What is an Event for Membrane Systems?

Gabriel Ciobanu[1,2] Dorel Lucanu[1]

[1]"A.I.Cuza" University of Iași, Faculty of Computer Science
[2]Romanian Academy, Institute of Computer Science, Iași
{gabriel, dlucanu}@info.uaic.ro

WMC8

Outline

Event Structures

Definition (Event Structure)

An event structures is a triple $(E, \leq, \#)$ where

- E is a set of *events*,
- $\leq \subseteq E \times E$ is a partial order, the *causality relation*,
- $\# \subseteq E \times E$ is an irreflexive and symmetric relation, the *conflict relation*,

  satisfying the *principle of conflict heredity*:

$$\forall e_1, e_2, e_3 \in E.\ e_1 \leq e_2 \wedge e_1 \# e_3 \Rightarrow e_2 \# e_3.$$

Event Structures

Event Structure Associated to a Labelled Transition System

Let $(S, \longrightarrow, L, s_0)$ be a transition system, where S is the set o states, $\longrightarrow$ is the transition relation consisting of triples $(s, \ell, s') \in S \times L \times S$, often written as $s \xrightarrow{\ell} s'$, L is the set of labels (actions), and s_0 is the initial state. A (sequential) *computation* is a sequence $s_0 \xrightarrow{\ell_1} s_1 \ldots \xrightarrow{\ell_n} s_n$ such that $(s_{i-1}, \ell_i, s_i) \in \longrightarrow$.

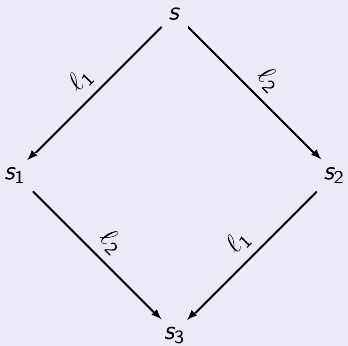
Definition (Events in a lts)

Let $\sim$ be the smallest equivalence satisfying: if (s, ℓ_1, s_1), (s, ℓ_2, s_2), (s_1, ℓ_2, s_3), (s_2, ℓ_1, s_3), $\in \longrightarrow$ and $(\ell_1 \neq \ell_2$ or $s_1 \neq s_2)$, then
$(s, \ell_1, s_1) \sim (s_2, \ell_1, s_3)$.
An *event* is a $\sim$-equivalence class written as $[s, \ell, s']$.

Event Structures

Event Structure Associated to a Labelled Transition System

Intuitively, two transitions are equivalent iff they are occurrences of the same event. The relation $\sim$ can be easier understood from the following picture:



We have two events which may occur in any order, i.e., the two events are concurrent.

Event Structures

Event Structure Associated to a Labelled Transition System

Definition (Configuration in a lts)

A *configuration* is a multiset of events $[s_{i-1}, \ell_i, s_i]$ corresponding to a computation $s_0 \xrightarrow{\ell_1} s_1 \ldots \xrightarrow{\ell_n} s_n$.

Theorem (Event Structure of a lts)

$(S, \rightarrow, L, s_0)$ *can be organized as an event structure.*

Sketch.

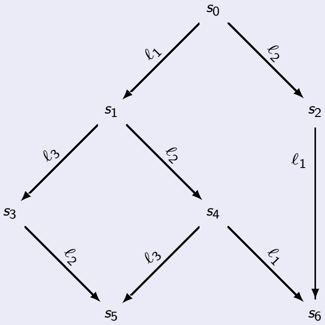
The event structure $(E, <, \#)$ *is defined by:*

- E *is the set of events as defined;*
- $e_1 < e_2$ *if every configuration which contains* e_2 *also contains* e_1*;*
- $e_1 \# e_2$ *if there is no configuration containing both* e_1 *and* e_2*.*

□

Event Structures

Example



The events defined by the lts described above are:

$$E_1 = \{(s_0, \ell_1, s_1)\} \qquad E_4 = \{(s_1, \ell_2, s_4), (s_3, \ell_2, s_5)\}$$
$$E_2 = \{(s_0, \ell_2, s_2)\} \qquad E_5 = \{(s_2, \ell_1, s_6)\}$$
$$E_3 = \{(s_1, \ell_3, s_3), (s_4, \ell_3, s_5)\} \qquad E_6 = \{(s_4, \ell_1, s_6)\}$$

Event Structures

Example

A configuration describe a (partial) computation expressed in terms of events. This lts defines the following configurations:

