Time Delays in Membrane Systems and Petri Nets

Bogdan Aman
A.I.Cuza University
Blvd. Carol I, no.11, 700506 Iaşi, Romania
bogdan.aman@gmail.com

Gabriel Ciobanu
Institute of Computer Science, Romanian Academy
and "A.I. Cuza" University of Iaşi, Romania
gabriel@info.uaic.ro

Timing aspects in formalisms with explicit resources and parallelism are investigated, and it is presented a formal link between timed membrane systems and timed Petri nets with localities. For both formalisms, timing does not increase the expressive power; however both timed membrane systems and timed Petri nets are more flexible in describing molecular phenomena where time is a critical resource. We establish a link between timed membrane systems and timed Petri nets with localities, and prove an operational correspondence between them.

1 Introduction

The evolution of complex real systems frequently involves various interactions among components. Some mathematical models of such systems combine both discrete and continuous evolutions on multiple time scales with many orders of magnitude. For example, the molecular operations of a living cell can be thought of as such a dynamical system. The molecular operations happen on time scales ranging from 10^{-15} to 10^4 seconds, and proceed in ways which are dependent on populations of molecules ranging in size from as few as approximately 10 to approximately as many as 10^{20}. Molecular biologists have used formalisms developed in computer science (e.g. hybrid Petri nets) to get simplified models of some molecular phenomena like transcription and gene regulation processes. According to molecular cell biology [13]: (i) "the life span of intracellular proteins varies from as short as a few minutes for mitotic cycles, which help regulate passage through mitosis, to as long as the age of an organism for proteins in the lens of the eye", and (ii) "Most cells in multicellular organisms ... carry out a specific set of functions over periods of days to months or even the lifetime of the organism (nerve cells, for example)". Lifetimes play an important role in the biological evolution; we mention an example from the immune system.

Example 1. *According to [13], T-cell precursors arriving in the thymus from the bone marrow spend up to a week differentiating there before they enter a phase of intense proliferation. In a young adult mouse the thymus contains around* 10^8 *to* 2×10^8 *thymocytes. About* 5×10^7 *new cells are generated each day; however, only about* 10^6 *to* 2×10^6 *(roughly* $2-4\%$*) of these will leave the thymus each day as mature T cells. Despite the disparity between the numbers of T cells generated daily in the thymus and the number leaving, the thymus does not continue to grow in size or cell number. This is because approximately* 98% *of the thymocytes which develop in the thymus also die within the thymus.*

Among the formalisms able to model these systems by using explicit resources, parallelism and timing, we refer to membrane systems [15] and Petri nets [10, 16]. Membrane systems were extended with timing aspects in [4, 5]. Petri Nets have two main extensions with time: Time Petri Nets [14] (a transition can fire within a time interval) and Timed Petri Nets [19] (a transition fires as soon as possible). In Petri nets, time can be considered relative both to places and transitions [17, 20]. In this paper, we define a timed extension (relative to transitions) for Petri nets with localities, and we establish a link between timed membrane systems and timed Petri nets with localities.

M. Massink and G. Norman (Eds.): 9th Workshop on
Quantitative Aspects of Programming Languages (QAPL 2011)
EPTCS 57, 2011, pp. 47–60, doi:10.4204/EPTCS.57.4

Some connections between membrane systems and Petri nets are presented for the first time in [9, 21]. A direct structural relationship between these two formalisms is established in [11, 12] by defining a new class of Petri nets called Petri nets with localities. Localities are used to model the regions of membrane systems. This new class of Petri nets has been used to show how maximal evolutions from membrane systems are faithfully reflected in the maximally concurrent step sequence semantics of their corresponding Petri nets with localities.

Despite the fact that various timed extensions exist for both membrane systems and Petri nets, we are not aware of any connection between these timed extensions. Thus, we relate timed membrane systems with timed Petri nets with localities. The existing links (marked by citation or easy to prove) between timed membrane systems and timed Petri nets are described in the following diagram.

$$\begin{tikzcd}
\text{Membrane Systems [15]} \arrow[r, "{[12]}"] \arrow[d] & \text{Petri Nets with Localities [12]} \arrow[d] \\
\text{timed Membrane Systems [4]} & \text{timed Petri Nets with Localities}
\end{tikzcd}$$

Surprisingly, we prove that adding timing aspects does not lead to more powerful formalisms, and the new links are expressed by the following diagram.

$$\begin{tikzcd}
\text{Membrane Systems [15]} \arrow[r, "{[12]}"] \arrow[d] & \text{Petri Nets with Localities [12]} \arrow[d] \\
\text{timed Membrane Systems [4]} \arrow[u, Rightarrow, "\text{Prop. 1}"] \arrow[r, Rightarrow, "\text{Prop. 3}"'] & \text{timed Petri Nets with Localities} \arrow[u, Rightarrow, "\text{Prop. 2}"]
\end{tikzcd}$$

We prove that timing does not increase the expressive power of both membrane systems and Petri nets with localities. However the timed formalisms are able to describe more naturally some real systems involving timing. Although there are few extensions with time for both membrane systems and Petri nets, it does not exist a connection between these timed extensions. An attempt is presented in [18] by using a software simulation (and having some decidability aims). We relate timed membrane systems to timed Petri nets with localities following the research line of [12], and prove an operational correspondence between them.

2 Timed Membrane Systems

Membrane systems (also called P systems) are introduced by Păun as a model of distributed, parallel and nondeterministic systems inspired by cell biology [15]. 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 regions where objects and evolution rules can be placed. The objects evolve according to the rules associated with each region, and the regions cooperate in order to maintain the proper behaviour of the whole system. The application of evolution rules is done in parallel, and is eventually regulated by priority relationships between rules. 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 [15]. Various applications of membrane systems are presented in [7]. Links between membrane systems and process calculi are presented in [6]. An updated bibliography can be found on the membrane systems webpage http://ppage.psystems.eu.

The structure of a membrane system is represented by a tree (with the skin as its root), or equivalently, by a string of correctly matching parentheses where each pair of matching parentheses corresponds to a membrane. Graphically, a membrane structure is represented by a Venn diagram in which two sets can be either disjoint, or one is the subset of the other. A membrane without any other membrane inside is said to be elementary. The membranes are labelled in a one-to-one manner.

Let $\mathbb{N}$ be the set of positive integers, and V a finite alphabet of symbols. A multiset over V is a mapping $u : V \to \mathbb{N}$. We use the string representation of multisets that is widely accepted and used in membrane systems; a multiset w described by a^2b^5 means that a appears twice in w, while b appears five times in w. We use a global clock to simulate the passage of time. The following definition of timed membrane systems is similar to that introduced in [4], but without considering catalysts, signal-promoters and output region.

