STATIC ANALYSIS OF USEFULNESS STATES IN TRANSITION P SYSTEMS

Juan Alberto Frutos, Luis Fernandez, Fernando Arroyo, Gines Bravo

Abstract: Transition P Systems are a parallel and distributed computational model based on the notion of the cellular membrane structure. Each membrane determines a region that encloses a multiset of objects and evolution rules. Transition P Systems evolve through transitions between two consecutive configurations that are determined by the membrane structure and multisets present inside membranes. Moreover, transitions between two consecutive configurations are provided by an exhaustive non-deterministic and parallel application of evolution rules. But, to establish the rules to be applied, it is required the previous calculation of useful, applicable and active rules. Hence, computation of useful evolution rules is critical for the whole evolution process efficiency, because it is performed in parallel inside each membrane in every evolution step. This work defines usefulness states through an exhaustive analysis of the P system for every membrane and for every possible configuration of the membrane structure during the computation. Moreover, this analysis can be done in a static way; therefore membranes only have to check their usefulness states to obtain their set of useful rules during execution.

Keywords: Evolution Rules, Usefulness States, Transition P System, Sequential Machines, Static Analysis

ACM Classification Keywords: F.1.1 Computation by abstract devices – Models of computation. D.1.m Miscellaneous – Natural Computing

Introduction

Membrane Computing was introduced by Gh. Păun in [Păun, 1998], as a new branch of natural computing, inspired on living cells. Membrane systems establish a formal framework in which a simplified model of cells is considered a computational device. Starting from a basic model, Transition P systems, many different variant have been considered; and many of them have been demonstrated to be, in power, equivalent to the Turing Machine. An overview of this model is described in the next section.

Nowadays, a challenge for researchers of these kinds of devices is to get real implementations of membrane systems with a high degree of parallelism. Accordingly with this fact, there are some published works related to parallel implementation of membrane systems [Ciobanu, 2004], [Syropoulos, 2003] and [Tejedor, 2007].

In [Tejedor, 2007] set up two different phases in the inner dynamic of the evolution step: first phase is related to inner application of evolution rules inside membranes; second phase is related to communication among membranes in the systems. Then it is computed the total time the system spend during the evolution step, and what is important to note is the fact that reducing the time membranes spend in the application phase, the system gets an important gain in the total time it needs for the evolution step. The work presents in this paper is to improve the first phase –application of evolution rules inside membranes- getting useful rules in a faster way. In order to do it, it is introduced the concept of *usefulness states* of membranes in Transition P systems. The main idea is to carry out a static analysis of the P system in order to obtain all usefulness states and transitions between states in each membrane. During execution, membranes will obtain the set of useful evolution rules directly from their usefulness states.

This paper is structures as follows: first Transition P systems are formally defined. Second, usefulness states associated to membranes of Transition P systems with rules able to dissolve membranes are established. Third, the inhibition capability in P systems is incorporated. Fourth, a way for encoding usefulness states is introduced in order to reduce the needed space for implementing. Finally, conclusions are presented.

Transition P Systems

Formally, a transition P system of degree m is a construct of the form

$$\Pi = (O, \mu, \omega_1, \ldots, \omega_m, (R_1, \rho_1), \ldots, (R_m, \rho_m), i_0)\text{, where:}$$

- O is the alphabet of objects
- μ is a membrane structure, consisting of m membranes, labelled with *1,2,...., m*. It is a hierarchically arranged set of membranes, contained in a distinguished external membrane, called skin membrane.

Several membranes can be placed inside a parent membrane; and finally, a membrane without any other membrane inside is said to be elementary.

- $\omega_i \mid 1 <= i <= m$ are strings over O, representing multisets of objects placed inside the membrane with label *i*.
- $R_i \mid 1 <= i <= m$ are finite sets of evolution rules associated to the membrane with label *i*. Rules have the form $u \to v$, $u \to v\ \delta$ or $u \to v\ \tau$, with $u \in O^+$ and $v \in (O^+ \times TAR)^*$, where $TAR=\{here, out\} \cup \{in_j \mid 1 <= i <= m\}$. Symbol δ represents membrane dissolution, while symbol τ represents membrane inhibition. ρ_i, $1 <= i <= m$, are priority relations defined over R_i, the set of rules of membrane *i*.
- i_0 represents the label of the membrane considered as output membrane.