E_1	$E_2\ E_5$
E_2	$E_1\ E_3\ E_4$
$E_1\ E_3$	$E_1\ E_4\ E_3$
$E_1\ E_4$	$E_1\ E_4\ E_6$

We have $E_1 < E_3$ because any configuration containing E_3 also contains E_1. Since any occurrence of E_3 is always after an occurrence of E_1, it follows that there is causal relationship between the two events. We get $E_1 < E_4 < E_6$ and $E_2 < E_5$ in a similar way.
Since there is no a configuration containing both E_1 and E_2, it follows that there is a conflict between the two events, i.e., $E_1 \# E_2$. We get $E_1 \# E_5$, $E_2 \# E_3$, $E_2 \# E_4$, $E_2 \# E_6$, $E_5 \# E_3$, $E_5 \# E_4$, and $E_5 \# E_6$ in a similar way.

Event Structure of an Evolution Step

Non-commutative Case

A context is a string of the form $w \bullet w'$, where w, w' are strings of objects, and $\bullet$ is a special symbol. Each rewriting step $wuw' \to wvw'$ is uniquely determined by the context $w \bullet w'$ and the rule $\ell : u \to v$. Therefore the transition $(wuw', \ell, wvw') = wuw' \xrightarrow{\ell} wvw'$ is denoted by $(w \bullet w', \ell)$.

The maximal parallel rewriting over strings is defined as follows: $w \overset{mpr}{\Rightarrow} w'$ if and only if there are $\ell_1, \ldots, \ell_n$, $\ell_i : u_i \to v_i$ $(i = \overline{1, n})$ such that $w = w_0 u_1 w_1 \ldots u_n w_n$, $w' = w_0 v_1 w_1 \ldots v_n w_n$ and $w_0 w_1 \ldots w_n$ irreducible (no rule can be applied).

Event Structure of an Evolution Step

Non-commutative Case

Definition

The labelled transition system associated to $w \stackrel{mpr}{\Rightarrow} w'$ is given by all the sequential rewritings starting from w and ending in w'. The event structure $ES(w, w')$ associated to $w \stackrel{mpr}{\Rightarrow} w'$ is the event structure associated to its labelled transition system.

Theorem

The event structure $ES(w, w') = (E, <, \#)$ associated to $w \stackrel{mpr}{\Rightarrow} w'$ consists of only independent events, i.e., $< = \emptyset$ and $\# = \emptyset$.

Proof.

We have $w \stackrel{mpr}{\Rightarrow} w'$ iff $w = w_0 u_1 w_1 \dots u_n w_n$, $w = w_1 v_1 w_2 \dots v_n w_n$, $\ell_i : u_i \rightarrow v_i$ is an evolution rule, for $i = 1, \dots, n$, and $w_1 \dots w_0 w_1 \dots w_n$ is irreducible. The conclusion of the theorem follows by the fact that $[w_0 \dots \bullet w_i \dots w_n, \ell_i]$ is a configuration (any of events can occur first). □

Event Structure of an Evolution Step

Non-commutative Case

Example

Let us consider a simple membrane consisting of the following three rules

$$\ell_1 : a \longrightarrow b$$
$$\ell_2 : b \longrightarrow a$$
$$\ell_3 : ab \longrightarrow d$$

and having the content $aabc$. We investigate the space of all sequential rewritings corresponding to the application of rules in the evolution step $aabc \overset{mpr}{\Rightarrow} bbac$ in order to discover the events of this step.

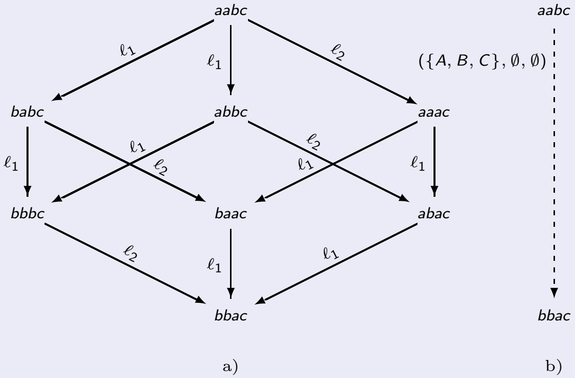
We have the following three independent events:

$$A = \{(\bullet abc, \ell_1), (\bullet bbc, \ell_1), (\bullet bac, \ell_1), , (\bullet aac, \ell_1)\}$$
$$B = \{(a \bullet bc, \ell_1), (b \bullet bc, \ell_1), (b \bullet ac, \ell_1), , (a \bullet ac, \ell_1)\}$$
$$C = \{(aa \bullet c, \ell_2), (ab \bullet c, \ell_2), (bb \bullet c, \ell_2), , (ba \bullet c, \ell_2)\}.$$