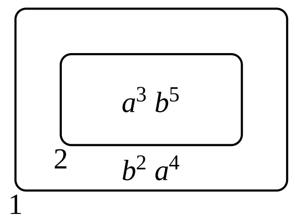
Definition 1. *A* timed membrane system $\Pi = (V, \mu, w_1, \dots, w_n, R_1, \dots, R_n, e)$ *is defined by*

- V *is an alphabet (its elements are called* objects*);*
- μ *describes the* membrane structure, *namely a structure consisting of a hierarchy of n membranes labelled from* 1 *to n which are either disjoint or included; we distinguish the external membrane, usually called "skin";*
- $w_1, \dots, w_n$ *are finite multisets over* V*;* w_i *represents the multiset of objects associated to membrane i;* $n \geq 1$ *is the initial degree of the system;*
- $R_1, \dots, R_n$ *are finite sets of evolution rules over* V *associated with the membranes of* μ*; the rules are of the form* $a \to v$*, where* $a \in V$ *and* v *is a multiset from* $\{(a, here), (a, out) \mid a \in V\} \cup \{(a, in_j) \mid a \in V, 1 \leq j \leq n\}$*;*
- $e : R_1 \cup \ldots \cup R_n \to \mathbb{N}$ *is a (computable) function indicating the execution time of each evolution rule; the time evolves according to a global clock that starts from 0 and splits time in equal intervals (units of time).*

The membrane structure and the multisets in Π determine a configuration of the system. We can pass from a configuration to another one by using the evolution rules. The use of a rule $u \to v$ in a region with a multiset w means to subtract the multiset identified by u from w, and then add the multiset represented by v. Since the right hand side v of a rule consists only of messages, an object introduced by a rule cannot evolve in the same step by means of another rule. If a message appears in v in the form $(c, here)$, then it remains in the same region. If it appears as (c, in_j), then a copy of c is introduced in the child membrane with the label j; if a child membrane with the label j does not exist, then the rule cannot be applied. If it appears as (c, out), then a copy of the object c is introduced in the parent (surrounding) membrane. The system may contain rules which are never applicable, and also rules which send objects out of the skin.

The evolution rules in a membrane are applied in a maximal parallel manner, and all membranes evolves in parallel. At each tick of the (global) clock, all the rules that can be applied must be applied in a maximal parallel manner (this means that no further rule could be applied at the same time unit). An evolution rule r started at the j-th tick of the clock ends its execution at the $j+e(r)$-th tick, meaning that the newly created objects by rule r can be used starting from the $j+e(r)+1$-th tick of the clock. When a rule starts, the objects from the left hand side of the rule become unavailable for other rules.

$$\begin{aligned} R_1 &= \{r_1 : a \to (a, in_2)\} \\ &\cup \{r_2 : b \to (a, in_2)\} \\ R_2 &= \{r_3 : a \to (b, out)(a, here)\} \\ &\cup \{r_4 : b \to (b, out)\} \end{aligned}$$



As an example, we consider a membrane system with two nested membranes (the inner membrane labelled by 2, the outer membrane labelled by 1), two sets R_1 and R_2 of evolution rules having the execution times $e(r_1) = 2$, $e(r_2) = 5$, $e(r_3) = 3$, $e(r_4) = 1$, a global clock and two symbols (a and b). Initially, membrane 1 contains the multiset $b^2\, a^4$, and membrane 2 contains the multiset $a^3\, b^5$.

Figure 1: A Timed Membrane System

In what follows we define the configurations of a membrane system, and the transition system given by considering each of the transition steps defined by maximally parallel rewriting and parallel communication, as in [8]. Let V be a finite alphabet of objects over which we consider the free commutative monoid V^* whose elements are multisets (the empty multiset is denoted by ε). Objects together with a target indication are enclosed in messages of form $(w, here)$, (w, out), and (w, in_l). For the sake of simplicity, hereinafter we consider that the messages with the same target indication merge into one message:

$$\prod_{i\in I}(v_i, here) = (w, here),\ \prod_{i\in I}(v_i, in_l) = (w, in_l),\ \prod_{i\in I}(v_i, out) = (w, out),$$

with $w = \prod_{i\in I} v_i$, I a non-empty set, and $(v_i)_{i\in I}$ a family of multisets over V.

A configuration for a membrane system is a tuple $C = (w_1, \ldots, w_n, k)$, namely the multisets of all regions together with the value of the global clock. An intermediary configuration is a tuple in which the objects have associated target indications. Each membrane system has an initial configuration which is characterized by the initial multiset of objects for each membrane of the initial membrane structure of the system. For two configurations C and C' of Π, we say that there is a transition from C to C', and write $C \Rightarrow C'$, if the following *steps* are executed in the given order:

1. *maximal parallel rewriting step* ($\stackrel{mpr}{\Longrightarrow}$): each membrane evolves in a maximal parallel manner;
2. *parallel communication of objects through membranes* ($\stackrel{tar}{\Longrightarrow}$), by sending and receiving messages.

The last step takes place only if there are messages resulting from the first step. If the first step is not possible, then neither is the second step, and we say that the system has reached a *halting configuration*. According to [3], a transition step between two configurations C, C' is given by: $C \Rightarrow C'$ iff C and C' are related by the following relation: $C \stackrel{mpr}{\Longrightarrow}\stackrel{tar}{\Longrightarrow} C'$. Starting from a configuration without messages, we apply the "mpr" step and get an intermediate configuration; if we have messages, then we apply the "tar" step. If the last configuration has no messages, then we say that the transition relation $\Rightarrow$ is well-defined as an evolution step between the first and last configurations.

The evolution of the system Π at time step k, from a configuration $C = (w_1, \ldots, w_n, k)$ to another configuration $C' = (w'_1, \ldots, w'_n, k+1)$ is made by applying a multiset of rules R in a maximally parallel manner. If the multiset R of rules is empty, then only the clock is incremented (from k to $k+1$). Given a multiset of rules R, we denote by $lhs_i = \sum_{r\in R} R(r)\cdot lhs_i^r$ the multiset of objects in the left hand sides of the rules in R which are associated to membrane i. In a similar way, by $rhs_{i,j}^k = \sum_{r\in R;\, e(r)=j} R(r)\cdot\ rhs_{i,j}^{r,k}$ is denoted the multiset of objects in the right hand sides of the rules in R applied at time k which is associated to membrane i after j units of time. We also denote by $m = max_{r\in R} e(r)$ the maximum delay inferred by the rules of R. C evolves to C' by a multiset R of rules (this is denoted by $C \overset{R}{\Longrightarrow} C'$) if for each membrane i the following conditions hold:

(i) $lhs_i \leq w_i$;

(ii) there is no rule $r \notin R$ such that $lhs_i^r + lhs_i \leq w_i$;

(iii) for each $a \in V$, $w_i'(a) = w_i(a) - lhs_i(a) + \sum_{s=max(0,k-m)}^{k} rhs_{i,0}^{s}(a)$.

