P SYSTEMS WITH COSTS AND THEIR RELATION TO PRICED TIME AUTOMATA AND PRICED TIME PETRI NETS (ABSTRACT)

Bogdan Aman[(A)] Péter Battyányi[(B)]
Gabriel Ciobanu[(A)] György Vaszil[(B,C)]

[(A)]Romanian Academy, Institute of Computer Science, Iasi, Romania
bogdan.aman@gmail.com, gabriel@info.uaic.ro

[(B)] Department of Computer Science, Faculty of Informatics,
University of Debrecen, Kassai út 26, 4028 Debrecen, Hungary
{battyanyi.peter,vaszil.gyorgy}@inf.unideb.hu

Membrane systems of P systems is a natural computational model capturing some of the features of living cells organized in tissues and higher ordered structures [5]. The model describes a distributed, parallel, synchronious computational model, where the objects are contained in compartments which are organized in an embedded, tree-like structure. The objects in each compartment evolve in a nondeterministic, highly parallel manner: the rules enable communication of membranes and membrane dissolution as well. At a computational step in each compartment the rules applied to the objects of the compartment are chosen nondeterministically and in a maximally parallel manner, so that no new rule can be applied to remaining object in that compartment. The rules involve communication of membranes by messages: which means that certain labels direct the movement of objects – they can either ooze into the parent membrane, permeate into one of the children membranes or stay in their places. A special message conveys the dissolution of the present membrane.

Here, in addition to the usual definition of rules, we add a function defining costs for the execution of rules or for the preservation of the content of a compartment. We consider the cost only as an external/observer variable, and thus whether a rule is applicable only depends on available resources (not cost value). In our first model, the costs assigned to the rules of the different compartments are kept constant in the membrane system. Additionally, storage costs for all the elements can be given in the compartments (with multiplicities), which are also fixed for a given P system. In the basic interpretation this storage cost is zero. In this talk we

[(C)]The work of Gy. Vaszil was supported in part by grant no. MAT120558 of the National Research, Development and Innovation Office, Hungary.

present an abstract syntax of the membrane systems with costs, and then define a structural operational semantics of P systems with costs.

Next we consider the model of linearly priced time automata with costs with both transitions and locations in the spirit of the work by Abdulla and Mayr [4] and Alur et al [1]. Timed automata are tools for modelling real-time processes: a discrete transition graph is equipped with a finite set of non-negative real valued variables, the clock variables. The semantics is given by an infinite transition system: the locations are designated by vertices of the transition graph, while the edges represent the different transitions which can be of two kinds: a change of location (discrete transition) or a time consumption (time transition). An edge is annotated with a guard, an action and a reset set. A transition is enabled only if the guard fulfills by the actual valuation. The actions are taken immediately by the transition, and the reset set resets the clock variables forming the set to zero. Additionally, costs are associated to each location and transition, hence costs can be calculated both for discrete and time transition steps.

Finally, we consider priced timed Petri nets [4]. As usual, the Petri net is constituted by places and transitions with the exception that tokens now denote pairs of places and nonnegative real numbers called the age of a place. A marking is a finite multiset of tokens. We distinguish timed and discrete transition relations: a timed transition relation increases the time components of the tokens in the initial marking by a certain positive real value, while the discrete transition relation alters the initial marking itself. Moreover, a cost is calculated by a cost-function for each type of transitions.

We continue the investigations started by Aman and Ciobanu in this area ([2, 3]). Namely, we target the question of relating the P systems with costs to priced timed automata and priced timed Petri nets. In the latter two computational models with costs mainly two kinds of cost problems are considered: the Cost-Threshold Problem, where an evolution cost under a certain threshold value is intended, and the Cost-Optimality Problem, where the minimal evolution cost is computed. In accordance with this, we examine P systems with costs with respect to the Cost-Threshold Problem and the Cost-Optimality Problem. We also intend to investigate networks of P systems with costs (tissues with costs), as well as networks of priced timed automata. It is also an interesting question how different evolution strategies influence the computed cost of reaching a desired configuration.

References

[1] R. ALUR, S. L. TORRE, G. J. PAPPAS, Optimal paths in weighted timed automata. In: M. D. D. BENEDETTO, A. SANGIOVANNI-VINCENTELLI (eds.), *Hybrid Systems: Computation and Control, 4th International Workshop, HSCC 2001 Rome, Italy, March 28-30, 2001 Proceedings*. LNCS 2034, Springer, Berlin, Heidelberg, 2001, 49–62.

[2] B. AMAN, G. CIOBANU, Time Delays in Membrane Systems and Petri Nets. In: M. MASSINK, G. NORMAN (eds.), *Proceedings Ninth Workshop on Quantitative Aspects of Programming Languages, QAPL 2011, Saarbrücken, Germany, April 1–3, 2011.*. Electronic Proceedings in Theoretical Computer Science 57, 2011, 47–60.

[3] B. AMAN, G. CIOBANU, Verifying P Systems with Costs by Using Priced-Timed Maude. In: C. GRACIANI, D. ORELLANA-MARTÍN, A. RISCOS-NÚEZ, Á. ROMERO-JIMÉNEZ, L. VALENCIA-CABRERA (eds.), *Proceedings of the Fourteenth Brainstorming Week on Membrane Computing, 14th BWMC, Sevilla, Spain, February 1–5, 2016*. LNCS 2034, Springer, Berlin, Heidelberg, 2016, 85–96.

[4] R. M. MAYR, P. A. ABDULLA, Priced Timed Petri Nets. *Logical Methods in Computer Science* 9 (2013) 4, 1–51.

[5] GH. PĂUN, Computing with membranes. *Journal of Computer and System Sciences* 61 (2000) 1, 108–143.