The initial configuration of a P system is given by specifying the membrane structure and the multisets of objects placed inside membranes. $C = (\mu, \omega_1, \ldots, \omega_m)$. A transition takes place by application of evolution rules inside each membrane in the system, in a non-deterministic and maximally parallel manner. This implies that every object in the system able to evolve by the application of one evolution rule must evolve and rules are applied in a non-deterministic way. A computation is defined as a sequence of transitions between system configurations in which the final configuration has no objects able to evolve at any membrane of the system.

Figure 1 shows an example of transition P system, although only multiset and rules associated to membrane 1 are represented.

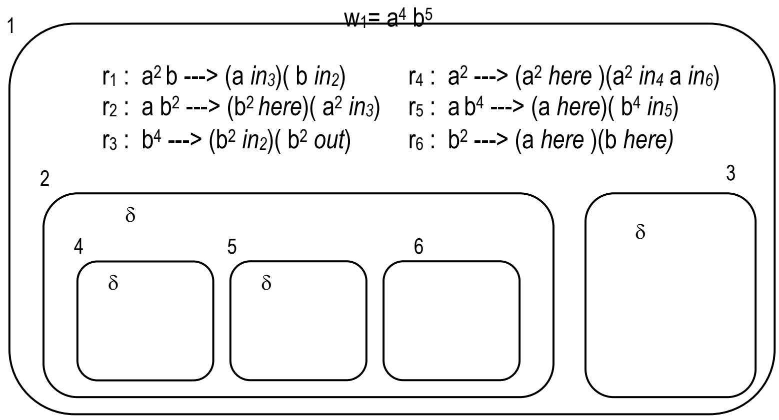

Figure 1: Transition P System

Usefulness states in transition P systems with membrane dissolving capability

Transition P systems with membrane dissolving capability are characterized by $OP_m(\alpha, tar, \delta, pri)$. This notation denotes the class of P systems with simple objects, priorities and dissolving capability. In this class of P systems, rules have the capability for dissolving membranes in the systems and, hence, they can modify the membrane structure of the P system during execution. Evolution rules in these systems are of the form $u \to v \quad or \quad u \to v\delta$. In a different way, $r = (u, v, \xi), where\ \xi \in \{\delta, \lambda\}$.

Evolution rules able to be applied at any evolution step of the P system must accomplish three requisites: useful, applicable and active. A rule is *useful* in an evolution step if all targets are adjacent and not dissolved. In membrane 1 of the Figure 1, evolution rule r_4 is not useful in the initial configuration, but if membrane 2 is dissolved then membrane 4 and 6 become adjacent, and rule r_4 useful. On the other hand, a rule is *applicable* if its antecedent is included in the multiset of the membrane. Finally, a rule is *active* if there is no other applicable rule with higher priority.

The main goal of this work is to reduce the time of getting useful rules, avoiding communication as much as possible. The proposed solution is to define the membrane context associated to membranes and configurations in the P system.

Definition: The *membrane context* in a time is the set of children membranes to which rules can send objects in the current membrane structure of the system. These membranes are adjacent to the current one.

The basic idea is the following: every membrane in the P system has to know its context at every time. When a membrane is dissolved, then it has to report the dissolution to its father, and the latter will update its context.

Definition: a *usefulness state* q_i^j in a membrane *j* is a valid context in that membrane, $C(q_i^j)$. A context in a membrane is valid if it could be reached in any configuration of the system.

The target of this work is to find out statically all valid usefulness states at any membrane of a P System, the useful rules associated to each usefulness state, and transitions between states when membranes are dissolved.

