Rule-based and object-based event structures for membrane systems

Oana Agrigoroaiei, Gabriel Ciobanu *

Romanian Academy, Institute of Computer Science, Blvd. Carol I No. 8, 700505 Iași, Romania
“A.I.Cuza” University of Iași, Romania

ARTICLE INFO

Article history:
Available online 2 April 2010

Keywords:
Membrane computing
Event structures
Operational semantics

ABSTRACT

We introduce two event structures for basic membrane systems with one membrane. In the rule-based structure an event is given by a single rule application, while in the object-based structure an event is given by the occurrence of a type of resource (in a certain quantity). Both event structures are introduced without the use of fresh names to distinguish between similar events. We discuss causality by using the order relation on the object-based event structure as well as a dependence relation on the rules of the membrane system.

© 2010 Published by Elsevier Inc.

1. Introduction

In this paper we study the nature of parallelism and nondeterminism of the membrane systems in terms of a widely recognized formal model for parallelism and nondeterminism, namely in terms of event structures [9].

Membrane computing is an area of computer science aiming to abstract computing models from the structure and the functioning of living cells. Membrane systems are essentially parallel and nondeterministic computing models processing multisets of objects in a localized manner (evolution rules and evolving objects are encapsulated into compartments delimited by membranes). An essential role is played by the communication between compartments. Starting from this rough description of a basic membrane system (also called P system), many different variants inspired by different aspects of living cells were defined; several of them are presented in [6]. Actually the large variety of suggestions from biology provide many possibilities in defining a membrane system, and the existing literature contains many models. We can say that the theory of membrane systems is not restricted to a specific model, and in fact we work in a framework of models. In this framework there are some notions, notations and models which are “standard”, and could be considered as basic elements of membrane systems.

In this paper we consider transition P systems with communication encoded into a single membrane. The rule-based event structures for such a membrane system consider events of the form \((\text{history}, \text{multiset})\) where \text{history} describes the multisets of rules applied at each step of the evolution, and \text{multiset} describes the resulting objects after applying these steps. Surprisingly, this approach avoids the use of fresh names in defining the event structure for membrane systems. While using fresh names makes it easier to define the associated event structure [2] or event-based semantics [1], it assumes the presence of a name generator and produces non-unique structures. It is proved that the newly defined order and conflict relations form an event structure, and this event structure is consistent with the single step operational semantics of membrane systems. The object-based event structures are defined over events of the form \((a^k, \text{history})\) where \(a^k\) represents the occurrence of \(k\) objects \(a\) after applying the multisets of rules described by \text{history}. It is proved that the new order and conflict relations over these events form an event structure, and this event structure is consistent with the maximally parallel step operational semantics of membrane systems.

* Corresponding author at: Romanian Academy, Institute of Computer Science, Blvd. Carol I No. 8, 700505 Iași, Romania. E-mail addresses: oanaag@iit.tuiasi.ro (O. Agrigoroaiei), gabriel@iit.tuiasi.ro (G. Ciobanu).

1567-8326/$ - see front matter © 2010 Published by Elsevier Inc.
doi:10.1016/j.jlap.2010.03.010
The motivation of introducing these event structures is given by the search for a clear definition of causality for membrane systems. We see that a unique definition may not be possible, since each of these structures comes with its own notion of causality given by its order relation. The object-based event structures emphasize the available resources at each step, while the rule-based event structures emphasize the application of rules. Their causalities are related, and we show such a relationship. Thus the object-based event structures seem more appropriate for studying the quantitative evolution of the membrane systems.

1.1. Multisets and membrane systems

The compartments of a cell contain several substances (ions, small molecules, macromolecules); there is no ordering, everything is close to everything. However, the concentration matters, i.e., the number of copies of each molecule is important. This means that we should work with sets of objects whose multiplicities matters, namely multisets.

We employ the description of multisets as functions; namely, a multiset over a set S is a function \( w: S \rightarrow \mathbb{N} \). We denote by MS the set of multisets over S. When describing a multiset characterized by, for example, \( w(5) = 1, w(2) = 2, w(3) = 0 \), we use its string representation \( s^2 \), to simplify its description. The support of a multiset \( w \) is the set of elements of S which have a non-zero image. The multiplicity of \( s \) in \( w \) is the number \( w(s) \). We use the (abusive) notation \( s \in w \) whenever \( w(s) > 0 \). A multiset is called non-empty if it has non-empty support. We denote the empty multiset by \( \emptyset \).

The sum of two multisets \( w \), \( w' \) over \( S \) is the multiset \( w + w': S \rightarrow \mathbb{N}, (w + w')(s) = w(s) + w'(s) \). For two multisets \( w, w' \) over \( S \) we say that \( w \) is contained in \( w' \) if \( w(s) \leq w'(s), \forall s \in S \). We denote this by \( w \leq w' \). If \( w \leq w' \) we can define \( w' - w \) by \( (w' - w)(s) = w'(s) - w(s) \). For two multisets \( w, w' \) over \( S \) we denote by \( w \cap w' \) the multiset over \( S \) defined by \( w \cap w'(s) = \min\{w(s), w'(s)\} \). For further simplification, for an \( s \in S \) we use \( s \) also to denote the multiset over \( S \) in which the element \( s \) has multiplicity 1 and the rest of elements of \( S \) have multiplicity 0.