According to (i), a configuration C has in each membrane labelled by i enough objects to enable the execution of the multiset R of rules. The maximal parallelism is captured by (ii), saying that an extra evolution rule cannot be added to R. Condition (iii) describes the effect of the rules application by adding all the objects having $j=0$ created in the last $min(k,m)$ steps which are ready to be used in the membrane system evolution. Before incrementing the global clock, all multisets $rhs_{i,j}^s$ are transformed into $rhs_{i,j-1}^s$ for $max(0,k-m) \leq s, j \leq k$.

Proposition 1. *For every timed membrane system $\Pi = (V,\mu,w_{0,1},\dots,w_{0,n},R_1,\dots,R_n,e)$ there exists an untimed membrane system $\Pi' = (V',\mu',w_{0,1}',\dots,w_{0,n}',R_1',\dots,R_n')$ that simulates the evolution of Π (restricted to the elements of V). Formally, for all $a \in V$ and $k \in \mathbb{N}$ we have $w_{k,i}(a) = w_{k,i}'(a)$, where $w_{k,i}$ and $w_{k,i}'$ are the multisets of objects from membrane i of Π and Π' at step k.*

Proof. In what follows we show how starting from a timed membrane system $\Pi = (V,\mu,w_{0,1},\dots,w_{0,n},R_1,\dots,R_n,e)$ we may construct an untimed membrane system $\Pi' = (V',\mu',w_{0,1}',\dots,w_{0,n}',R_1',\dots,R_n')$, where

- $V' = V \cup \{a_j \mid a \in V, 0 \leq j \leq m-1\}$, where $m = max_{r\in R} e(r)$;
- $\mu' = \mu$ and $w_{0,i}' = w_{0,i}$ for $1 \leq i \leq n$;
- for each rule $r : u \to v$ of R_i, $1 \leq i \leq n$ having $e(r) = 0$, we add r to R_i';
- for each rule $r : u \to v$ of R_i, $1 \leq i \leq n$ having $e(r) > 0$, we add to R_i' the following sets of rules which simulate properly the passage of $e(r)$ units of time:
  - $u \to v'$, where v' is derived from v by replacing each $a \in V$ by $a_{e(r)-1} \in V'$;
  - $a_j \to a_{j-1}$, $1 \leq j \leq e(r)-1$;
  - $a_0 \to a$.

We show that each step of the timed membrane system can be simulated by the corresponding untimed membrane system, using induction on the number of steps (time units) in timed membrane system.

Firstly, we consider a configuration $C_0 = (w_{0,1},\dots,w_{0,n},0)$ of the timed membrane system and a maximal multiset R of rules such that $C_0 \overset{R}{\Longrightarrow} C_1$. The resulting configuration $C_1 = (w_{1,1},\dots,w_{1,n},1)$ is given by $w_{1,i}(a) = w_{0,i}(a) - lhs(i)(a) + rhs_{i,0}^0(a)$ for all $1 \leq i \leq n$ and $a \in V$. Following the construction above, the initial configuration of the untimed membrane system is $C_0' = (w_{0,1}',\dots,w_{0,n}')$ where $w_{0,i}'(a) = w_{0,i}(a)$ for all $1 \leq i \leq n$ and $a \in V$. R' is the multiset of rules obtained from R such that $C_0' \overset{R'}{\Longrightarrow} C_1'$. The resulting configuration C_1' is given by $w_{1,i}'(a) = w_{0,i}'(a) - lhs_i(a) + rhs_i(a)$ for all $1 \leq i \leq n$ and $a \in V'$. This configuration contains all the elements of C_1 and some additional objects from V' introduced to simulate properly the passage of time. Regarding the elements $a \in V$, it results that $rhs_{i,0}^0(a) = rhs_i(a)$, namely $w'_{1,i}(a) = w_{1,i}(a)$. Therefore C'_1 equals C_1 regarding the elements of V (we ignore the new elements of V' because they are used only to simulate the passage of time).

Secondly, we consider a configuration $C_k = (w_{k,1},\dots,w_{k,n},k)$ of the timed membrane system and a maximal multiset R of rules such that $C_k \stackrel{R}{\Longrightarrow} C_{k+1}$. The resulting configuration $C_{k+1} = (w_{(k+1),1},\dots, w_{(k+1),n},k+1)$ is given by $w_{(k+1),i}(a) = w_{k,i}(a) - lhs(i)(a) + \sum_{s=max(0,k-m)}^{k} rhs^s_{i,0}(a)$ for all $1 \le i \le n$ and $a \in V$. In the same time, the multisets $rhs^s_{i,j}$ are transformed into $rhs^s_{i,j-1}$ for $max(0,k-m) \le s, j \le k$. Following the construction above, the configuration of the untimed membrane system is $C'_k = (w'_{k,1},\dots,w'_{k,n})$, where $w'_{k,i}(a) = w_{k,i}(a)$ for all $a \in V$, and $w'_{k,i}(a_j) = \sum_{s=max(0,k-m)}^{k} rhs^s_{i,j}(a)$ for all $a_j \in V' \backslash V$. This means that for all $1 \le i \le n$, the multiset $w'_{k,i}$ contains all the objects from $w_{k,i}$ and some additional objects from V'. For each $a \in V$ from the multiset $rhs^s_{i,j}$, the multiset $w'_{k,i}$ contains additional objects a_j. The restriction $max(0,k-m) \le s \le k$ used when creating the object a_j in membrane i means that an object a has appeared in the right hand side of a rule from timed membrane systems in the last $min(k,m)$ units of time, but has to wait j units of time until it should be added to membrane i in timed membrane systems. R' is the multiset of rules obtained from R such that $C'_k \stackrel{R'}{\Longrightarrow} C'_{k+1}$, with $w'_{k+1,i}(a) = w'_{k,i}(a) - lhs_i(a) + rhs_i(a)$ for all $1 \le i \le n$ and $a \in V'$. Moreover, in this step some objects of V' are transformed into objects of V by applying the generic rule $a_0 \to a$ (the other objects $a_j \in V'$ are transformed into objects $a_{j-1} \in V'$ by applying the generic rules $a_j \to a_{j-1}$). Finally, the number of objects $a \in V$ obtained in Π' at this step corresponds to $\sum_{s=max(0,k-m)}^{k} rhs^s_{i,0}(a)$. It results that $\sum_{s=max(0,k-m)}^{k} rhs^s_{i,0}(a) = rhs_i(a)$, namely $w'_{(k+1),i}(a) = w_{(k+1),i}(a)$. Therefore C'_{k+1} equals C_{k+1} regarding the elements of V (we ignore the elements $a_j \in V'$ because they are used only to simulate the passage of time). $\square$

In what follows we give an example that illustrates the statement of Proposition 1.

