Journal of Automata, Languages and Combinatorics 2 (1997) 3, 143–149 © Otto-von-Guericke-Universität Magdeburg

# NON-SYNCHRONIZED PATTERN LANGUAGES ARE IO-MACRO LANGUAGES

# JÜRGEN DUSKE and RAINER PARCHMANN Institut für Informatik, Universität Hannover D-30167 Hannover, Germany e-mail: {duske,parchmann}@informatik.uni-hannover.de

#### ABSTRACT

We will show that the family of non-synchronized pattern languages as defined in [4] is a proper subfamily of the inside-out macro languages [2].

Keywords: pattern systems, IO-macro languages.

### 1. Introduction

The notion of a pattern and pattern languages introduced in [1] yield to further investigations in this area in the last years, where in particular different possibilities to generate languages with patterns are intensively studied (see e.g. [3, 5, 6]). One of the questions considered is the relationship of these families with respect to other known families of languages. In this paper we will show that the family of nonsynchronized pattern languages (*NSPL*) of [4] is properly contained in the family of inside-out macro languages (*IOL*) of [2], by constructing an IO-grammar for a given pattern system. The proper inclusion then follows from the fact that the language  $\{a^n b^n c^n \mid n \ge 0\}$  is an IO-language, but not a non-synchronized pattern language. We will give a new proof of this fact which corrects a corresponding one in [4].

## 2. Basic Definitions and Notations

First we will define pattern systems and one way they can generate formal languages [4].

Let T be an alphabet of terminal symbols and  $V = \{x_1, \ldots, x_n\}$  a disjoint alphabet of variables. Let  $\alpha \in (T \cup V)^*$  and let  $\beta_1, \ldots, \beta_n \in T^*$ . Then  $\alpha|_{x_i \leftarrow \beta_i}, i \in [1:n]$ , denotes the word which is obtained from  $\alpha$  by simultaneously substituting each  $x_i$ by  $\beta_i$ .

Let  $Z_i$  for  $1 \leq i \leq n$  be nonempty subsets of  $T^*$ . Then  $\sigma(\alpha, Z_1, \ldots, Z_n) = \{w \mid w = \alpha \mid_{x_i \leftarrow \beta_i}, \beta_i \in Z_i, i \in [1:n]\}.$