Multiset processing of objects in a membrane system is given by rewriting rules of the form \( u \rightarrow v \), where \( u \) and \( v \) are multisets of objects (represented by strings). For a rule \( r: u \rightarrow v \) we use the notations \( lhs(r) = u \) and \( rhs(r) = v \). These notations are extended to multisets \( f \) of rules: \( lhs(f) \) is defined by \( lhs(f)(s) = \sum_{r \in f} lhs(r)(s) \cdot f(r) \) (where \( \cdot \) is the multiplication over the set \( \mathbb{N} \) of natural numbers) and \( rhs(f) \) is defined similarly.

Membrane computing has two important features. The first one is related to nondeterminism. Since we do not distinguish among several copies of an object, they are considered identical; the rules compete for the available objects, and so the rules and the objects are chosen in a nondeterministic manner. The second one is related to parallelism. Cell biochemistry is not only nondeterministic, but it is also parallel. If two (types of) objects can react, then the reaction can take place for all copies of the objects. This aspect suggests the maximal parallelism used in membrane computing: at each step, rules are applied (to the available objects) until no further application of rules is possible.

Basic membrane systems are usually called transition \( P \) systems. When presenting such a system we have to specify the alphabet \( O \) of objects (a finite nonempty alphabet of abstract symbols identifying the objects), the membrane structure, the multisets of objects present in each region of the system (represented by strings), the sets of evolution rules associated with each region, and the indication about the way the output is defined.

Formally, a transition \( P \) system (of degree \( m \geq 1 \)) is a construct of the form

\[
\Pi = (O, \mu, w_1, w_2, \ldots, w_m, R_1, R_2, \ldots, R_m, i_0),
\]

where

1. \( O \) is the finite and nonempty alphabet of objects;
2. \( \mu \) is a membrane structure, consisting of \( m \) membranes labelled by \( 1, \ldots, m \);
3. \( w_1, w_2, \ldots, w_m \) are strings over \( O \), representing the multisets of objects existing in regions \( 1, 2, \ldots, m \) of the membrane structure;
4. \( R_1, R_2, \ldots, R_m \) are finite sets of evolution rules associated with regions \( 1, 2, \ldots, m \) of the membrane structure;
5. \( i_0 \) is either one of the labels \( 1, 2, \ldots, m \), and the respective region is the output region of the system, or it is 0 indicating that the result of a computation is collected in the environment of the system.

The rules are of the form \( u \rightarrow v \) or \( u \rightarrow v_i \), with \( u \in O^* \), where by \( O^* \) we denote the set of all strings over \( O \) (the empty string \( \epsilon \) included), and by \( O^+ \) we denote the set \( O^* \setminus \{\epsilon\} \) of all nonempty strings over \( O \). If at least one of the rules of a membrane introduces the dissolving symbol \( \delta \), then the membrane is dissolved and its contents become part of the parent membrane. Communication is described by using \( v \in (O \times Tar)^* \), where \( Tar = \{here, in, out\} \). The rules can be either cooperative (with \( u \) arbitrary in \( O^+ \)), or non-cooperative (with \( u \in O \)). A possible restriction about the region \( i_0 \) in the case when it is an internal one is to consider only regions enclosed by elementary membranes for output.

The membrane structure and the multisets of objects from its compartments identify a configuration of a \( P \) system. The initial configuration is given by \((\mu, w_1, \ldots, w_m)\) specifying the membrane structure and the multisets of objects available in its compartments at the beginning of a computation. During the evolution of the system, both the multisets of objects and the membrane structure can change. A transformation of a configuration of the system is called a transition, and it takes place by applying the rules in each region in a nondeterministic and maximally parallel manner. A sequence of transitions constitutes a computation. A computation is successful if it halts, namely reaches a configuration where no rule can be applied to the existing objects, and the output region \( i_0 \) still exists in the halting configuration (membrane \( i_0 \) is not dissolved during the
computation). The result of a successful computation is usually represented by the number of objects present in the output region in the halting configuration. If we distinguish among different objects, the result can be a vector of natural numbers. Non-halting computations provide no output.

Starting from an initial configuration and because of the nondeterminism in the application of rules, we can get several successful computations and their associated results. Thus, a membrane system computes (or generates) a set of numbers (or a set of vectors of numbers, or a language). Several computability results are presented in [6], together with several other results involving formal languages and grammars, register machines and complexity theory. Several applications of these systems are presented in [3]. An updated bibliography can be found at the P systems web page http://ppage.psystems.eu.

In this paper we simplify the presentation by considering basic membrane systems with one membrane. This model is enough to encode transition P systems with communication, either cooperative or non-cooperative, by pairs $(u, i)$ of objects and labels together with the corresponding rules over these pairs [4]. Thus, we consider systems without dissolution, promoters and inhibitors. We expect that the constructions of the two event structures could be extended to general membrane systems.