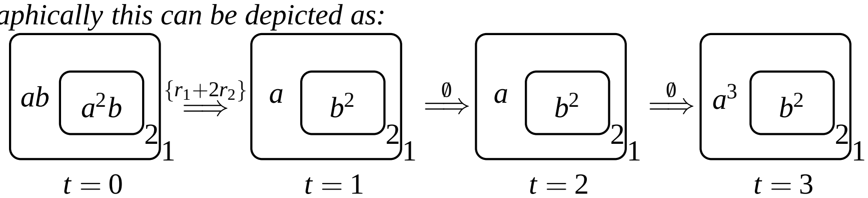
Example 2. *We consider a timed membrane system* $\Pi = (V,\mu,w_1,w_2,R_1,R_2,e)$*, where:*

- $V = \{a,b\}$; $\quad \mu = [[\,]_2]_1$; $\quad w_1 = ab$; $\quad w_2 = a^2b$;
- $R_1 = \{r_1 : b \to (b,in_2)\}$; $\quad R_2 = \{r_2 : a \to (a,out)\}$; $\quad e(r_1) = 0, e(r_2) = 2.$

Since the initial configuration of the timed membrane system Π *is* $(ab,a^2b,0)$*, then the evolution of the timed membrane system in terms of configurations is:*

$$(ab,a^2b,0) \stackrel{\{r_1+2r_2\}}{\Longrightarrow} (a,b^2,1) \stackrel{\emptyset}{\Longrightarrow} (a,b^2,2) \stackrel{\emptyset}{\Longrightarrow} (a^3,b^2,3)$$

Graphically this can be depicted as:



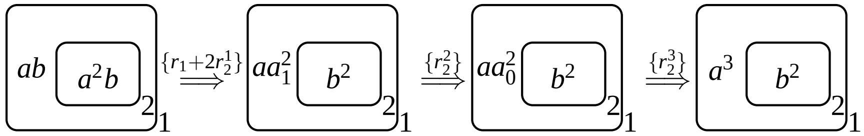
We construct an untimed membrane system $\Pi' = (V',\mu',w'_1,w'_2,R'_1,R'_2)$*, where:*

- $V' = \{a,a_0,a_1,b,b_0,b_1\}$; $\quad \mu = [[\,]_2]_1$; $\quad w_1 = ab$; $\quad w_2 = a^2b$;
- $R_1 = \{r_1 : b \to (b,in_2)\}$; $\quad R_2 = \{r^1_2 : a \to (a_1,out);\ r^2_2 : a_1 \to a_0;\ r^3_2 : a_0 \to a\}$.

Since the initial configuration of the untimed membrane system Π' *is the same as the initial configuration of the timed membrane system* Π*, namely* $(ab,a^2b,0)$*, then the evolution of the untimed membrane system in terms of configurations is:*

$$(ab,a^2b) \stackrel{\{r_1+2r^1_2\}}{\Longrightarrow} (aa^2_1,b^2) \stackrel{\{r^2_2\}}{\Longrightarrow} (aa^2_0,b^2) \stackrel{\{r^3_2\}}{\Longrightarrow} (a^3,b^2)$$

Graphically this can be depicted as:



If we are interested only in the symbols of V in the untimed evolution, then we have:

$$\begin{array}{ccccccc}
(ab,\ a^2b,\ 0) & \stackrel{\{r_1+2r_2\}}{\Longrightarrow} & (a,\ b^2,\ 1) & \stackrel{\emptyset}{\Longrightarrow} & (a,\ b^2,\ 2) & \stackrel{\emptyset}{\Longrightarrow} & (a^3,\ b^2,\ 3) \\
\| \quad \| & & \| \quad \| & & \| \quad \| & & \| \quad \| \\
(ab\cap V, a^2b\cap V) & \stackrel{\{r_1+2r_2^1\}}{\Longrightarrow} & (aa_1^2\cap V, b^2\cap V) & \stackrel{\{r_2^2\}}{\Longrightarrow} & (aa_0^2\cap V, b^2\cap V) & \stackrel{\{r_2^3\}}{\Longrightarrow} & (a^3\cap V, b^2\cap V)
\end{array}$$

and thus the statement of Proposition 1 holds.

It is easy to prove that the class of timed membrane systems includes the class of untimed membrane systems, since we can assign 0 to all the rules by the timing function e.

3 Timed Petri Nets with Localities

An extension of Petri nets with localities is defined by adding delays to transitions (like in coloured Petri nets [10]). The value of the global clock is kept in a variable gc.

Definition 2. *A timed Petri net with localities $\mathcal{N} = (P,T,W,L,D,M_0)$ is given by:*

(i) *finite disjoint sets P of places and T of transitions;*

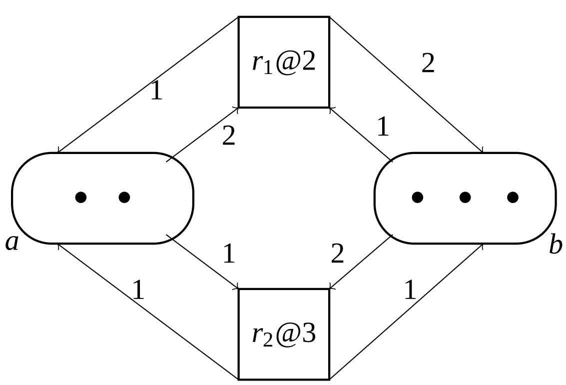
(ii) *a weight function $W : (T \times P) \cup (P \times T) \to \mathbb{N}$;*

(iii) *a locality mapping $L : T \to \mathbb{N}$;*

(iv) *a delay mapping $D : T \to \mathbb{N}$;*

(v) *an initial marking $M_0 : P \cup \{gc\} \to \mathbb{N}$.*

If $W(x,y) \geq 1$ for some $(x,y) \in (T \times P) \cup (P \times T)$, then (x,y) is an arc from the place (transition) x to the transition (place) y. The locality mapping L defines sets of transitions called localities (depending on the number associated to each transition). The delay mapping D introduces a time delay to each object created by a transition; the delays indicate how long the objects cannot be used in other transitions. The initial marking M_0 assigns to each place a number of tokens, and value 0 to the global clock gc.



Places are drawn as rounded lines with tokens placed inside. A transition is drawn as a rectangle containing a label, and the delay it introduces for the newly created tokens. Transitions are connected to places by weighted directed arcs.

Figure 2: A Timed Petri Net

Markings represent global states of the timed Petri nets with localities, and they are defined as functions from $P \cup \{gc\}$ to $\mathbb{N}$. A Petri net $\mathcal{N}$ evolves at a time step k from a marking M to another marking M' by a multiset of transitions $U : T \to \mathbb{N}$ (e.g., $U(tr) = 2$ for $tr \in T$ means that U contains twice the transition tr). If the multiset U of transitions is empty, then the only action is incrementing the global clock gc. Given a multiset of transitions U, we denote by $pre(U)(p) = \sum_{tr \in U} U(tr) \cdot W(p,tr)$ the multiset of tokens associated to the input arcs $(P \times T)$ of all transitions $tr \in U$. In a similar way, by $post^k_j(p) = \sum_{tr \in U;\, D(tr)=j} U(tr) \cdot W(tr,p)$ is denoted the multiset of tokens associated to the output arcs $(T \times P)$ which are added to their corresponding places after j units of time (k represents the current time). We denote by $m' = max_{tr \in U} D(tr)$ the maximum delay inferred by the transitions of U. A marking M leads in a max-enabled way to a marking M' via a multiset U of transitions (denoted by $M[U\rangle_{max}M'$) if $M'(gc) = M(gc) + 1$ and for each place $p \in P$ the following conditions hold:

(i) $pre(U)(p) \leq M(p)$;

(ii) there is **no** transition $tr \in U$ such that $pre(\{tr\})(p) + pre(U)(p) \leq M(p)$;

(iii) $M'(p) = M(p) - pre(U)(p) + \sum^k_{s=max(0,k-m')} post^s_0(p)$.

