Subobject Transformation Systems and Elementary Net Systems

Andrea Corradini,[1] Frank Hermann[2] and Paweł Sobociński[3]

[1] Dipartimento di Informatica, Università di Pisa, Italy
[2] Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Germany
[3] Computer Laboratory, University of Cambridge, United Kingdom

Graph transformation systems (GTSs) [10] are a powerful specification formalism for concurrent and distributed systems, generalising another classical model of concurrency, namely Place/Transition Petri nets [8]. Along the years, the concurrent behaviour of GTSs has been deeply studied and a consolidated theory of concurrency is now available. In particular, by exploiting the relationship with Petri nets, several concurrent semantics developed for nets, like (deterministic and non-deterministic) processes and the unfolding construction, have been extended to GTSs (see, e.g., [5, 9, 2, 3]).

Recently, the algebraic Double-Pushout (DPO) Approach to graph transformation has been extended to rewriting in arbitrary *adhesive categories* [7, 6], and in cooperation with Paolo Baldan and Barbara König the authors are working on the generalization of the concurrent semantics of nets and GTS to this more abstract setting: first results concerning deterministic processes appeared in [1].

A key ingredient in the definition of processes and unfoldings for a given transformation system, is the analysis of the relationships that emerge among the occurrences of rules in the possible computations of the system. Such relations include the classical *parallel* and *sequential independence*, *causality* and *conflict*, *asymmetric conflict*, and the less known *cocausality*, *disabling* and *codisabling*, introduced in [1]. Actually, such relations are meaningful for restricted classes of transformation systems only, including *occurrence systems*, which satisfy suitable safety and acyclicity conditions.

Even if the possible ways in which the rules of an occurrence system can be related are quite well understood, a systematic study of this topic is still missing, and it is an ongoing work by the authors who introduced, to this aim, *Subobject Transformation System (STSs)* [4]. Intuitively, STSs can be understood as DPO rewriting systems in the category of subobjects of a given object of an adhesive category (corresponding to the *type graph* in the typed approches to DPO). In this framework, the usual pushout and pullback constructions are replaced by union and intersection of subobjects. In general, one works with a set theoretical syntax rather than with a categorical one.

The STS framework is exploited to identify possible basic relations among rules, using them to define other derived relations which are shown to coincide with those used in the literature, and to recast known results about parallel and sequential independence. Next a construction is presented that builds an STS from a given derivation tree of a DPO system, and it is shown that the analysis of the relationships among rule occurrences in the derivation tree can be faithfully reduced to the analyis of such relationships in the generated STS.

Known examples of STSs in the area of DPO graph transformation systems are the graph processes as defined in [5, 2], and the unfolding presented in [3], but STSs are more general, because no acyclicity constranint is enforced in their definition. More interstingly, it turns out that STSs in category **Set** with rules having empty interfaces correspond precisely to *Elementary Net System* [11], a class of Petri nets widely studied in the literature.

In this talk, we will first introduce Subobject Transformation Systems and their theory. Next we shall discuss their relationship to Elementary Net Systems and their theory. We will emphasize the methodological value of this analysis: like the theory of Place/Transition nets has been a constant source of inspiration during the last years for researchers working on both theoretical and more practical aspects of Graph Transformation Systems (as withnessed for example by the various concurrent semantics proposed for GTSs, and by their application to the verification of such systems), we expect that also the theory of ENSs will provide challegning intuitions that could be generalized, at least in part, to the more abstract setting of transformation systems over adhesive categories.

References

1. P. Baldan, A. Corradini, T. Heindel, B. König, and P. Sobociński. Processes for adhesive rewriting systems. In Luca Aceto and Anna Ingólfsdóttir, editors, *FoSSaCS*, volume 3921 of *Lecture Notes in Computer Science*, pages 202–216. Springer, 2006.
2. P. Baldan, A. Corradini, and U. Montanari. Concatenable graph processes: relating processes and derivation traces. In *Proc. of ICALP'98*, volume 1443 of *Lecture Notes in Computer Science*, pages 283–295. Springer Verlag, 1998.
3. Paolo Baldan. *Modelling concurrent computations: from contextual Petri nets to graph grammars*. PhD thesis, Department of Computer Science, University of Pisa, 2000. Available as technical report n. TD-1/00.
4. A. Corradini, F. Hermann, and P. Sobociński. Subobject transformation systems. Submitted for publication, 2006.
5. A. Corradini, U. Montanari, and F. Rossi. Graph processes. *Fundamenta Informaticae*, 26:241–265, 1996.
6. H. Ehrig, K. Ehrig, U. Prange, and G. Taentzer. *Fundamentals of Algebraic Graph Transformation*. EATCS Monographs in Theoretical Computer Science. Springer Verlag, 2006.
7. S. Lack and P. Sobociński. Adhesive and quasiadhesive categories. *Theoretical Informatics and Applications*, 39(2):511–546, 2005.
8. W. Reisig. *Petri Nets: An Introduction*. EACTS Monographs on Theoretical Computer Science. Springer Verlag, 1985.
9. L. Ribeiro. *Parallel Composition and Unfolding Semantics of Graph Grammars*. PhD thesis, Technische Universität Berlin, 1996.
10. G. Rozenberg, editor. *Handbook of Graph Grammars and Computing by Graph Transformation, Vol. 1: Foundations*. World Scientific, 1997.
11. G. Rozenberg and J. Engelfriet. Elementary Net Systems. In W. Reisig and G. Rozenberg, editors, *Lectures on Petri Nets I: Basic Models*, volume 1491 of *Lecture Notes in Computer Science*, pages 12–121. Springer Verlag, 1996.