Therefore we work with P systems $\Pi$ defined as $\Pi = (O, w_0, R)$, where $w_0$ is the initial multiset of objects placed in the membrane and $R$ is a set of rules of form $r : u \rightarrow v$, with $u$, $v$ multisets of objects. For such a simplified system, a configuration is given by a multiset of objects.

**Definition 1.** A multiset of rules $f$ is maximally valid with respect to a multiset of objects $w$ if $lhs(f) \subseteq w$ and there is no rule $r$ such that $lhs(f + r) \subseteq w$.

### 1.2. Operational semantics of membrane systems

The maximally parallel evolution of a membrane system can be described operationally in two ways. The first one is a maximally parallel step semantics in which all rules are considered to be applied at once. We refer to a transition in the maximally parallel step transition system as an evolution step. The states of the maximally parallel step transition system are the configurations of the membrane system.

**Definition 2.** For two multisets $w$, $w'$ over $O$ we set $w \xrightarrow{f} w'$ whenever $f$ is a multiset of rules which is maximally valid with respect to $w$ and $w' = w - rhs(f) + rhs(f)$.

The second semantics is called a single step semantics, the rules being applied one at a time. For this reason, the states of the transition system are pairs $(u, v)$, where the first element $u$ is the multiset of objects available for rule application and the second, $v$, is the multiset of objects that have been produced by rule application in the same maximally parallel step of the evolution.

**Definition 3.** For two pairs of multisets $(u, v)$ and $(u', v')$ we set:

- $(u, v) \xrightarrow{f} (u', v')$ whenever $lhs(r) \subseteq u$ and $u' = u - lhs(r)$, $v' = v + rhs(r)$;
- $(u, v) \xrightarrow{f} (u', v')$ whenever there exists no rule $r$ such that $lhs(r) \subseteq u$ and when $u' = u + v$, $v' = 0$.

**Remark 4.** A maximally parallel step consists of a series of single steps:

$w \xrightarrow{f} w'$ if and only if $(w, 0) \xrightarrow{f_1} \cdots \xrightarrow{f_k} (w - lhs(f), rhs(f)) \rightarrow (w', 0)$, with $f = f_1 + \cdots + f_k$.

### 1.3. Event structures

Event structures are defined for systems described as a set of events (usually action occurrences) together with a causality relation over these events. The causality between actions is expressed by a partial order, and the nondeterminism is expressed by a conflict relation on actions. For every two events $d$ and $e$ it is specified either whether one of them is a prerequisite for the other, whether they exclude each other, or whether they may happen in parallel. The behaviour of an event structure is formalized by associating to it a family of configurations representing sets of events which occur during (partial) runs of the system.

**Definition 5.** An (unlabelled) event structure is a structure $(E, \leq, \#)$ consisting of a set $E$ with an order relation $\leq$ and an irreflexive symmetric conflict relation $\# \subseteq E \times E$ such that:

- the set $\{e' \in E/ e' \leq e\}$ is finite, $\forall e \in E$;
- if $e_1 \# e_2$ and $e_2 \leq e_3$ then $e_1 \# e_3$.

A detailed presentation of event structures can be found in [9], as well as in [8].
2. Rule-based event structure

We denote by \( \mathcal{M} \) the set of finite sequences of multisets of rules in \( R \) and by \( \epsilon \in \mathcal{M} \) the empty sequence. The elements of \( \mathcal{M} \) are called histories.

We denote by \( \mathcal{E} \) the set \( \mathcal{M} \times M \); this is the set from which we choose events for our first event structure. Note that the emphasis is placed upon the sequence of multisets of rules, not upon the objects. For an element \( u = (f_1 \cdots f_n, w) \in \mathcal{E} \) we use the following notations: \( \text{length}(u) = n \), \( \text{hist}(u) = f_1 \cdots f_n \) and \( w(u) = w \).

For each \( n \in \mathbb{N} \) we define the event structures \( (E_n^\mathcal{M}, \leq_n, \#^n) \) with \( E_n^\mathcal{M} \subset \mathcal{E} \) for the membrane system \( \Pi \) inductively, such that each \( E_n^\mathcal{M} \) contains only events of length \( n \); the event structure \( E_n^\mathcal{M} \) is defined as the union of the \( E_n^\mathcal{M} \) sets, with the order and conflict relations on \( E_n^\mathcal{M} \) to be defined later.

The idea behind the construction of \( E_n^\mathcal{M} \) is that an event \( (f_1 \cdots f_n, w) \) of length \( n \) marks the application of the multiset of rules \( f_n \) in the \( n \)th evolution step, while recording the multiset of objects \( w \) which remains available for the application of other rules and the complete history \( f_1 \cdots f_{n-1} \) keeps track of the source of the multiset \( w \). The complete history contains exactly the multisets of rules \( f_i \) previously applied in the \( i \)th evolution step. A useful property of the complete history is that our inductive construction ensures that each \( f_i \) is maximally valid with respect to a certain multiset which is the “ancestor” of \( w \) in the \( i \)th evolution step.