Definition: Let $Child\_Of(j) = \{i \mid 1 <= i <= m;\ i \text{ is a child of } j \text{ en } \mu\}$, that is, all membrane *j* children in μ.

Definition: Let Q^j the set of *Usefulness states* associated to the membrane labelled with *j* in the P system Π, defined as follows:

1. if the membrane *j* is an elementary membrane: $Q^j = \{q_0^j\}$, *where* $C(q_0^j) = \{\varnothing\}$
2. if the membrane *j* is not an elementary membrane:

$$Q^j = \underset{i \in Child\_Of(j)}{X} Q'^{\,i}, \text{ where } Q'^{\,i} = \begin{cases} \{q_N^i\} & \text{if membrane } i \text{ cannot be dissolved} \\ \{q_N^i\} \cup Q^i & \text{if membrane } i \text{ can be dissolved} \end{cases}$$

q_N^i is a state representing that membrane *i* is not dissolved, therefore the context in q_N^i is $C(q_N^i) = \{i\}$.

Context for each one of the states belonging to the *Cartesian product* is obtained by the union of contexts which configure the corresponding state. $C((q_{s_1}^{i_1},..,q_{s_n}^{i_n})) = \bigcup_{i_k \in Child\_Of(j)} C(q_{s_k}^{i_k})$.

Considering the P system depicted in Figure 1, only evolution rules associated to membrane 1 are shown. Other membranes only show if there is any rule which can dissolve them; hence membranes with labels 2, 3, 4 and 5 can be dissolved during execution of the system. In order to determine *Usefulness states* per membrane, we shall start from inside to outside of the membrane system; that is, from elementary membranes to the skin membrane.

It seems to be clear that elementary membranes cannot have more than one state, with null context. Therefore, $Q^3 = \{q_0^3\}$, $Q^4 = \{q_0^4\}$, $Q^5 = \{q_0^5\}$ and $Q^6 = \{q_0^6\}$. Each one of these states has context $\{\varnothing\}$.

For membrane 2:

$$Q^2 = Q'^{\,4} \times Q'^{\,5} \times Q'^{\,6}$$

$$Q'^{\,4} = \{q_N^4\} \cup \{q_0^4\} \quad Contexts = \{\{4\},\{\varnothing\}\}$$

$$Q'^{\,5} = \{q_N^5\} \cup \{q_0^5\} \quad Contexts = \{\{5\},\{\varnothing\}\}$$

$$Q'^{\,6} = \{q_N^6\} \quad Contexts = \{\{6\}\}$$

$$Q^2 = \{\overbrace{(q_N^4,q_N^5,q_N^6)}^{q_0^2},\overbrace{(q_N^4,q_0^5,q_N^6)}^{q_1^2},\overbrace{(q_0^4,q_N^5,q_N^6)}^{q_2^2},\overbrace{(q_0^4,q_0^5,q_N^6)}^{q_3^2}\} \quad Contexts = \{\{4,5,6\},\{4,6\},\{5,6\},\{6\}\}$$

And finally, for membrane 1:

$$Q^1 = Q'^{\,2} \times Q'^{\,3}$$

$$Q'^{\,2} = \{q_N^2\} \cup \{q_0^2,q_1^2,q_2^2,q_3^2\} \quad Contexts = \{\{2\},\{4,5,6\},\{4,6\},\{5,6\},\{6\}\}$$

$$Q'^{\,3} = \{q_N^3\} \cup \{q_0^3\} \quad Contexts = \{\{3\},\{\varnothing\}\}$$

$$Q^1 = \{\overbrace{(q_N^2,q_N^3)}^{q_0^1},\overbrace{(q_N^2,q_0^3)}^{q_1^1},\overbrace{(q_0^2,q_N^3)}^{q_2^1},\overbrace{(q_0^2,q_0^3)}^{q_3^1},\overbrace{(q_1^2,q_N^3)}^{q_4^1},\overbrace{(q_1^2,q_0^3)}^{q_5^1},\overbrace{(q_2^2,q_N^3)}^{q_6^1},\overbrace{(q_2^2,q_0^3)}^{q_7^1},\overbrace{(q_3^2,q_N^3)}^{q_8^1},\overbrace{(q_3^2,q_0^3)}^{q_9^1}\}$$

$$Contexts = \{\{2,3\},\{2\},\{4,5,6,3\},\{4,5,6\},\{4,6,3\},\{4,6\},\{5,6,3\},\{5,6\},\{6,3\},\{6\}\}$$

