Journal of Automata, Languages and Combinatorics 7 (2002) 4, 571–577

DESCRIPTIONAL COMPLEXITY OF SCATTERED REWRITING AND MULTIREWRITING: AN OVERVIEW [1]

ALEXANDER MEDUNA
Department of Information Systems, Brno University of Technology
Božetěchova 2, Brno 612 66, Czech Republic
e-mail: meduna@fit.vutbr.cz

ABSTRACT

During a derivation step, partially parallel grammars rewrite some symbols of the sentential form while leaving the others unrewritten. The present paper discusses grammars that perform two types of this parallelism – *scattered rewriting* and *multirewriting*. It gives an overview of the main results concerning their descriptional complexity with respect to the number of nonterminals or productions. In the conclusion, some open problems are pointed out.

Keywords: Scattered context grammars, multigrammars, descriptional complexity

1. Introduction

Indisputably, parallel computation fulfills a crucial role in modern computer science as a whole. Whenever investigating this computation, we face the problem of choosing its most appropriate model in order to grasp it as rigorously as possible. In formal language theory, it is more than natural to base this model upon a suitable type of grammars.

To keep the grammatical model of parallel computation simple, we surely prefer grammars based on context-free productions to those based on context-dependent productions. However, sequential grammars, such as ordinary context-free grammars, can hardly serve as a model of this kind because they rewrite only a single symbol during a derivation step. Although purely parallel grammars, such as L systems, reflect the parallel computation more appropriately, this reflection is still not quite adequate from a realistic point of view. Indeed, these parallel grammars work in a completely parallel way since they rewrite all symbols of the sentential form during a derivation step. In reality, however, parallel computation is usually performed in a partially parallel way: some parts of information are processed in parallel while the rest remains unchanged. Of course, this partially parallel computation is most appropriately formalized by partially parallel grammars, which represent a compromise

[1] Full version of an invited lecture presented at the Third International Workshop on *Descriptional Complexity of Automata, Grammars and Related Structures* (Vienna, Austria, July 20 – 22, 2001).