While keeping track of the complete history may seem encumbering at first, we will see that it is exactly what allows us to avoid fresh names in the definition of the event structure. To give a short example (see the figure below): consider the case of a system with rules \( r_1 : a \rightarrow b, r_2 : a \rightarrow c, r_3 : b \rightarrow d, r_4 : c \rightarrow d, r_5 : d \rightarrow e \).

\[
\begin{array}{c}
\text{Suppose the starting configuration is } w_0 = \{a\}. \text{ No matter which of the rules } r_1, r_2 \text{ we choose to apply in the first evolution step, after the third evolution step the membrane will contain exactly one object, } e. \text{ The events } u_1 \text{ and } u_2 \text{ marking the application of rule } r_1 \text{ and } r_2, \text{ respectively, should be conflicting, since the two rules cannot be applied together in the same evolution step. If we were to consider an event to be given only by current information available, i.e., the application of a rule, we would have an event } u \text{ depending only on } r_5, e \text{ and possibly } 3, \text{ the number of the evolution step in which it occurs. Then we would have } u \succeq u_1 \text{ and } u \succeq u_2, \text{ since the application of } r_5 \text{ can have as cause either the application of } r_1 \text{ or the application of } r_2. \text{ However, we would also have } u_1 \not\succeq u_2, \text{ which cannot take place in an event structure. Another example is that of a cyclic } P \text{ system, with rules } a \rightarrow b, b \rightarrow a, \text{ in which a similar problem arises [2]. As previously mentioned, solutions for such issues consist mainly in using fresh names to differentiate between events. Our construction shows that it is enough to use the history consisting of previously applied (maximally valid) multisets of rules for the representation of events.}
\end{array}
\]

2.1. Inductive definition

Let \( E_n^0 = \{ (\epsilon, w_0) \} \) with \( \#^0 = \emptyset \) and \( \leq^0 = E_n^0 \times E_n^0 \).

Given \( E_n^k \), for \( k \in \{1, \ldots, n\} \) we consider \( E_n^{k+1} \) to be the smallest set which respects the following conditions:

- For all \( u = (f_1 \cdots f_n, w) \in E_n^k \) maximal elements with respect to \( \leq^k \) we have \( (f_1 \cdots f_n, 0, w + \text{rhs}(f_n)) \in E_n^{k+1} \);

- If \( (f_1 \cdots f_n f_n+1, w) \in E_n^{k+1} \) and \( r \) is a rule such that \( \text{lhs}(r) \leq w \) then \( (f_1 \cdots f_n f_n+1, r, w - \text{lhs}(r)) \in E_n^{k+1} \).

**Proposition 6.** \( E_n^{n+1} = \{ (f_1 \cdots f_n g, v) \mid \exists (f_1 \cdots f_n, w) \in E_n^n \text{ maximal such that } \text{lhs}(g) \leq w + \text{rhs}(f_n), v = w + \text{rhs}(f_n) - \text{lhs}(g) \} \).

**Proof (Sketch).** The set \( S = \{ (f_1 \cdots f_n g, v) \mid \exists (f_1 \cdots f_n, w) \in E_n^n \text{ maximal such that } \text{lhs}(g) \leq w + \text{rhs}(f_n), v = w + \text{rhs}(f_n) - \text{lhs}(g) \} \) verifies the rules of the inductive definition, so \( E_n^{n+1} \subseteq S \). Also, each element of \( S \) is in \( E_n^{n+1} \), by induction on the cardinal of the image of \( g \) (i.e., the number of rules in \( g \), with their multiplicity). Thus \( E_n^{n+1} = S \). \( \square \)

**Definition 7.**

- We define the relation \( \leq^{n+1} \) on \( E_n^{n+1} \) as follows:
  \( (f_1 \cdots f_{n+1}, v) \leq^{n+1} (g_1 \cdots g_{n+1}, w) \iff f_1 \cdots f_n = g_1 \cdots g_n \) and \( f_{n+1} \leq g_{n+1} \)

- The conflict relation \( \#^{n+1} \) on \( E_n^{n+1} \) is given by:
  \( (f_1 \cdots f_{n+1}, v) \#^{n+1} (g_1 \cdots g_{n+1}, w) \) if and only if \( f_1 \cdots f_n \neq g_1 \cdots g_n \) or if \( f_1 \cdots f_n = g_1 \cdots g_n \) and \( \text{lhs}(g_{n+1} - (f_{n+1} \cap g_{n+1})) \leq v \) (for \( n = 0 \) we consider only this second case).
Lemma 8
1. If \((f_1 \cdots f_{n+1}, v), (f_1 \cdots f_{n+1}, w) \in E_{\Pi_1}^{n+1}\) then \(v = w\);
2. If \((f_1 \cdots f_{n+1}, v) \leq_{\Pi_1}^{n+1} (g_1 \cdots g_{n+1}, w)\) then \(v = w + \text{lhs}(g_{n+1} - f_{n+1});\)
3. If \((f_1 \cdots f_n, v), (f_1 \cdots f_n, w) \in E_{\Pi_1}^n\) and \(\text{lhs}(g - (f \cap g)) \leq v\) then \(\text{lhs}(f - (f \cap g)) \leq w\).

Proof. The first statement is proved by induction, using Proposition 6. The second statement follows from the first, using the fact that \((f_1 \cdots f_{n+1}, v) \leq_{\Pi_1}^{n+1} (g_1 \cdots g_{n+1}, w)\).