Useful rules associated to usefulness states

Every Usefulness state is characterized by its context, that is, the set of children membranes directly enclosed in the original membrane. Hence, the context or state determines the set of useful rules in the membrane. Moreover, what is important to note is that the set of usefulness states, contexts and, hence, the set of evolution rules for each one of the membranes and possible configuration of the system can be established in a static analysis.

Lemma: An evolution rule $r = (u, v\xi), where\ \xi \in \{\delta, \lambda\}$ is useful in q_i^j if and only if $\forall\ TAR\ in_k \in v, k \in C(q_i^j)$.

Considering the previous P system Π for membrane 1, the table 1 shows the whole set of usefulness states – contexts and their corresponding sets of useful evolution rules accordingly to the states.

Usefulness states	Useful Rules
q_0^1 {2, 3}	r_1, r_2, r_3, r_6
q_1^1 {2}	r_3, r_6
q_2^1 {4, 5, 6, 3}	r_2, r_4, r_5, r_6
q_3^1 {4, 5, 6}	r_4, r_5, r_6
q_4^1 {4, 6, 3}	r_2, r_4, r_6
q_5^1 {4, 6}	r_4, r_6
q_6^1 {5, 6, 3}	r_2, r_5, r_6
q_7^1 {5, 6}	r_5, r_6
q_8^1 {6, 3}	r_2,, r_6
q_9^1 {6}	r_6

Table 1: Useful Evolution Rules associated to Usefulness States for Membrane 1

transitions between usefulness states

Definition: Let $Child\_D(j) = \{i \in Child(j) \wedge \exists r = (u,v,\delta) \in R_j\}$, be the set of child membranes to membrane j that can be dissolved.

Definition: Let $TC\_D(j) = Child\_D(j) \bigcup_{i \in Child\_D(j)} TC\_D(i)$, be the total context for membrane j, including only those membranes that can be dissolved. By total context is understood those membrane that eventually can become children of membrane j.

A transition between two *usefulness states* in a membrane is produced when a child membrane is dissolved. In this case, father membrane is affected and its usefulness state must change. The way for representing this behaviour is through a Moore's Sequential Machine in every membrane labelled with j.

$$MS^j = \left(\sum{}_I^j, \sum{}_O^j, Q^j, q_0^j, g^j, f^j\right), \text{ where:}$$

- Input alphabet: $\sum{}_I^j = \{\delta(i,q_s^i) \mid i \in TC\_D(j), q_s^i \in Q^i\}$, the sequential machine will transits when a child membrane is dissolved. Child membrane must send to membrane j that is dissolved and its usefulness state because the context of the membrane child will pass to be part of the parent context.
- Output alphabet: $\sum{}_O^j = \{r_k \mid r_k \in R_j\}$, the set of useful rules in membrane j.
- Set of states: $Q^j = \{(q_{s_1}^{i_1},..,q_{s_n}^{i_n}) \mid i_k \in Child\_Of(j), q_{s_k}^{i_k} \in Q'^{\,i_k}\}$, the set of usefulness states of membrane j.
- Initial state: $(q_N^{i_1},..,q_N^{i_n}) \mid i_k \in Child\_Of(j)$, that is, the state in which every child membrane is not dissolved.
- Output function: $g^j : Q \to \mathcal{P}(R_j)$. the function that assigns a set of useful rules to each one of the *usefulness state* of the membrane j; as it was shown in table 1.
- Transition function: $f^j : Q \times \Sigma_I^j \to Q$. the function provides the new usefulness state to transit given the current one and the dissolution of a child membrane. This function is defined as follows: $\forall i_k \in Child\_D(j)$