According to (i), a marking M has in each place p enough tokens to enable the execution of the multiset U of transitions. The maximal parallelism is captured by (ii), saying that an extra transition cannot be added to U. Condition (iii) describes the effect of the transitions application by adding all the tokens having $j = 0$ created in the last $min(k,m')$ steps which are ready to be used in Petri nets evolution. Before incrementing the global clock, all the multisets $post^s_j(p)$ are transformed into $post^s_{j-1}(p)$ for $max(0,k-m') \leq s, j \leq k$.

Proposition 2. *For every timed Petri net with localities $\mathcal{N} = (P,T,W,L,D,M_0)$ there exists a Petri net with localities $\mathcal{N}' = (P',T',W',L',M'_0)$ that simulates the evolution of $\mathcal{N}$ (with respect to places of P). Formally, for all $p \in P$ and $k \in \mathbb{N}$ we have $M_k(p) = M'_k(p)$, where M_k and M'_k are markings of $\mathcal{N}$ and $\mathcal{N}'$ at step k.*

Proof. In what follows we show how starting from a timed Petri net with localities $\mathcal{N} = (P,T,W,L,D,M_0)$, we construct an untimed Petri net with localities $\mathcal{N}' = (P',T',W',L',M'_0)$, where

- for every $p \in P$ and $tr \in T$ such that $W(p,tr) > 0$, we consider additional places $p, p^0_{tr}, \ldots, p^{D(tr)-1}_{tr}$ in P'; if $D(tr) = 0$ then only $p \in P'$;
- for every $tr \in T$ and $p \in P$ such that $W(tr,p) > 0$, we consider additional transitions $tr, tr^0, \ldots, tr^{D(tr)-1}$ in T'; if $D(tr) = 0$ then $tr \in T'$;
- for every $p \in P$ and $tr \in T$ such that $W(p,tr) > 0$, we consider the weights $W'(p,tr)$ in $\mathcal{N}'$:
  - if $D(tr) = 0$ then $W'(p,tr) = W(p,tr)$;
  - if $D(tr) > 0$ then $W'(p,tr) = W(p,tr)$, and
    $W'(tr, p^{D(tr)-1}_{tr}) = W'(p^j_{tr}, tr^j) = W'(tr^i, p^{i-1}_{tr}) = W(tr,p)$ for $0 \leq j < i \leq D(tr) - 1$;
- for every $p \in P$ and $tr \in T$ such that $W(tr,p) > 0$, we consider the following weights $W'(tr,p)$ in $\mathcal{N}'$: if $D(tr) = 0$ then $W'(tr,p) = W(tr,p)$, else $W'(tr^0,p) = W(tr,p)$;
- for every $tr \in T$, we take the same locality label $l = L(tr)$ for the new transitions $tr, tr^0, \ldots, tr^{D(tr)-1}$;
- if $p \in P$ then $M'_0(p) = M_0(p)$, and if $p \in P' \backslash P$ then $M'_0(p) = 0$.

We show that each step of the timed Petri nets with localities can be simulated by the corresponding untimed Petri nets with localities; we prove this by induction on the number of steps (time units) in timed Petri nets with localities.

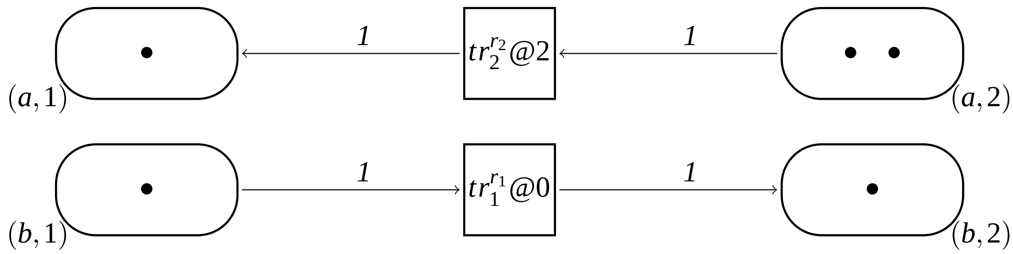
Firstly, we consider a marking M_0 of the timed Petri net with localities and a multiset of transitions U such that $M_0[U\rangle_{max}M_1$. The resulting marking M_1 is given by $M_1(p) = M_0(p) - pre(U)(p) + post_0^0(p)$ for all $p \in P$. Following the construction above, the initial marking of the untimed Petri net with localities is M'_0, where $M'_0(p) = M_0(p)$ for all $p \in P'$. U' is the multiset of transitions obtained from U such that $M'_0[U'\rangle_{max}M'_1$. The resulting marking M'_1 is given by $M'_1(p') = M'_0(p') - pre(U')(p) + post(U')(p)$ for all $p' \in P'$, where $post(U')(p) = \sum_{tr \in U'}(U'(tr) \cdot W'(tr,p))$. This marking contains all the places of M_1 and some additional places from P'. Regarding the places $p \in P$, it results that $post_0^0(p) = post(U')(p)$, namely $M'_1(p) = M_1(p)$. Therefore M'_1 equals M_1 regarding the number of tokens from the places of P (we ignore the new places of P' because they do not play any role at this step).