To prove the third statement, consider \((f_1 \cdots f_n, v')\) and \((f_1 \cdots f_n, w') \in E_{\Pi_1}^n\) as in Proposition 6, maximal in \(E_{\Pi_1}^n\) with respect to \(\leq_{\Pi_1}^n\), such that:
1. \(\text{lhs}(f) \leq v' + \text{rhs}(f_0), v = v' + \text{rhs}(f_0) - \text{lhs}(f);\)
2. \(\text{lhs}(g) \leq w' + \text{rhs}(g_0), w = w' + \text{rhs}(f_0) - \text{lhs}(g).\)

By the first statement, we have \(v' = w'\). Suppose now that \(\text{lhs}(f - (f \cap g)) \leq w\). Since \(w = v' + \text{rhs}(f_0) - \text{lhs}(g)\) we obtain \(\text{lhs}(g - f \cap g) \leq v' + \text{rhs}(f_0) - \text{lhs}(f) = v\), i.e., a contradiction. \(\square\)

Proposition 9. \((E_{\Pi_1}^n, \leq_n, \#^n)\) is an event structure, \(\forall n \in \mathbb{N}\).

Proof. The fact that \(\leq_n\) is an order relation follows from the definition. The finiteness condition follows from the fact that each \(E_{\Pi_1}^n\) is finite.

Clearly, \(#^n\) is irreflexive. It is also symmetric, by Lemma 8.3.

We have that \(e_1, e_2, e_3 \in E_{\Pi_1}^n\) such that \(e_1 \leq_n e_2\) and \(e_1 \#^n e_3\) then \(e_2 \#^n e_3\). Let \(e_i = (f_{i1} \cdots f_{in}, v_i)\) for \(i \in \{1, 2, 3\}\). If there is some \(j \leq n - 1\) such that \(f_{i1} \neq f_{i\#} f_{i3}\) then \(f_{i2} \neq f_{i3}\), thus \(e_2 \#^n e_3\).

If \(f_{i1} = f_{i3}\) for all \(j \leq n - 1\), then \(f_{i2} = f_{i3}\). Suppose that \((e_2, e_3) \notin \#^n\). Then \(\text{lhs}(f_{i3} - f_{i3} \cap f_{i2}) \leq v_2\).

We use the following: if \(f, g, h\) are multisets of rules such that \(f \leq g\) then

\[ h \cap g - h \cap f \leq g - f \]

Therefore \(\text{lhs}(f_{i3} - f_{i3} \cap f_{i1}) = \text{lhs}(f_{i3} - f_{i3} \cap f_{i2}) + \text{lhs}(f_{i3} \cap f_{i2} - f_{i3} \cap f_{i1}) \leq v_2 + \text{lhs}(f_{i3} - f_{i1}) = v_1.\) Since \(f_{i1} = f_{i3}\) for all \(j \leq n - 1\), this contradicts the fact that \(e_1 \#^n e_3\). \(\square\)

Having established an event structure on each \(E_{\Pi_1}^n\), we proceed to construct the larger structure \(E_{\Pi_1}\).

Definition 10. We define \((E_{\Pi_1}, \leq, \#)\) as follows:

1. \(E_{\Pi_1} = \bigcup_{n \in \mathbb{N}} E_{\Pi_1}^n;\)
2. \((e, w_0) \leq x, \forall x \in E_{\Pi_1};\)
3. \((f_1 \cdots f_n, v) \leq (g_1 \cdots g_m, w)\) if and only if \(n \leq m, f_1 = g_1, \ldots, f_{n-1} = g_{n-1}\) and \(f_n \leq g_n;\)
4. \(x \# y\) if and only if one of the following takes place:
   - \(\text{length}(x) = \text{length}(y) = n \text{ and } x \#^n y;\)
   - \(\text{length}(x) = n < \text{length}(y)\) and there is some \(z \in E_{\Pi_1}^n\) such that \(x \#^n z\) and \(z \leq y;\)
   - \(\text{length}(y) = n < \text{length}(x)\) and there is some \(z \in E_{\Pi_1}^n\) such that \(y \#^n z\) and \(z \leq x.\)

Note that the restrictions of \(\leq\) and \(\#\) to \(E_{\Pi_1}^n\) are exactly \(\leq^n\) and \(#^n\). For this reason we sometimes use the non-indexed notations \(\leq\) and \(\#\) even when referring to the relations on \(E_{\Pi_1}^n\), in order to simplify the notation.

Remark 11. Keeping in mind the way \(E_{\Pi_1}^n\) are defined, we note that for any \(x = (f_1 \cdots f_n, v)\) and any \(k \leq n, k \geq 1\) there exists an unique \(y \in E_{\Pi_1}^k\) such that \(\text{hist}(y) = f_1 \cdots f_k\). We denote this element by \(\text{pred}_k(x)\); clearly, \(\text{pred}_k(x) \leq x.\)

Lemma 12. Consider \(x, y \in E_{\Pi_1}\), with \(\text{length}(x) = n \leq \text{length}(y)\). Then:

1. \(x \leq y \Leftrightarrow x \leq \text{pred}_n(y);\)
2. \(x \# y \Leftrightarrow x \# \text{pred}_n(y).\)

Proof. The first property follows from the definition of \(\leq\). For the second one, it suffices to notice that if \(x \# y\) then there exists \(z \in E_{\Pi_1}^n\) such that \(x \# z, z \leq y\), which implies that \(x \# z\) and \(z \leq \text{pred}_n(y)\) in \(E_{\Pi_1}^n\), which is an event structure in itself. \(\square\)

Theorem 13. \((E_{\Pi_1}, \leq, \#)\) is an event structure.