1) If i_k is dissolved, $i_k : f^j((q_{s_1}^{i_1},...,q_N^{i_k},...,q_{s_n}^{i_n}), \delta(i_k, q_s^{i_k})) = (q_{s_1}^{i_1},...,q_s^{i_k},...,q_{s_n}^{i_n})$.
2) If membrane m is dissolved being child of j and $m \in TC\_D(i_k)$:

$$f^j((q_{s_1}^{i_1},...,q_{s_k}^{i_k},...,q_{s_n}^{i_n}), \delta(m, q_s^m)) = (q_{s_1}^{i_1},...,q_p^{i_k},...,q_{s_n}^{i_n}) \quad where\ f^{i_k}(q_{s_k}^{i_k}, \delta(m, q_s^m)) = q_p^{i_k}$$

Hence, starting from states transition tables of children membranes, it will be obtained the transition table for membrane j. Of course, elementary membranes have not transition tables because of they have only one state. As an example, the transition function f^1 for membrane 1 of the P system of Figure 1 is depicted in table 2.

	$\delta(2,q_0^2)$	$\delta(2,q_1^2)$	$\delta(2,q_2^2)$	$\delta(2,q_3^2)$	$\delta(4,q_0^4)$	$\delta(5,q_0^5)$	$\delta(3,q_0^3)$
(q_N^2,q_N^3)	(q_0^2,q_N^3)	(q_1^2,q_N^3)	(q_2^2,q_N^3)	(q_3^2,q_N^3)	---	---	(q_N^2,q_0^3)
(q_N^2,q_0^3)	(q_0^2,q_0^3)	(q_1^2,q_0^3)	(q_2^2,q_0^3)	(q_3^2,q_0^3)	---	---	---
(q_0^2,q_N^3)	---	---	---	---	(q_2^2,q_N^3)	(q_1^2,q_N^3)	(q_0^2,q_0^3)
(q_0^2,q_0^3)	---	---	---	---	(q_2^2,q_0^3)	(q_1^2,q_0^3)	---
(q_1^2,q_N^3)	---	---	---	---	---	(q_3^2,q_N^3)	(q_1^2,q_0^3)
(q_1^2,q_0^3)	---	---	---	---	---	(q_3^2,q_0^3)	---

(q_2^2, q_N^3)	---	---	---	---	(q_3^2, q_N^3)	---	(q_2^2, q_0^3)
(q_2^2, q_0^3)	---	---	---	---	(q_3^2, q_0^3)	---	---
(q_3^2, q_N^3)	---	---	---	---	---	---	(q_3^2, q_0^3)
(q_3^2, q_0^3)	---	---	---	---	---	---	---

Table 2: Usefulness states transition function for membrane 1.

In Table 2 transitions for dissolutions of membranes 4 and 5 have been obtained from transition function of membrane 2 –shows in Table 3-, because they belong to the total context of membrane 2.

	$\delta(4, q_0^4)$	$\delta(5, q_0^5)$
q_0^2	q_2^2	q_1^2
q_1^2	---	q_3^2
q_2^2	q_3^2	---
q_3^2	---	---

Table 3: Usefulness states transition function for membrane 2.

As an example, if from the state (q_0^2, q_N^3) with context {4,5,6,3}, it is produced $\delta(4, q_0^4)$, then looking at transition table for membrane 2 from q_0^2 with $\delta(4, q_0^4)$, the result is q_2^2, and then the corresponding transition is to (q_2^2, q_N^3) with context {5,6,3}.

Finally, it can be changed the notation for representing usefulness states, in this case, they are numbering in a correlative manner starting from 0. That is, $\{q_0^1, q_1^1, q_2^1, q_3^1, q_4^1, q_5^1, q_6^1, q_7^1, q_8^1, q_9^1\}$, like in table 3 for membrane 2.

Usefulness states in transition P systems with Dissolution and Inhibition Capability.