Secondly, we consider a marking M_k of the timed Petri net with localities and a multiset U of transitions such that $M_k[U\rangle_{max}M_{k+1}$. The resulting configuration M_{k+1} is given by $M_{k+1}(p) = M_k(p) - pre(U)(p) + \sum_{s=max(0,k-m')}^{k} post_0^s(p)$ for all $p \in P$. In the same time, the multisets of tokens $post_j^s(p)$ are renamed by $post_{j-1}^s(p)$ for all $max(k-m',0) \leq s, j \leq k$ and $p \in P$. Following the construction above, the marking of the untimed Petri net with localities is M'_k, where $M'_k(p) = M_k(p)$ for all $p \in P$, and $M'_k(p_{tr}^j) = \sum_{s=max(0,k-m')}^{k} post_j^s(p)$ for all additional $p_{tr}^j \in P' \backslash P$, $tr \in T$ and $0 \leq j \leq D(tr) - 1$. This means that the common places of both nets have the same number of tokens, while for the additional places appearing only in P' we add tokens such that for each token from $post_j^s(p)$ obtained after firing the transition tr, the place p_{tr}^j contains a token. The restriction $max(0,k-m') \leq s \leq k$ (used when creating a token in a new place p_{tr}^j of P') means that a token appears on an output arc of transition tr in timed Petri nets during the last $min(k,m')$ units of time; this token has to wait j units of time until it is added to place p of P. The multiset of rules U' is obtained from U such that $M'_k[U'\rangle_{max}M'_{k+1}$, with $M'_{k+1}(p) = M'_k(p) - pre(U')(p) + post(U')(p)$ for all $p \in P'$. Moreover, in this step some tokens are transferred from places of P' into places of P by firing the transitions tr^0 (the other tokens from places $p_{tr}^j \in P'$ are transferred into places $p_{tr}^{j-1} \in P'$ by firing the transitions tr^j). Thus, the number of tokens obtained in places $p \in P$ at each step k is equal to $\sum_{s=k-m'}^{k} post_0^s(p)$. It results that $\sum_{s=k-m'}^{k} post_0^s(p) = post(U')(p)$, namely $M'_{k+1}(p) = M_{k+1}(p)$ for all $p \in P$. Therefore M'_{k+1} equals M_{k+1} regarding the number of tokens from the places of P (we ignore the remaining places $p_{tr}^j \in P'$ because they are used only to simulate the passage of time). $\square$

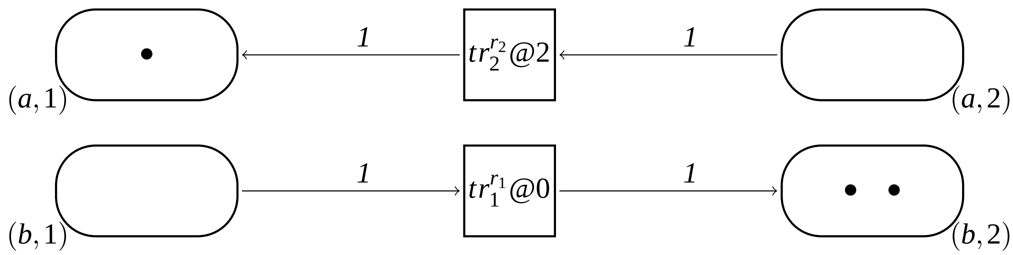
Example 3. *We consider a timed Petri net with localities* $\mathscr{N} = (P,T,W,L,D,M_0)$*, where*

- $P = \{(a,1),(a,2),(b,1),(b,2)\}$; $\quad T = \{tr_1^{r_1}, tr_2^{r_2}\}$;
- $D(tr_1^{r_1}) = 0$; $\quad D(tr_2^{r_2}) = 2$; $\quad L(tr_1^{r_1}) = 1$; $\quad L(tr_2^{r_2}) = 2$;
- $W((a,1),t_2^{r_2}) = W(tr_2^{r_2},(a,2)) = W((b,1),t_1^{r_1}) = W(tr_1^{r_1},(b,2)) = 1$
- $M_0((a,1)) = M_0((b,1)) = M_0((b,2)) = 1$; $\quad M_0((a,2)) = 2$; $\quad M_0(gc) = 0$.

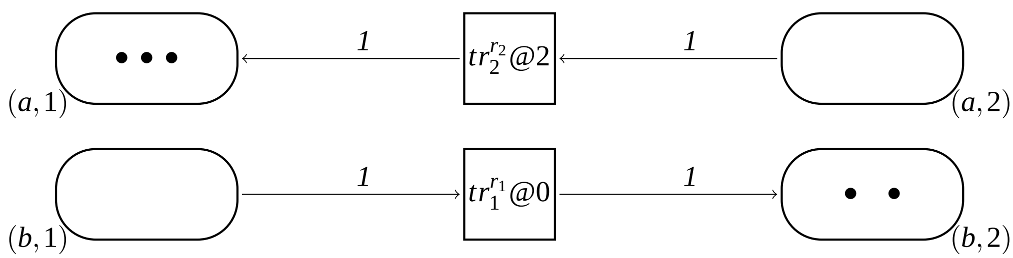
Graphically the system at time unit $gc = 0$ *can be represented as follows:*



For $gc = 1$ *and* $gc = 2$*, the timed Petri net with localities can be represented as follows:*



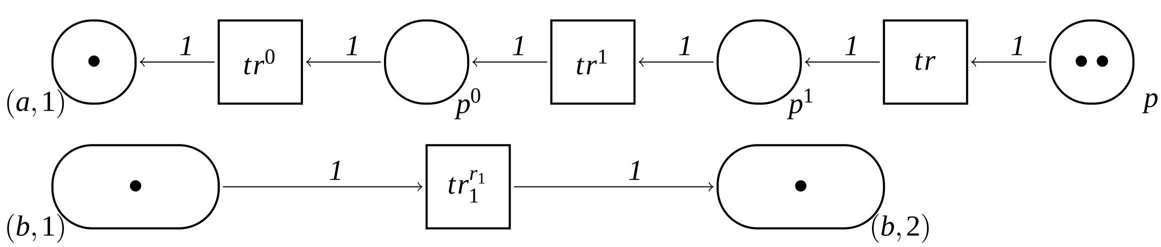
while for all $gc \geq 3$ *we have the following representation*



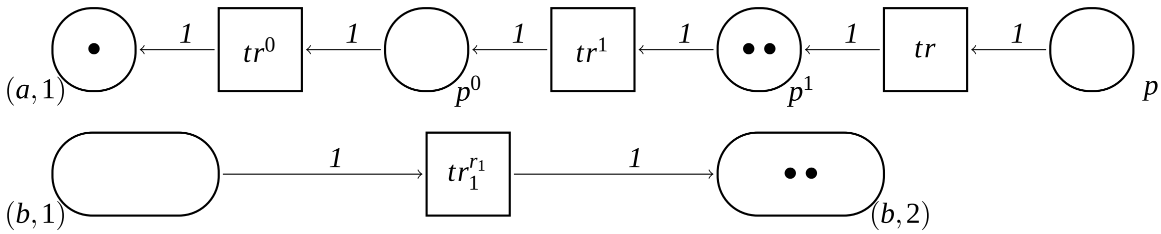
We construct an untimed Petri net with localities $\mathscr{N}' = (P', T', W', L', M'_0)$*, where*

- $P = \{(a,1), p, p^0, p^1, (b,1), (b,2)\}$ *and* $T = \{tr_1^{r_1}, tr, tr^0, tr^1\}$*, where* $p = (a,2)$ *and* $tr = tr_2^{r_2}$*;*
- $L(tr_1^{r_1}) = 1$*;* $\qquad L(tr) = L(tr^0) = L(tr^1) = 2$*;*
- $W((b,1), t_1^{r_1}) = W(tr_1^{r_1}, (b,2)) = 1$
- $W(p,tr) = W(tr, p^1) = W(p^1, tr^1) = W(tr^1, p^0) = W(p^0, tr^0) = W(tr^0, (a,1)) = 1$
- $M_0((a,1)) = M_0((b,1)) = M_0((b,2)) = 1$*;* $\qquad M_0(p) = 2$*;* $\qquad M_0(p^0) = M_0(p^1) = 0.$

