Transforming state-based models to P Systems models in practice

Petros Kefalas[1], Ioanna Stamatopoulou[2], George Eleftherakis[1] and Marian Gheorghe[3]

[1] Department of Computer Science, CITY College, Thessaloniki, Greece
{kefalas, eleftherakis}@city.academic.gr
[2] South-East European Research Centre, Thessaloniki, Greece
istamatopoulou@seerc.org
[3] Department of Computer Science, University of Sheffield, UK
M.Gheorghe@dcs.shef.ac.uk

Abstract. We present an automatic practical transformation of Communicating X-machines to Population P Systems. The resulting compiler is able to take as input a Communicating X-machine model written in an appropriately designed language (XMDL) and produce a Population P System in another notation (PPSDL). The latter contains only transformation and communication rules. However, the user can further enhance the models with more rules that deal with the reconfiguration of structure of the network of cells. XMDL, PPSDL and their accompanied compilers and animators are briefly presented. The principles of transformations and the transformation templates of the compiler are discussed. We use an example model of a biological system, namely an ant colony, to demonstrate the usefulness of this approach.

1 Introduction

State-based methods, such as finite state machines and their counterparts, are widely used for modelling reactive systems [7]. In particular, X-machines (XMs) possess an intuitive modelling style since they reduce the number of the model's states due to their associated memory structure and they are directly linked to implementation due to transition functions between states. Most importantly, however, X-machines are coupled with techniques for formal verification and testing, reassuring correctness of implementation with respect to their models [2, 4]. There exist several tools that facilitate modelling with X-machines as well as the testing and verification of models [5, 16]. In addition, X-machine models can communicate, thus forming larger scale systems. Communicating X-machines (CXMs) provide the necessary modelling message passing means and computation that demonstrate the feasibility of scaling up models [8]. However, they do suffer from a major drawback: the organisational structure of the composed system is predefined and remains static throughout the computation. Although for some systems this is a virtue, for some others, such as multi-agent systems, reorganisation is an important feature that should be addressed in a