[5] Stollnitz, E. J., DeRose T. D., Salesin, D. H., Wavelets for Computer Graphics: A Primer. *IEEE Computer Graphics and Applications*, 1994.

Matrix approach to graph transformation

*Juan de Lara, Pedro Pablo Pérez Velasco**

Escuela Politécnica Superior, Dept. Informática, Universidad Autónoma de Madrid
[jdelara@uam.es; pedro.perez@uam.es]

2000 Mathematics Subject Classification. 68Q42

Grammar theory in general and Chomsky grammars in particular are among the basic pillars of computer science. Graph grammars are a generalization of Chomsky grammars, using graphs instead of strings for building grammar productions. There are different approaches to graph transformation, one of the most used is based on category theory [1], which includes the single and double pushout, and their generalizations to adhesive HLR categories.

We propose a new approach to graph transformation using an algebra of boolean matrices [2], where simple digraphs are represented using a matrix for edges and a vector for nodes. A production can thus be specified using matrices for edge and node deletion and addition $\left(e^N, r^N; e^E, r^E\right)$. These matrices contain a non-null element if the edge or node has to be deleted or created, respectively. Therefore, the effects of a production $p : L \to R$ can be modelled using boolean matrix operations only, $R = r \vee (\overline{e} \wedge L)$. This purely algebraic approach has several advantages over categorical approaches. It is possible to study the properties of productions with no need to associate them to a given state. Basic concepts such as sequential and parallel independence can be generalized to consider any finite sequence of productions. Besides, some new concepts such as coherence, minimal initial digraph and G-congruence have been introduced. These new concepts provide a rich amount of information about productions and how they are related to each other, including limitation in their application, dependencies and dynamical behaviour.

In addition, we introduce an operational notation for rules similar to that of functional analysis for operators. Thus, productions can be depicted as $R = \langle L, p\rangle$, splitting the static part (initial state, L) from the dynamics (element addition and deletion, p). Matchings from a rule's left hand side to a host graph enter in a natural way as morphisms and can be expressed as operators acting on productions, showing that not only productions act on host graphs modifying the system state but also the system somehow interacts with the grammar temporarily *adapting* the rules. Moreover, we have also modelled and generalized the notions of *graph constraint* and *application condition* currently used in the categorical approach [1].

In our future work, on the theoretical side, we shall extend the theory to cope with multigraphs. On the practical side, we shall develop a framework for graph transformation specification and experimentation based on this matrix approach.

[1] Ehrig, H., Engels, G., Kreowski, H.-J., Rozenberg, G. 1999. *Handbook of Graph Grammars and Computing by Graph Transformation.* (1). World Scientific.

[2] Pérez Velasco, P. P., de Lara, J. 2006. *Towards a New Algebraic Approach to Graph Transformation. Basic Concepts, Sequentialization and Parallelism: Long Version.* Tech. Rep. of the School of Comp. Sci., Univ. Autónoma Madrid. Available at:
http://www.ii.uam.es/~jlara/investigacion/techrep_03_06.pdf.