_i, \beta_i \in V^* \textit{ such}\\ \textit{that } w_ix_iw_{i+1} = \alpha_iu_ix_iv_i\beta_i \textit{ and } w_i = \alpha_iu_i\ ,w_{i+1} = v_i\beta_i. \end{aligned}$$

*Remark 1.* For usual derivation $\Rightarrow$ we use one selector-pair, with no restriction; in parallel derivations the whole string is decomposed into selectors.

**Definition 3.** *For an insertion grammar $G = (V, A, P)$ we define the parallel derivation with maximum use of insertions (in short, we say maximum parallel derivation), denoted $\Rightarrow_{pM}$, by:*

$$\begin{aligned} w \Rightarrow_{pM} z \textit{ iff } w = w_1w_2\ldots w_s,\ z = w_1x_1w_2x_2w_3\ldots x_{s-1}w_s,\ w \Rightarrow_p z\\ \textit{and there is no } n > s \textit{ such that } w = w_1'w_2'\ldots w_n',\\ z' = w_1'x_1'w_2'x_2'w_3'\ldots x_{n-1}'w_n',\ w \Rightarrow_p z'. \end{aligned}$$

*Remark 2.* The main difference between parallel derivation ($\Rightarrow_p$) and maximum parallel derivation ($\Rightarrow_{pM}$) with respect to an insertion grammar is that in the former we can insert any number of strings in a derivation step and in the later we insert the maximum possible number of strings in a derivation step.

For $\alpha \in \{p, pM\}$, we denote by $L_\alpha(G)$ the language generated by the grammar $G$ in the mode $\alpha$:

$$L_\alpha(G) = \{z \in V^* \mid w \overset{*}{\Rightarrow}_\alpha z, \text{ for some } w \in A\}.$$

The family of such languages is denoted by $INS_\alpha$, $\alpha \in \{p, pM\}$.

We give here (without proofs) some results regarding the relations between $INS_{pM}$ and Chomsky hierarchy.