Proof. The properties for the relations \(\leq\) and \(\#\) considered separately follow from their definitions. All we have left to prove is that \(x \leq y, x \# z\) implies \(y \# z\). Let \(n = \text{length}(x), m = \text{length}(y).\) If \(n < m\) then \(x \# \text{pred}_n(z)\); since \(x \leq y\) it follows that \(z \# y.\)

If \(n > m\) then \(z \# \text{pred}_m(x)\) and \(\text{pred}_m(x) \leq x \leq y\) hence \(z \# y.\) \(\square\)
The following result shows the consistency of the event structure $E_{\Pi}$ with the operational semantics of the membrane system. Note that we chose to express this result with respect to both operational semantics only to simplify the notations. Due to the equivalence of the two semantics, we could have expressed Proposition 14 only in terms of the single step semantics.

Proposition 14. $(f_1 \cdots f_n, f, v) \in E_{\Pi}$ if and only if there exist $w_1, \ldots, w_n, w$ multisets over $O$ such that $w_0 \overset{f_1}{\rightarrow} w_1 \cdots \overset{f_m}{\rightarrow} w_n$ and $(w_n, 0) \overset{f_n}{\rightarrow} \cdots \overset{f_1}{\rightarrow} (v, rhs(f))$, where $f = r_1 + \cdots + r_k$.

Proof (Sketch). We prove by induction on $n$ that $(f_1 \cdots f_n, w)$ is maximal in $E_{\Pi}$, with respect to $\leq n$ if and only if there exist $w_1, \ldots, w_n, w$ multisets of objects such that $w_0 \overset{f_1}{\rightarrow} w_1 \cdots \overset{f_m}{\rightarrow} w_n$ and $w_n = w + rhs(f_n)$. The result follows from Proposition 6 and the definition of the single step operational semantics, by induction on $k$, the number of rules (with their multiplicities) in the multiset $f$.

3. Object-based event structure

3.1. Definitions

Let $MO_1$ be the set of multisets over $O$ which have support of cardinality at most 1. In other words, the elements of $MO_1$ are either 0 or of form $a^k$, where $a \in O$ and $k > 0$. Let $\mathcal{F} = MO_1 \times M$; this is the set from which we draw events for our second structure. For an element $u = (a^k, f_1 \cdots f_n) \in \mathcal{F}$ we denote $\text{length}(u) = n$, $\text{obj}(u) = a$ and $w(u) = a^k$. For a subset $U$ of $\mathcal{F}$ we denote by $w(U)$ the multiset of objects $\sum_{u \in U} w(u)$. For a multiset of rules $f$, we denote by $f^a$ the multiset of rules given by: $f^a(r) = f(r)$ if $a \in rhs(r)$ and $f^a(r) = 0$ if $a \not\in rhs(r)$.

Due to the simpler nature of the object-based events, we can define the conflict relation over all the elements of $\mathcal{F}$ (this is not the case in the rule-based event structure).

Definition 15. The relation $\#$ over $\mathcal{F}$ is defined by:

$$(a^k, f_1 \cdots f_n) \# (b^m, g_1 \cdots g_m)$$

whenever $f_1 \cdots f_s \neq g_1 \cdots g_s$, where $s = \min(n, m)$.

This definition is possible by the fact that we no longer have events describing intermediary small steps which take place “inside” an evolution step.

Definition 16. Let $F$ be a subset of $\mathcal{F}$. A subset $U$ of $F$ is called a state in $F$ if $U$ is conflict-free and is maximal with this property (with respect to inclusion).

In other words, $U$ is a state in $F$ if and only if $U \times U \cap \# = \emptyset$ and $\forall x \in U, y \in F, (x, y) \not\in \#$ implies $y \in U$.

The event structure $(F_{\Pi}, \leq, \#)$ built in this section is defined in a similar manner to that of Section 2. We start by defining inductively the sets $F_{\Pi}^n$ which contain exactly the events of length $n$.

Definition 17. Let $F_{\Pi}^0 = \{(a^k, \epsilon)/a \in w_0, k = w_0(a)\}$. Given $F_{\Pi}^k, k \in \{1, \ldots, n\}$ we define $F_{\Pi}^{n+1}$ as follows. For any state $U$ in $F_{\Pi}^n$ and any multiset of rules $f$ maximally valid in $w(U)$ we denote by $u(U, f, a)$ the object $(a^k, f_1 \cdots f_n, f)$ given by:

- If $(w(U) - lhs(f) + rhs(f))(a) = 0$ then $u(U, f, a)$ is not defined.
- If $(w(U) - lhs(f) + rhs(f))(a) > 0$ then $k = (w(U) - lhs(f) + rhs(f))(a)$.
- $f_1 \cdots f_n = \text{hist}(u)$ for some $u \in U$. 