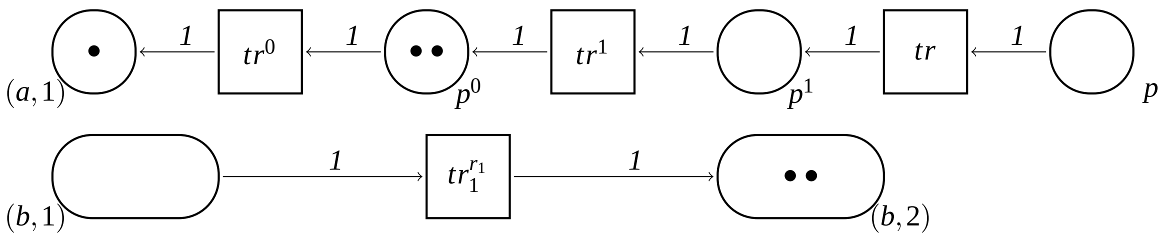
Graphically, the initial system can be represented as follows:



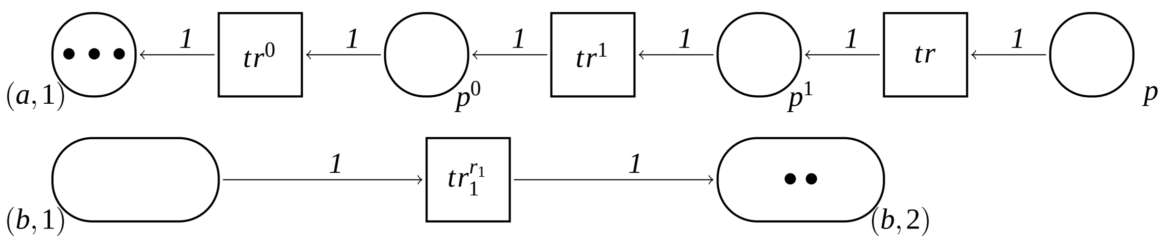
After one step, we obtain



The system evolves to



The system stops its evolution after reaching the configuration



We notice that indeed, if we refer only to the markings of the places from P during the evolution of timed and untimed Petri nets with localities, the markings are the same.

It is easy to prove that the class of timed Petri net with localities includes the class of Petri net with localities, since we can assign 0 to all values of the function D, namely all transitions fire instantaneously.

4 Linking Timed Membrane Systems to Timed Petri Nets

Following the approach given in [12] where membrane systems are translated into Petri nets with localities, we present a translation of timed membrane systems into timed Petri nets with localities, and then prove an operational correspondence between them.

Definition 3. *Let $\Pi = (V, H, \mu, w_1, \dots, w_n, R_1, \dots, R_n, e)$ be a timed membrane system. Then the corresponding timed Petri net with localities is $\mathcal{N}_\Pi = (P, T, W, L, D, M_0)$ with its components defined as follows:*

- $P = V \times \{1, \dots, n\}$ - *to each object a of membrane i there corresponds a place $p = (a, i)$;*
- $T = \{tr_j^r \mid r \in R_j, 1 \le j \le n\}$ - *to each rule r of membrane j corresponds a transition tr_j^r;*
- *for every place $p = (a, i) \in P$ and every transition $tr = tr_j^r \in T$*

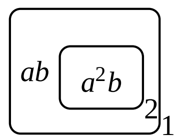
$$W(p,tr) = \begin{cases} lhs_i^r(a) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \text{ and } W(tr,p) = \begin{cases} rhs_{i,e(r)}^{r,0}(a) & \text{if } i = j \\ rhs_{i,e(r)}^{r,0}((a,out)) & \text{if } (i,j) \in \mu \\ rhs_{i,e(r)}^{r,0}((a,in_j)) & \text{if } (j,i) \in \mu \\ 0 & \text{otherwise} \end{cases}$$

- *for every place* $p = (a,i) \in P$*, we have* $M_0(p) = w_i(a)$*;*
- *for every transition* $tr_j^r \in T$*, we have* $L(t) = j$*;*
- *for every* $tr_j^r \in T$*, we have* $D(tr) = e(r)$*.*

Example 4. *We consider a timed membrane system* $\Pi = (V, \mu, w_1, w_2, R_1, R_2, e)$*, where*

- $V = \{a,b\}$*;* $\mu = [[\ ]_2]_1$*;* $w_1 = ab$*;* $w_2 = a^2b$*;*
- $R_1 = \{r_1 : b \to (b, in_2)\}$*;* $R_2 = \{r_2 : a \to (a, out)\}$*;* $e(r_1) = 0$, $e(r_2) = 2$.

Graphically, the initial configuration can be depicted as:

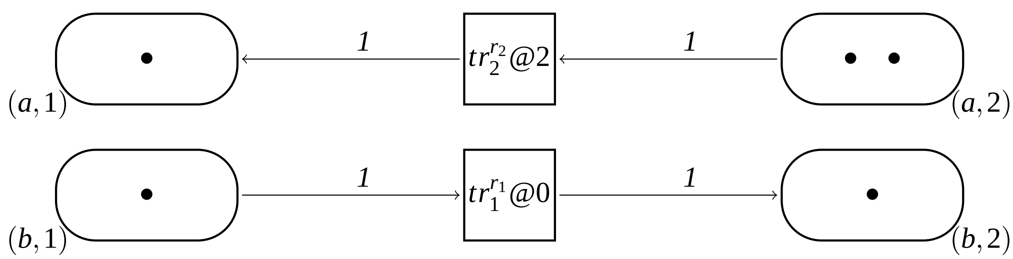


$t = 0$

The corresponding timed Petri net with localities is $\mathscr{N} = (P, T, W, L, D, M_0)$*, where:*

- $P = \{(a,1),(a,2),(b,1),(b,2)\}$*;* $T = \{tr_1^{r_1}, tr_2^{r_2}\}$*;*
- $D(tr_1^{r_1}) = 0$*;* $D(tr_2^{r_2}) = 2$*;* $L(tr_1^{r_1}) = 1$*;* $L(tr_2^{r_2}) = 2$*;*
- $W((a,1), t_2^{r_2}) = W(tr_2^{r_2}, (a,2)) = W((b,1), t_1^{r_1}) = W(tr_1^{r_1}, (b,2)) = 1$
- $M_0((a,1)) = M_0((b,1)) = M_0((b,2)) = 1$*;* $M_0((a,2)) = 2$*;* $M_0(gc) = 0$.