Evolution rules in these systems are of the form, $u \rightarrow v\ \xi$, *where* $\xi \in \{\delta, \tau, \lambda\}$. Symbol τ indicates that after rule application membrane containing the rule will be not permeable to objects communication. This membrane will come back to be permeable to objects communication by the application of one evolution rule having the symbol *δ*. If during the application phase of evolution rules different rules having symbols δ and τ are applied, then membrane will not change its communication state.

Hence, it would be considered three different membrane states concerning to objects communication: Dissolved, Permeable and inhibited or impermeable. These three states and their transition graph are depicted in Figure 2

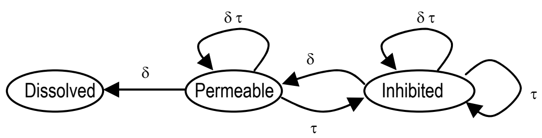

Figure 2

Inhibition capability modifies the previous study for *usefulness states* where only dissolution was allowed. Application of rules having the τ symbol in a membrane could modify its capability for accepting objects coming from outside. Hence, this fact modifies the context of the parent membrane, because rules sending objects to a membrane in the inhibited state are not useful.

Definition: Let Q^j the set of *Usefulness states* associated to the membrane labelled with *j*, defined as follows:

1. if the membrane *j* is an elementary membrane: $Q^j = \{q_0^j\}$, *where* $C(q_0^j) = \{\varnothing\}$
2. if the membrane *j* is not an elementary membrane: $Q^j = \underset{i \in Child\_Of(j)}{X} Q'^i$, where

$$Q'^i = \begin{cases} \{q_N^i\} \text{ if membrane } i \text{ can be neither dissolved nor inhibited} \\ \{q_N^i, q_I^i\} \text{ if membrane } i \text{ can be inhibited, but not dissolved} \\ \{q_N^i\} \cup Q^i \text{ if membrane } i \text{ can be dissolved, but not inhibited} \\ \{q_N^i, q_I^i\} \cup Q^i \text{ if membrane } i \text{ can be inhibited and dissolved} \end{cases}$$

q_N^i represents the permeable state fo the membrane i, therefore $C(q_N^i) = \{i\}$.

q_I^i represents the inhibited state of the membrane *i*, therefore $C(q_I^i) = \varnothing$.

Useful rules associated to usefulness states

In order to determine which evolution rules are useful in a determined membrane and evolution step, now it is needed to assure not only that evolution rules targets of type in_k are all of then in the membrane context, but also current membrane must be permeable if target of type *out* is included. Hence, it is needed to consider the *usefulness state* and permeability state of the membrane; and then, it could be possible to abroad the static analysis of usefulness states for P systems with membrane dissolution and permeability control.

Lemma: An evolution rule $u \rightarrow v\ \xi$, *where* $\xi \in \{\delta, \tau, \lambda\}$ is useful in q_i^j *and* q_{perm}^j if and only if $\forall\ TAR\ in_k \in v, k \in C(q_i) \wedge Si\ \exists\ TAR\ out \in v, q_{perm}^j = Permeable$

Transitions between usefulness states

Definition: Let $Child\_I(j) = \{i \in Child\_Of(j) \wedge \exists r = (u, v, \tau) \in R_j\}$ be the set of children membranes of membrane *j* that can be inhibited.

Definition: Let $TC\_I(j) = Child\_I(j) \bigcup_{i \in Child\_D(j)} TC\_I(i)$, be the membrane *j* total context considering only those children membranes that can be inhibited.