We define $F_{n+1}$ as the set of all $u(U,f,a)$ which are defined according to the rules above, i.e., for all $U$ state in $F_n$, $f$ maximally valid in $w(U)$ and $a$ such that $(w(U) - lhs(f) + rhs(f))(a) > 0$.

Note that since $U$ is conflict-free and all the events in $U$ have the same length, it follows that all the events in $U$ have the same history.

Let $F_n = \bigcup_{n \in \mathbb{N}} F_{n+1}$. The conflict structure on $F_n$ is the one inherited from $F$. To give the order relation $\leq$ on $F_n$, we start from a relation between events whose length differs by 1. Namely, we define $\prec$ as the transitive and reflexive closure of $\leq$.

**Definition 18.** For $x \in F_{n+1}$, $y \in F_{n+1}$ we set $x \prec y$ whenever there exist $U,f,a$ such that $y = u(U,f,a), x \in U$ and $obj(x) \in rhs(f^n)$ or $obj(x) = a$ and $a \in w(U) - lhs(f)$.

The relation $\leq$ is the one giving the complexity of the structure $(F_n, \leq, \#)$ because the conflict relation gives only the information about what objects were produced together as part of an evolution step. The order relation gives the causal information, meaning that if $(a^k,f_1 \cdots f_n) \leq (b^m,g_1 \cdots g_m)$, then the objects $a$ actively took part in making the objects $b$ appearing. In more detail, some objects $a$ were either consumed by some rules which produced objects $b$ or $a = b$ and some objects $a$ were not consumed by any rule, thus remaining in the membrane. The order relation $\leq$ provides a link between occurrences of $a$ and $b$, with respect to the quantities in which they appear.

**Example 19.** Consider the membrane system with rules $r_1 : a \rightarrow bc$, $r_2 : a^2 \rightarrow ac$, $r_3 : ab \rightarrow cd$ and with the initial multiset of objects $a^2b$. The event structure obtained is:

![Event Structure Diagram]

where the rectangles indicate states and the arrows indicate the relation $\prec$.

We show now that $(F_n, \leq, \#)$ is an event structure.

**Remark 20.** If $x = (a^k,f_1 \cdots f_n)$ then either $x = y$ or $x \neq y$. $n < m$ and $f_1 \cdots f_n = g_1 \cdots g_n$.

It follows from Remark 20 that $\leq$ is an order relation on $F_n$.

**Proposition 21.** $(F_n, \leq, \#)$ is an event structure.

**Proof.** If $x \leq y, y \neq z$, let $hist(x) = f_1 \cdots f_n$, $hist(y) = g_1 \cdots g_m$ and $hist(z) = h_1 \cdots h_k$. If $x = y$ then clearly $y \neq z$. If not, we have $n < m$. Let $s = \min(n,k)$, then $f_1 \cdots f_s = g_1 \cdots g_s \neq h_1 \cdots h_s$ and $s = \min(m,k)$. Thus $y \neq z$. □

The following result shows the consistency of the event structure $F_n$ with the maximally parallel step operational semantics of the membrane system.

**Proposition 22.** $(a^k,f_1 \cdots f_n) \in F_n$ if and only if there exist $w_1, \ldots, w_n$ multisets over $O$ such that $w_0 \xrightarrow{f_1} w_1 \cdots \xrightarrow{f_n} w_n$ and $k = w_0(a) > 0$.

**Proof.** Proof by induction over $n$, involving the maximal validity of each $f_i$ with respect to $w_{i-1}, i \in \{1, \ldots, n\}$. □

### 3.2. Causality

In this part of the paper we argue that the order relation of the second event structure, $F_n$, is better suited than the order on $E_f$ to describe causality in a membrane system.
The order on $E_{\Pi}$ is intuitively given by the order of rule applications in different multisets. To elaborate, consider two rules $r, s$ with the property that whenever $(f_1, \ldots, f_n, s, v) \in E_{\Pi}$ there exists $i \in \{1, \ldots, n\}$ such that $r \in f_i$. In other words, $(f_1, \ldots, f_n, s, v) \in E_{\Pi}$ always implies that $(f_1, \ldots, f_{i-1}, r, w) \in E_{\Pi}$ for some $w$, and necessarily that $(f_1, \ldots, f_{i-1}, r, w) \leqslant (f_1, \ldots, f_n, s, v)$.

We look at the following example.

**Example 23.** Consider a system with rules $r_1 : a \rightarrow c$, $r_2 : b \rightarrow d$, $r_3 : d \rightarrow e$ and initial multiset of objects $ab$. Rules $r_1, r_3$ have the property from above, yet we cannot say that rule $r_1$ causes rule $r_3$. The fact that rule $r_1$ is always applied in a multiset previous to the one in which rule $r_3$ is applied, is conditioned only by our choice of initial multiset and offers no pertinent causality condition. On the other hand, it is natural to say that applying rule $r_2$ may cause rule $r_3$ to be applied. We define a notion of dependence on rules to describe this situation and present the relation with the order on $F_{\Pi}$.

