Theoretical Computer Science 598 (2015) 87–101

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Verification of membrane systems with delays via Petri nets with delays

Bogdan Aman, Gabriel Ciobanu *

Romanian Academy, Institute of Computer Science, Blvd. Carol I no. 8, 700505 Iaşi, Romania

ARTICLE INFO

Article history:
Received 8 August 2014
Received in revised form 8 March 2015
Accepted 28 March 2015
Available online 20 April 2015
Communicated by M.J.P. Jimenez

Keywords:
Membrane systems
Petri nets
Delays
Verification of qualitative and quantitative properties of membranes
Coloured Petri nets software tools

ABSTRACT

In this paper we verify certain properties of membrane systems through the use of specific techniques and a software tool developed for coloured Petri nets. We characterise some subclasses of membrane systems in which various qualitative properties (reachability, liveness) and quantitative properties (boundedness) are decidable. We establish a formal connection between membrane systems with delays and Petri nets with delays, and prove an operational correspondence between them. By assigning delays to rules and transitions, respectively, we make a distinction between the occurrence of the rule (transition) and its effects which become apparent after the delay. For both Petri nets and membrane systems we prove that adding finite delays does not increase their expressive power; however, both formalisms become more flexible in describing molecular phenomena where time represents a critical resource.

1. Introduction

Membrane computing is a known branch of natural computing that aims to abstract computing ideas and formal models from the structure and functioning of living cells, as well as from the organisation of cells in tissues, organs or other higher order structures such as colonies of cells. Membrane systems are parallel and distributed models working with multisets of symbols in cell-like compartmental architectures. The standard approaches in membrane computing deal with computational power, complexity and efficiency of various variants of the main model. In this paper we deal with the automated verification of certain qualitative and quantitative properties of membrane systems through the use of tools and techniques developed for Petri nets.

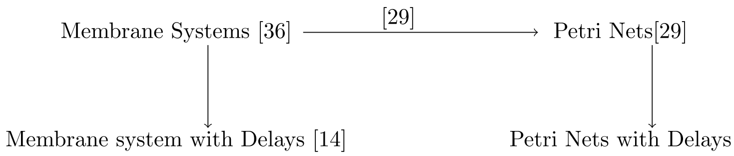
We provide first a rigorous presentation of our theoretical results: we extend both membrane systems and Petri nets by adding delays over rules and transitions, and then relate these extended formalisms and prove an operational correspondence between them. As a consequence, it is possible now to describe, analyse, and automatically verify the behavioural properties of membranes, both qualitative (reachability, liveness) and quantitative properties (boundedness).

Adding delays to a formalism can make it more adapted and adequate for modelling real systems in which timing aspects are important. Previously, membrane systems dealing with timing aspects were presented in [14]. On the other side, Petri nets have two main extensions with time: Time Petri Nets [33] in which a transition can fire within a time interval, and Timed Petri Nets [41] in which a transition fires as soon as possible. Moreover, in Petri nets time can be added to both places and transitions [43]. Here we work with a timed extension of Petri nets having delays for transitions. Despite the fact that various timed extensions exist for both membrane systems and Petri nets, we are not aware of any formal connection

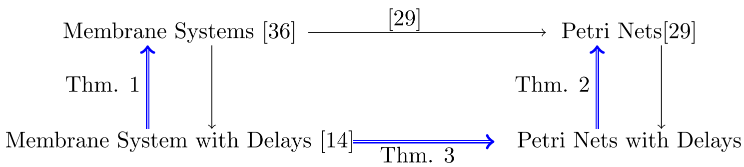
* Corresponding author.
E-mail addresses: baman@iit.tuiasi.ro (B. Aman), gabriel@info.uaic.ro (G. Ciobanu).

http://dx.doi.org/10.1016/j.tcs.2015.03.051

between these timed extensions of membrane systems and Petri nets. An attempt is presented in [39] by using a small software simulation (having some decidability aims). There exist more connections just between membrane systems and Petri nets, namely ignoring the time aspects [8,18,44]. A direct structural relationship between these membranes and Petri nets is established in [28,29] by defining a new class of Petri nets called Petri nets with localities (localities are used to model the regions of membrane systems). Petri nets with localities have been used to show how maximal evolutions taken from membrane systems are faithfully reflected in the maximally concurrent step semantics of this new class of Petri nets. In this paper we relate in a formal way the membrane systems with delays to Petri nets with delays. The already existing connections (marked by citation or easy to prove) between membrane system and Petri nets (with and without delays) are described in the following diagram.



We prove that adding time delays does not lead to more powerful formalisms. The new connections presented in this paper are described in the following diagram.



On the other hand, despite the fact that finite delays do not increase the expressive power of membrane systems and Petri nets, these new formalisms using delays are able to describe in a more elegant and natural way the systems involving timing. This paper is the extended and revised version of [4].

2. Membranes and Petri nets as multiset rewriting systems

Membrane systems (also called P systems) are introduced by Gheorghe Păun as a model of distributed, parallel and non-deterministic systems inspired by cell biology [36]. A cell is divided in various compartments, each compartment with a different task, with all of them working simultaneously to accomplish a more general task for the whole system. The membranes determine compartments where objects and evolution rules can be placed. The objects evolve according to the rules associated with each compartment, and the compartments cooperate in order to maintain the proper behaviour of the whole system. There exist a number of formal semantics which express how different membrane systems evolve [16]. Several results and variants of membrane systems (inspired by different aspects of living cells like symport and antiport communication through membranes, catalytic objects, membrane charge, etc.) are presented in [36,37]. Thus, membrane computing is essentially concerned with multiset rewriting. Various applications of membrane systems are presented in [17,23]. Links between membrane systems and process calculi are presented in [15]. An updated bibliography can be found on the membrane systems webpage http://ppage.psystems.eu.

After presenting some technical notions, we define flat membrane systems. Let $\mathbb{N}$ be the set of non-negative integers, and consider a finite alphabet V of objects. A multiset over V is a mapping $u : V \to \mathbb{N}$. The empty multiset is represented by ε. We use the string representation of multisets that is widely used in the membrane systems; when a multiset is represented by a string u, it means that every permutation of this string is allowed as a representation of the multiset. For an alphabet $V = \{a_1, \dots, a_n\}$, we denote by V^* the set of all multisets over V. An example of such a representation is $u = aabaca$, where $u(a) = 4$, $u(b) = 1$, $u(c) = 1$. Using such a representation, the operations over multisets are defined as operations over strings. Given two multisets u, v over V, for any $a \in V$, we have $(u + v)(a) = u(a) + v(a)$ as the multiset union and $(u - v)(a) = max\{0, u(a) - v(a)\}$ as the multiset difference. Also the multiset inclusion is denoted by $u \leq v$ if $u(a) \leq v(a)$, for all $a \in V$.

Definition 1. A *flat membrane system* $\Pi = (V, \mu, w, R)$ is defined by

- V is a finite *alphabet* (its elements are called *objects*);
- μ is the *membrane structure* containing only the “skin” membrane;
- w is a *finite multiset* over V; it represents the multiset of objects associated with the skin membrane;
- R is a *finite set of evolution rules* r over V associated with the skin membrane; the rules are of the form $a \to v$, where $a \in V$ and $v \in V^*$.