In these systems, transitions are not only produced by membranes dissolution (δ), but also with membranes inhibition (τ) and come back permeable ($\neg\tau$). Therefore, the alphabet for the sequential states machines is:

$$\sum{}_I^j = \{\delta(i, q_s^i) \mid i \in TC\_D(j), q_s^i \in Q^i\} \cup \{\tau\, i \mid i \in TC\_I(j)\} \cup \{\neg\tau\, i \mid i \in TC\_D(j) \cap TC\_D(j)\}$$

And the transition function is:

$\forall i_k \in Child\_Of(j)$

If i_k is dissolved: $f^j((q_{s_1}^{i_1}, ..., q_N^{i_k}, ..., q_{s_n}^{i_n}), \delta(i_k, q_s^{i_k})) = (q_{s_1}^{i_1}, ..., q_s^{i_k}, ..., q_{s_n}^{i_n})$

if *m* is dissolved being child of *j* and $m \in TC\_D(i_k)$: $f^j((q_{s_1}^{i_1}, ..., q_{s_k}^{i_k}, ..., q_{s_n}^{i_n}), \delta(m, q_s^m)) = (q_{s_1}^{i_1}, ..., q_p^{i_k}, ..., q_{s_n}^{i_n})$

$$where\ f^{i_k}(q_{s_k}^{i_k}, \delta(m, q_s^m)) = q_p^{i_k}$$

If i_k is inhibited: $f^j((q_{s_1}^{i_1}, ..., q_N^{i_k}, ..., q_{s_n}^{i_n}), \tau\, i_k) = (q_{s_1}^{i_1}, ..., q_I^{i_k}, ..., q_{s_n}^{i_n})$

If *m* is inhibited being child of *j* and $m \in TC\_I(i_k)$: $f^j((q_{s_1}^{i_1}, ..., q_{s_k}^{i_k}, ..., q_{s_n}^{i_n}), \tau\, m) = (q_{s_1}^{i_1}, ..., q_p^{i_k}, ..., q_{s_n}^{i_n})$

$$where\ f^{i_k}(q_{s_k}^{i_k}, \tau\, m) = q_p^{i_k}$$

If i_k comes back to be permeable: $f^j((q_{s_1}^{i_1}, ..., q_I^{i_k}, ..., q_{s_n}^{i_n}), \neg\tau\, i_k) = (q_{s_1}^{i_1}, ..., q_N^{i_k}, ..., q_{s_n}^{i_n})$

if *m* comes back to be permeable being child of *j* and $m \in TC\_D(i_k) \cap TC\_I(i_k)$:

$$f^j((q_{s_1}^{i_1}, ..., q_{s_k}^{i_k}, ..., q_{s_n}^{i_n}), \neg\tau\, m) = (q_{s_1}^{i_1}, ..., q_p^{i_k}, ..., q_{s_n}^{i_n})\ where\ f^{i_k}(q_{s_k}^{i_k}, \neg\tau\, m) = q_p^{i_k}$$

Encoding usefulness states

The main problem when usefulness states are encoded in a determined Hardware/Software architecture could be the size of transition states tables used for representing usefulness states transition functions in membranes. This is the reason why in this paper is proposed a way for encoding usefulness states with the purpose of making transition without using usefulness states transition tables.

Definition:

Let $TC(j) = Child\_Of(j) \bigcup_{i \in Child\_D(j)} TC(i)$, the total context for membrane j , independently of dissolving or inhibition.

The appearing membranes order in $TC(j)$, is normalized going down into the sub-tree of μ starting in membrane *j* in depth and in pre-order. And they are represented in this order in the *Normalized Total Context* of membrane *j*.

Definition:

Let $TC_{Normal}(j) = (i_1, TC_{Normal}(i_1), ..., i_n, TC_{Normal}(i_n))$ $where\ i_k \in Child\_Of(j)\ from\ left\ to\ right\ in\ \mu$

Each one of the usefulness states of membrane *j*, q_i^j is enconded on $TC_{Normal}(j)$ depending on its context, $C(q_i^j)$, with binary logic. The value 1 set out that membrane *k* is present in $C(q_i^j)$, while value 0 will represents that membrane k is not in $C(q_i^j)$. As an example, for membrane 1 of the P system depicted in Figure 1, it is obtained the total context $TC_{Normal}(1) = (2,4,5,6,3)$, and the usefulness states enconded are represented in table 4.