Event Structure of an Evolution Step

Non-commutative Case

Example



Lts corresponding to $aabc \stackrel{mpr}{\Rightarrow} bbac$

Event Structure of an Evolution Step

Commutative Case

A context is a string of the form $[\bullet w]$, where w is a multiset of objects, and $\bullet$ is a special symbol. It is easy to see now that the position of $\bullet$ in a context is not important, and therefore we write $\bullet$ at the beginning. Each rewriting step $[uw] \longrightarrow [vw]$ is uniquely determined by the context $[\bullet w]$ and the rule $\ell : u \longrightarrow v$. Therefore the transition $([uw], \ell, [vw]) = [uw] \xrightarrow{\ell} [vw]$ is denoted by $([\bullet w], \ell)$.

The maximal parallel rewriting over multisets is defined as follows: $[w] \overset{mpr}{\Rightarrow} [w']$ iff there are $\ell_1, \ldots, \ell_n$, $\ell_i : u_i \longrightarrow v_i$ $(i = \overline{1, n})$ such that $[w] = [u_1 \ldots u_n r]$, $[w'] = [v_1 \ldots v_n r$ and r is irreducible (no rule can be applied).

Event Structure of an Evolution Step

Commutative Case

Definition

The labelled transition system associated to $[w] \stackrel{mpr}{\Rightarrow} [w']$ is given by all the sequential rewritings starting from $[w]$ and ending in $[w']$.

Theorem

The event structure $(E, <, \#)$ associated to the labelled transition system defined by $[w] \stackrel{mpr}{\Rightarrow} [w']$ has the following properties:

- $[[\bullet w_1], \ell_1] < [[\bullet w_2], \ell_2]$ *if and only if* $\ell_1 = \ell_2$, $[[\bullet w_1], \ell_1]$ *corresponds to the i-th application of the rule* ℓ_1, $[[\bullet w_2], \ell_1]$ *corresponds to the j-th application of the rule* ℓ_1, *and* $i < j$;
- $\# = \emptyset$.

Definition

The event structure associated to $[w] \stackrel{mpr}{\Rightarrow} [w']$ is $ES([w], [w']) = (E, \emptyset, \emptyset)$, where $(E, <, \emptyset)$ is the event structure associated to the labelled transition system defined by $[w] \stackrel{mpr}{\Rightarrow} [w']$.

Event Structure of an Evolution Step

Commutative Case

Example

Let us consider a simple membrane consisting of the following three rules

$$\ell_1 : a \rightarrow b$$

$$\ell_2 : b \rightarrow a$$

$$\ell_3 : ab \rightarrow d$$

and having the content *aabc*. We investigate the space of all sequential rewritings corresponding to the application of rules in the evolution step $aabc \stackrel{mpr}{\Rightarrow} bbac$ in order to discover the events of this step.

We also have three events, but they are not totally independent:

$$\overline{A} = \{([\bullet abc], \ell_1), ([\bullet aac], \ell_1)\}$$

$$\overline{B} = \{([\bullet bbc], \ell_1), ([\bullet bac], \ell_1)\}$$

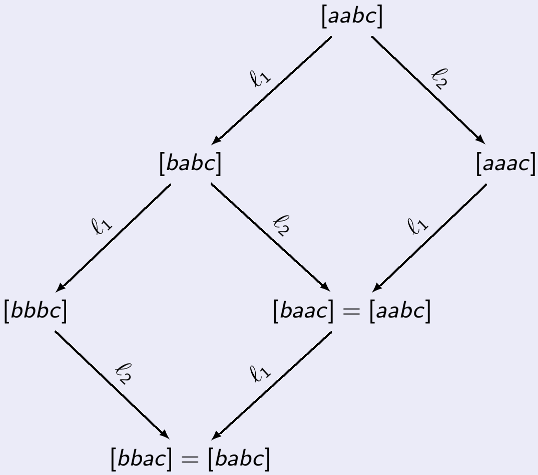
$$\overline{C} = \{([\bullet abc], \ell_2), ([\bullet bac], \ell_2), ([\bullet bbc]\ell_2)\}$$