Graphically, the system at time unit $gc = 0$ *can be represented as*



According to this translation, M_C denotes the marking of $\mathscr{N}_\Pi$ corresponding to a configuration C of the timed membrane system Π. Moreover, for each multiset R of applied rules in a timed membrane system, the corresponding multiset of transitions in timed Petri nets with localities is denoted by U_R. Using these notations, we have the following operational correspondence:

Proposition 3. $C \xRightarrow{R} C'$ *if and only if* $M_C[U_R\rangle_{max} M_{C'}$*.*

Proof. Let us consider the membrane configuration $C = (w_1, \ldots, w_n)$. According to Definition 3, we have $M_C(p) = w_i(a)$ for each place $p = (a,i)$. This is a consequence of the fact that there is a correspondence between membranes and places, and between the multiset inside membranes and the marking of the places. After applying the multiset R of rules in C, we obtain a configuration $C' = (w_1', \ldots, w_n')$ where for each membrane i and each object a we have $w_i'(a) = w_i(a) - lhs_i(a) + \sum_{s=max(0,k-m)}^{k} rhs_{i,0}^{s}(a)$. In the corresponding timed Petri net with localities, starting from the marking M_C and applying the multiset U_R of transitions, we obtain a new marking M' where for each place p we get $M'(p) = M_C(p) - pre(U_R)(p) + \sum_{s=max(0,k-m')}^{k} post_0^s(p)$. It is easy to note that $M_C(p) = w_i(a)$, $pre(U_R)(p) = lhs_i(a)$ and $\sum_{s=max(0,k-m')}^{k} post_0^s(p) = \sum_{s=max(0,k-m)}^{k} rhs_{i,0}^{s}(a)$. Therefore, it results that $M'(p) = w_i'(a)$ and $M' = M_C'$. $\square$

5 Conclusion

There exist papers in the field of membrane computing in which the concept of time is used mainly as timers for objects and membranes [1, 2], and as execution period for each rule [4, 5]. The idea of adding time to Petri nets is described in [16]: "addition of timing information might provide a powerful new feature for Petri nets, but may not be possible in a manner consistent with the basic philosophy of Petri nets". Different ways of incorporating timing information into Petri nets were proposed by many researchers; specific application fields represent the inspiration for different proposals of modelling time. For Petri nets with localities [12], time constrains are added in a way inspired by the coloured Petri nets.

In this paper we prove that adding timing to both membrane systems and Petri nets with localities does not increase the expressive power of the corresponding untimed formalisms, establish a link between these timed formalisms by defining a relationship between timed formalisms under the assumption of maximal firing, and prove an operational correspondence between them. This relationship allows to use the Petri nets tools to verify certain behavioural properties (reachability, boundedness, liveness and fairness) of membrane systems. An attempt to use Petri nets software to simulate timing aspects in membrane systems is presented in [18].

As further work, we can mention the use of timed membrane systems to model some biological systems, while Petri nets tools can be used to analyze and verify automatically the (timing) behavioural properties of these models.

Acknowledgements. The work of Bogdan Aman is supported by POSDRU/89/1.5/S/49944.

References

[1] B. Aman, G. Ciobanu. Mutual Mobile Membranes with Timers. *Electronic Proceedings in Theoretical Computer Science*, vol.6, 1–15, 2009. doi:10.4204/EPTCS.6.1

[2] B. Aman, G. Ciobanu. Adding Lifetime to Objects and Membranes in P Systems. *Int. Journal of Computers, Communication & Control*, vol.5(3), 268–279, 2010.

[3] O. Andrei, G. Ciobanu, D. Lucanu. A Rewriting Logic Framework for Operational Semantics of Membrane Systems. *Theoretical Computer Science*, vol.373, 163–181, 2007. doi:10.1016/j.tcs.2006.12.016

[4] M. Cavaliere, D. Sburlan. Time-Independent P Systems. *Lecture Notes in Computer Science*, vol.3365, 239–258, 2005. doi:10.1007/978-3-540-31837-8_14

[5] M. Cavaliere, D. Sburlan. Time in Membrane Computing. *Oxford Handbook of Membrane Computing*, 594–604, 2010.

[6] G. Ciobanu. *Membrane Computing and Biologically Inspired Process Calculi*. "A.I.Cuza" University Press, Iaşi, 2010.

[7] G. Ciobanu, Gh. Păun, M.J. Pérez-Jiménez. *Applications of Membrane Computing*, Springer, Natural Computing Series, 2006. doi:10.1007/3-540-29937-8

[8] G. Ciobanu. Semantics of P Systems. *Oxford Handbook of Membrane Computing*, 413–436, 2010.

[9] S. Dal Zilio, E. Formenti. On the Dynamics of PB Systems: a Petri Net View. *Lecture Notes in Computer Science*, vol.2933, 153–167, 2004. doi:10.1007/978-3-540-24619-0_11

[10] K. Jensen. *Coloured Petri Nets; Basic Concepts, Analysis Methods and Practical Use. Vol. 1,2,3*. Monographs in Theoretical Computer Science, Springer, 1992, 1994, 1997.

[11] J. Kleijn , M. Koutny. Petri Nets and Membrane Computing. *Oxford Handbook of Membrane Computing*, 389–412, 2010.

[12] J. Kleijn , M. Koutny, G. Rozenberg. Towards a Petri Net Semantics for Membrane Systems. *Lecture Notes in Computer Science*, vol.3850, 292–309, 2006. doi:10.1007/11603047_20

[13] H. Lodish, A. Berk, P. Matsudaira, C. Kaiser, M. Krieger, M. Scott, L. Zipursky, J. Darnell. *Molecular Cell Biology*, 6th Edition, Freeman, 2008.

[14] P. M. Merlin. *A Study of the Recoverability of Computing Systems*. PhD thesis, Department of Information and Computer Science, University of California, Irvine, CA, 1974.

[15] Gh. Păun. *Membrane Computing. An Introduction*. Springer, 2002. doi:10.1007/BF03037282

[16] J.L. Peterson. *Petri Net Theory and the Modeling of Systems*. Prentice-Hall, 1981.

[17] M. Pezzé and M. Young. Time Petri Nets: A Primer Introduction. *Multi-Workshop on Formal Methods in Performance Evaluation and Applications*, 1999.

[18] A. Profir, E. Guţuleac, E. Boian. Encoding Continuous-Time P Systems with Descriptive Times Petri Nets. *TAPS'05, IEEE Computer Press*, 91–94, 2005.

[19] C. Ramchandani. *Analysis of Asynchronous Concurrent Systems by Timed Petri Nets*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1974.

[20] J. Sifakis. Performance Evaluation of Systems using Nets. *Lecture Notes in Computer Science*, vol. 84, 307–319, 1980. doi:10.1007/3-540-10001-6_30

[21] Z. Qi, J. You, H. Mao. P Systems and Petri Nets. *Lecture Notes in Computer Science*, vol.2933, 286–303, 2004. doi:10.1007/978-3-540-24619-0_21