Usefulness states	Encoding
q_0^1 {2, 3}	10001
q_1^1 {2}	10000
q_2^1 {4, 5, 6, 3}	01111
q_3^1 {4, 5, 6}	01110
q_4^1 {4, 6, 3}	01011
q_5^1 {4, 6}	01010
q_6^1 {5, 6, 3}	00111
q_7^1 {5, 6}	00110
q_8^1 {6, 3}	00011
q_9^1 {6}	00010

Table 4: Encoding of usefulnes states

If $q^j(t) = (i_1,....,i_k,.....,i_n)$ *encoded by* $TC_{Normal}(j)$ is the usefulness state of membrane *j* at time *t*, the transitional logic will be the following:

1. If the child membrane of *j*, i_k, at time *t* is inhibited: $q^j(t+1) = (i_1,....,0,.....,i_n)$
2. If the child membrane of *j*, i_k, at time *t* comes back to be permeable: $q^j(t+1) = (i_1,....,1,.....,i_n)$
3. If the child membrane of *j*, i_k, at time *t* is dissolved, it has to send its usefulness state $q^{i_k}(t+1)$, encoded by its normalized total context, $TC_{Normal}(i_k)$. It can be considered in a deeper sight the usefulness state for membrane *j* as $q^j(t) = (i_1,....,i_k,TC(i_k),.....,i_n)$ and the transition is $q^j(t+1) = (i_1,....,0,q^{i_k}(t+1),.....,i_n)$

In the proposed example, if membrane 1 is in usefulness state $q^1(t) = (10001)$ and membrane 2 is dissolved in $q^2(t) = (101)$ encoded by its normalized total context $TC_{Normal}(2) = (4,5,6)$, it is obtained the transition $q^1(t+1) = (01011)$. This is the transition of table 2 $f^1((q_N^2, q_N^3), \delta(2, q_1^2)) = (q_1^2, q_N^3)$ without making use of table.

Conclusion

This paper presents the study of usefulness states associated to membranes of Transition P system. The aim of the work developed here is to reduce the evolution rules application time. In order to get the necessary efficiency in the application phase of rules, the analysis of usefulness states can be done in a static manner, and this implies an important reduction in time needed for evolution steps in the system. Moreover, not only usefulness states are defined here, but also the logic of transition between them. Each one of the usefulness states is associated to its own set of useful rules, and in this way there is no computation needed to obtain them because the computation of usefulness states and context is done before starting system execution or simulation.

Bibliography

[Ciobanu 2004] G.Ciobanu, G.Wenyuan, "A P System runnning on a cluster of computers", Proceedings of Membrane Computing. International Workshop, Tarragona (Spain). Lecture Notes in Computer Science, vol 2933, 123-150.

[Păun, 1998] Gh.Păun, "Computing with Membranes", Journal of Computer and System Sciences, 61(2000), and Turku Center of Computer Science-TUCS Report nº 208, 1998.

[Syropoulos 2003] A. Syropoulos, E.G. Mamatas, P.C. Allilomes, K.T. Sotiriades, "A distributed simulation of P systems". Preproceedings of the Workshop on Membrane Computing (A. Alhazov, C.Martin-Vide and Gh.Păun, eds); Tarragona, vol July 17-22 (2003), 455-460.

[Tejedor, 2007] J.Tejedor, L.Fernández, F.Arroyo, G.Bravo, *An architecture for attacking the bottleneck communication in P systems*. In: M. Sugisaka, H. Tanaka (eds.), Proceedings of the 12th Int. Symposium on Artificial Life and Robotics, Jan 25-27, 2007, Beppu, Oita, Japan, 500-505.

Authors' Information

Juan Alberto de Frutos – e-mail: jafrutos@eui.upm.es

Luis Fernández – e-mail: setillo@eui.upm.es

Fernando Arroyo – e-mail: farroyo@eui.upm.es

Gines Bravo – e-mail: gines@eui.upm.es

Dpto. Lenguajes, Proyectos y Sistemas Informáticos (LPSI) de la Escuela Universitaria de Informática (EUI) de la Universidad Politécnica de Madrid (UPM); Ctra. Valencia, km. 7, 28031 Madrid (Spain).