$$\overline{A} < \overline{B}$$

Event Structure of an Evolution Step

Commutative Case

Example



Event Structure of a Membrane

We first note that the notation for events is not longer suitable for the case of membranes. We assume that we have a membrane with two evolution rules: $\ell : a \to b$ and $\ell' : b \to a$. Let us consider the computation $aa \stackrel{mpr}{\Rightarrow} bb \stackrel{mpr}{\Rightarrow} aa \stackrel{mpr}{\Rightarrow} bb$. Since the events of $aa \stackrel{mpr}{\Rightarrow} bb$ occur always before the events of $bb \stackrel{mpr}{\Rightarrow} aa$, and the events of $bb \stackrel{mpr}{\Rightarrow} aa$ occur before the events of $aa \stackrel{mpr}{\Rightarrow} bb$, then we get $[a\bullet, \ell] < [b\bullet, \ell'] < [a\bullet, \ell]$, i.e., the causality relation $<$ is cyclic. Therefore each event $[c, \ell]$ in $ES(w, w')$ is denoted with a new fresh name e, and we define $action(e) = [c, \ell]$. In this way, the event sets corresponding to different computation steps are disjoint.

Event Structure of a Membrane

Non-commutative Case

The event structure $ES(M) = (E_M, <_M, \#_M)$ associated to a membrane M is defined as follows:

1. for each reachable w and each w' such that $w \stackrel{mpr}{\Rightarrow} w'$, $ES(w, w') \subseteq ES(M)$ (we recall that the events in $ES(w, w')$ are renamed);
2. for each reachable w and each ready-to-fire permutation w' of w, $w \neq w'$, we consider in E_M a distinct event e with $action(e)$ equal to $w =_c w'$, and
   1. for each reachable w_1 such that $w_1 \stackrel{mpr}{\Rightarrow} w$ and for each event e_1 in $ES(w_1, w)$ we have $e_1 < e$, and
   2. for each w_2 such that $w' \stackrel{mpr}{\Rightarrow} w_2$ and for each event e_2 in $ES(w', w_2)$ we have $e < e_2$;
3. if $e_1, e_2 \in E_M$ such that $action(e_1)$ is $w =_c w_1$ and $action(e_2)$ is $w =_c w_2$ with $w_1 \neq w_2$, then we have $e_1 \# e_2$;
4. if $w \stackrel{mpr}{\Rightarrow} w_1$, $w \stackrel{mpr}{\Rightarrow} w_2$ and $w_1 \neq w_2$, then we have $e_1 \# e_2$ in $ES(M)$ for each e_1 in $ES(w, w_1)$ and e_2 in $ES(w, w_2)$.

Event Structure of a Membrane

Non-commutative Case

Example

Let us consider a simple membrane consisting of the following three rules

$$\ell_1 : a \rightarrow b \qquad \ell_2 : b \rightarrow a \qquad \ell_3 : ab \rightarrow d$$

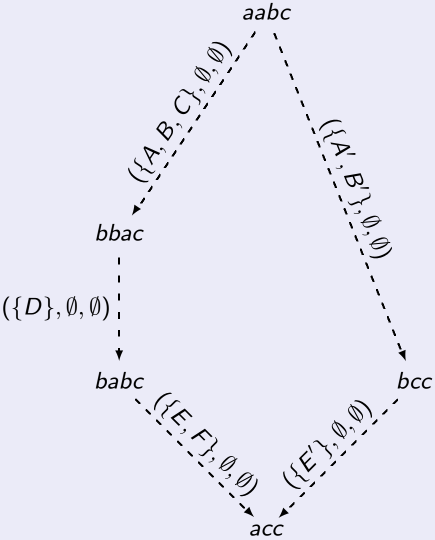
and having the content $aabc$. We investigate the space of all sequential rewritings corresponding to the application of rules in the evolution step $aabc \stackrel{mpr}{\Rightarrow} bbac$ in order to discover the events of this step.

Let M be the membrane presented in the above example. We consider the computation subspace given by $aabc \stackrel{mpr}{\Rightarrow} bbac =_c babc \stackrel{mpr}{\Rightarrow} acc$ and $aabc \stackrel{mpr}{\Rightarrow} bcc \stackrel{mpr}{\Rightarrow} acc$. We denote by D the event corresponding to $bbac =_c babc$, by $(\{E, F\}, \emptyset, \emptyset)$ the event structure corresponding to $babc \stackrel{mpr}{\Rightarrow} acc$, by $(\{A', B'\}, \emptyset, \emptyset)$ the event structure corresponding to $aabc \stackrel{mpr}{\Rightarrow} bcc$, and by $(\{E'\}, \emptyset, \emptyset)$ the event structure corresponding to $bcc \stackrel{mpr}{\Rightarrow} acc$.