**Definition 24.** Let $r, s$ be two rules of the membrane system $\Pi$, not necessarily distinct. We denote by $r \prec s$ the relation given by $\text{obj} (\text{lhs}(r)) \cap \text{obj} (\text{lhs}(s)) \neq \emptyset$ and by $r \triangleleft s$ the transitive and reflexive closure of $\prec$. We call $\triangleleft$ the dependence relation over the rules of the membrane system.

**Theorem 25.** If $(a^k, f_1 \cdots f_m) \preceq (b^q, f_1 \cdots f_n)$ and $a \neq b$, then there exist rules $r, s$ such that $r \triangleleft s$ and $a \in \text{lhs}(r), b \in \text{rhs}(s)$.

**Proof.** The proof is obtained by noticing that if $(c^l, g_1 \cdots g_m) \prec (d^q, g_1 \cdots g_{m+1})$ then either there exists a rule $r \in g_{m+1}$ such that $c \in \text{lhs}(r), d \in \text{rhs}(r)$ or $c = d$.

We have $m < n$ because $a \neq b$ thus $(a^k, f_1 \cdots f_m) \prec (b^q, f_1 \cdots f_n)$. Consider the case $m < n - 1$. Since $(a^k, f_1 \cdots f_m) \preceq (b^q, f_1 \cdots f_n)$ there exist $x_{m+1}, \ldots, x_{n-1} \in F_{\Pi}$ such that

\[ (a^k, f_1 \cdots f_m) \prec x_{m+1} \cdots x_{n-1} \prec (b^q, f_1 \cdots f_n) \]

Let $c_i = \text{obj}(x_i), i \in \{m + 1, \ldots, n - 1\}$. We prove our statement by induction over the number $n - m - 1$ of elements $x_i$.

If $n - m - 1 = 0$ then the result is obtained immediately for some $r = s \in F_{\Pi}$. If $n - m - 1 > 0$, from $(a^k, f_1 \cdots f_m) \preceq x_{m+1}$ we obtain that either $\exists r_{m+1} \in f_{m+1}$ such that $a \in \text{lhs}(r_{m+1}), c_{m+1} \in \text{rhs}(r_{m+1})$ or $a = c_{m+1}$.

Since $x_{m+1} \cdots x_{n-1} \prec (b^q, f_1 \cdots f_n)$ we know by the inductive hypothesis that there exist $t, s$ rules such that $t \triangleleft s$ and $c_{m+1} \prec x_{n-1} \prec (b^q, f_1 \cdots f_n)$. If there exists $r_{m+1} \in f_{m+1}$ such that $a \in \text{lhs}(r_{m+1}), c_{m+1} \in \text{rhs}(r_{m+1})$ then we set $r = r_{m+1}$ and we have $r \prec t \triangleleft s$ and $a \in \text{lhs}(r), b \in \text{rhs}(s)$. If $a = c_{m+1}$ then we set $r = t$. \(\square\)

Theorem 25 shows that the causality given by the order relation on $F_{\Pi}$ is closely related to rule dependence. While we have chosen not to present in detail the case of $a = b$, we note that a similar (if slightly more complicated) statement can be given for it, by also adding the possibility that some objects $a = b$ can be left unconsumed by any rule at each step of the evolution of the system.

**4. Conclusion and related work**

In this paper we presented two event structures for basic membrane systems, a rule-based and an object-based one. While the rule-based event structure seems more natural (and similar ideas have been described in a previous paper [2]), its order relation does not distinguish between more subtle differences in rule causality (see Example 23). The object-based event structure is a novel approach defined in this paper; its order relation offers sufficient causal information to be directly linked to a natural dependence relation over the rules of a membrane system. This structure also seems more appropriate for quantitative analysis from a biological point of view.

Both structures presented in this paper use computational histories to keep track of similar yet different events (such as those that can appear in a cyclic computation, for example). In contrast with the previous approaches, neither structure uses fresh names to distinguish between such events.

The paper [2] constructs an event structure similar to the rule-based one, starting by viewing membrane contents as strings. When membrane contents are viewed as multisets, the event structure constructed in [2] is equivalent to a structure $E_{\Pi}^n$. However, the paper lacks a rigorous construction for the entire evolution of a membrane system, and it only makes the observation that the fresh names used to “chain” the event structures employed have to depend on previous evolution steps.

The paper [1] mentions rules being causally related by sharing objects, but without offering a formal approach. Instead, it focuses on giving a semantics for the membrane system in which causal information is defined by the labels of a transition system, depending heavily on the use of fresh names.

The paper [5] discusses causality in terms of two Petri net representations of a membrane system. The notion of causality is rule-based and is obtained by unfolding each Petri net representation. However, these notions are not explicitly related to the initial membrane system.

The paper [7] proposes an event based view of membrane evolution by translating membrane systems into zero safe nets and associating an event automaton to the resulting nets. However, as the authors mention, "the main drawback of event automata is that dependencies among events are not represented explicitly". Thus the paper does not discuss causality in the proper sense of event structures.
Acknowledgements

We thank the anonymous reviewers for their helpful comments.
This research work is partially supported by CNCSIS grant IDEI 402/2007.

References