Event Structure of a Membrane

Non-commutative Case

Example



Event Structure of a Membrane

Commutative Case

The construction of the event structure $\overline{ES}(M) = (E_M, <_M, \#_M)$ associated to a membrane M is simpler than that for the case of strings:

1. for each reachable $[w]$ and each $[w']$ such that $[w] \stackrel{mpr}{\Rightarrow} [w']$, $ES([w],[w']) \subseteq \overline{ES}(M)$;
2. if $[w] \stackrel{mpr}{\Rightarrow} [w_1]$, $[w] \stackrel{mpr}{\Rightarrow} [w_2]$ and $[w_1] \neq [w_2]$, then we have $e_1 \# e_2$ in $ES(M)$ for each e_1 in $ES([w],[w_1])$ and e_2 in $ES([w],[w_2])$;
3. if $[w] \stackrel{mpr}{\Rightarrow} [w_1]$, $[w_1] \stackrel{mpr}{\Rightarrow} [w_2]$, then we have $e_1 < e_2$ in $ES(M)$ for each e_1 in $ES([w],[w_1])$ and e_2 in $ES([w_1],[w_2])$.

Event Structure of a Membrane

Commutative Case

Example

Let us consider a simple membrane consisting of the following three rules

$$\ell_1 : a \rightarrow b \qquad \ell_2 : b \rightarrow a \qquad \ell_3 : ab \rightarrow d$$

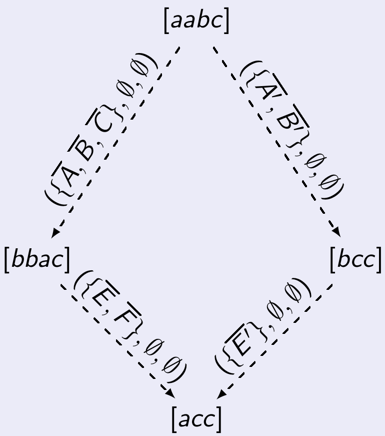
and having the content $aabc$. We investigate the space of all sequential rewritings corresponding to the application of rules in the evolution step $aabc \stackrel{mpr}{\Rightarrow} bbac$ in order to discover the events of this step.

We consider the computation subspace given by $[aabc] \stackrel{mpr}{\Rightarrow} [bbac] \stackrel{mpr}{\Rightarrow} [acc]$ and $[aabc] \stackrel{mpr}{\Rightarrow} [bcc] \stackrel{mpr}{\Rightarrow} [acc]$. We denote by $(\{\overline{E}, \overline{F}\}, \emptyset, \emptyset)$ the event structure corresponding to $[babc] \stackrel{mpr}{\Rightarrow} [acc]$, by $(\{\overline{A'}, \overline{B'}\}, \emptyset, \emptyset)$ the event structure corresponding to $[aabc] \stackrel{mpr}{\Rightarrow} [bcc]$, and by $(\{\overline{E'}\}, \emptyset, \emptyset)$ the event structure corresponding to $[bcc] \stackrel{mpr}{\Rightarrow} [acc]$.

Event Structure of a Membrane

Commutative Case

Example



Conclusion

In this paper we study the event structure for membrane systems, considering both the causality and the conflict relations. We investigate how an event structures can be associated to an parallel evolution step. We found that the meaning of an event depends on the algebraic structure used for the contents of membranes: string or multiset.

The construction of the event structures for the cases of priorities and promoters can be reduced to the case of maximal parallel rewriting. A computation step $w \stackrel{pri}{\Rightarrow} w'$ ($[w] \stackrel{pri}{\Rightarrow} [w']$) in the presence of priorities is the same with $w \stackrel{mpr}{\Rightarrow} w'$ ($[w] \stackrel{mpr}{\Rightarrow} [w']$) by taking into account only the rules of maximal priority applicable on w. Similarly, a computation step $w \stackrel{prom}{\Rightarrow} w'$ ($[w] \stackrel{prom}{\Rightarrow} [w']$) in the presence of promoters is the same with $w \stackrel{mpr}{\Rightarrow} w'$ ($[w] \stackrel{mpr}{\Rightarrow} [w']$) where the rules requiring promoters not in w are not considered.