

# Simulating Tritrophic Interactions by Means of P Systems

M. Angels Colomer <sup>\*1</sup>, Ignacio Pérez-Hurtado <sup>#2</sup>, Mario J. Pérez-Jiménez <sup>#4</sup>, Agustín Riscos-Núñez <sup>#3</sup>

<sup>#</sup> *Dpt. of Computer Science and Artificial Intelligence, University of Seville*

*Avda. Reina Mercedes s/n, 41012 Sevilla, Spain*

<sup>2</sup> [perezh@us.es](mailto:perezh@us.es)

<sup>3</sup> [marper@us.es](mailto:marper@us.es)

<sup>4</sup> [ariscosn@us.es](mailto:ariscosn@us.es)

<sup>\*</sup> *Department of Mathematics, University of Lleida*

*Avda. Alcalde Rovira Roure, 191, 25198 Lleida, Spain*

<sup>1</sup> [colomer@matematica.udl.es](mailto:colomer@matematica.udl.es)

**Abstract**—P systems provide a high level computational modelling framework that combines the structural and dynamical aspects of ecosystems in a compressive and relevant way. The inherent randomness and uncertainty in biological systems is captured by using probabilistic strategies. The design of efficient simulation algorithms in order to reproduce the behavior of these computational models over conventional computers is fundamental for the validation and virtual experimentation processes. In this paper, we describe the modelling framework and two different simulation algorithms. As a case study, a P system based model of an ideal ecosystem with three trophic levels is designed and simulated by both simulation algorithms, providing comparisons of efficiency between them.

## I. INTRODUCTION

Modeling a biological system is usually very complicated, if one wants to take into account the most important factors involved and the relationships between them. Therefore it is normal to define a scenario in which the number of variables and interactions between variables is as limited as possible.

Most of the existing models for the study of population dynamics are based on differential equations, but this approach has some drawbacks. On one hand, when the number of species in a model is greater than two, the equations system proposed is so complex that it is usually solved using numerical methods. Besides, improvements on the performance of the models are generally obtained by the addition of ingredients, which in the case of ODEs means that the whole modeling process needs to be done again from scratch.

There already exists a quite large literature concerning several approaches to modeling different phenomena within the membrane computing framework (see e.g. [1], [9], [11] and the chapter devoted to *Probabilistic/stochastic models* in the handbook [8]). Computer models based on P systems offer significant advantages: modularity, parallelism, and no limitation on the number of interrelated variables that evolve in parallel. These properties make them very attractive for modeling complex ecosystems.

Each ecosystem has its own important peculiarities, but nevertheless, there are some aspects common to most ecosystems such as:

- they contain a large number of individuals and a large number of species.
- the life cycle includes some basic processes such as: feeding, growth, reproduction and death.
- these processes are annually repeated.
- the evolution often depends on the environment: climate, soil, ...
- the natural dynamics suffer modifications due to human activities.

These common features yield some requisites for the model, from a computational point of view: many processes take place simultaneously, there is cooperation between individuals and elements of the ecosystem, partial synchronization among the dynamic evolution sub-ecosystems (for example, there could be adverse weather conditions some year, and this does not affect a single sub-ecosystem, but has a global influence on the entire ecosystem), situations need to be restored annually.

These considerations led to the definition of an appropriate modeling semantic context for the P system. In particular, a precise semantics of the multienvironment functional probabilistic P system with active membranes has been used to model two real ecosystems: One, dealing with the scavenger birds in the Catalan Pyrenees (Spain) [2] and another one focusing on the *zebra mussel* in the reservoir of Ribarroja (Spain) [3]. In the first case an endangered species is modeled, the purpose of the obtained model is the study of the evolution of the ecosystem under different scenarios to make the most appropriate management decisions for the conservation of species. The second case study corresponds to a completely different situation: zebra mussel is an exotic species that has shown an excellent adaptation after being introduced in the reservoir. Its uncontrolled reproduction causes significant economic and ecological damage.

In both cases we have designed a simulator to validate the results, managers now have two tools enabling them to perform virtual experiments under different conditions.

This paper introduces an idealized ecosystem that can be used when testing simulators, instead of running tests on huge instances like the models corresponding to the above

mentioned real case studies. The presented ecosystem contains species belonging to three trophic levels: in the lower level we have the grass, then we have 5 species of herbivores, and above all of them we have a carnivore species at the third level.

This is an idealized ecosystem, in the sense that it does not correspond to any real system. However, it has been designed under the guidance of ecologists, and it takes into account some relevant facts that make it somehow plausible. Some of these details will be explained later on.

The paper is structured as follows. The next section describes in general the modeling framework based on P systems. In section III the main contribution of the paper is detailed, according to the previous framework. Then, two different simulation algorithms are explained, and simulation results comparing their performance are shown in section VI. The paper ends with some conclusions and final remarks.

## II. A P SYSTEM BASED MODELING FRAMEWORK

*Definition 1:* A multienvironment probabilistic functional extended P system with active membranes of degree  $(q, m)$  with  $q \geq 1$ ,  $m \geq 1$ , taking  $T$  time units,  $T \geq 1$ , is a tuple

$$\Delta_{(q,m)} = (G, \Gamma, \Sigma, R_E, \Pi, \{f_{r,j} : r \in R_{\Pi}, 1 \leq j \leq m\}, \{\mathcal{M}_{ij} : 0 \leq i \leq q-1, 1 \leq j \leq m\})$$

where:

- $G = (V, S)$  is a directed graph such that  $(x, x) \in S$ , for each  $x \in V$ . Let  $V = \{e_1, \dots, e_m\}$  whose elements are called environments;
- $\Gamma$  is the working alphabet and  $\Sigma \subsetneq \Gamma$  is an alphabet representing the objects that can be present in the environments;
- $R_E$  is a finite set of communication rules between environments of the form

$$(x)_{e_j} \xrightarrow{p(x,j,j_1,\dots,j_h)} (y_1)_{e_{j_1}} \dots (y_h)_{e_{j_h}}$$

where  $x, y_1, \dots, y_h \in \Sigma$ ,  $(e_j, e_{j_l}) \in S$  ( $l = 1, \dots, h$ ) and  $p(x,j,j_1,\dots,j_h)(t) \in [0, 1]$ , for each  $t = 1, \dots, T$ . If  $p(x,j,j_1,\dots,j_h)(t) = 1$ , for each  $t$ , then we omit the probabilistic function. These rules verify the following:

- $\star$  for each  $e_j$  and for each  $x$ , the sum of functions associated with the rules from  $R_E$  whose left-hand side is  $(x)_{e_j}$  coincides with the constant function equal to 1.
- $\Pi = (\Gamma, \mu, R_{\Pi})$  where
  - $\mu$  is a membrane structure consisting of  $q$  membranes, with the membranes injectively labeled by  $0, \dots, q-1$ . The skin membrane is labeled by 0. We also associate electrical charges from the set  $\{0, +, -\}$  with membranes; and
  - $R_{\Pi}$  is a finite set of evolution rules of the form  $r : u[v]_i^\alpha \rightarrow u'[v']_i^{\alpha'}$  where  $u, v, u', v' \in M(\Gamma)$ ,  $i \in \{0, 1, \dots, q-1\}$ , and  $\alpha, \alpha' \in \{0, +, -\}$ ;
- For each  $r \in R_{\Pi}$  and for each  $j$ ,  $1 \leq j \leq m$ ,  $f_{r,j}$  is a computable function whose domain is  $\{1, 2, \dots, T\}$  and its range is contained in  $[0, 1]$ , verifying the following:

- $\star$  For each  $u, v \in M(\Gamma)$ ,  $i \in \{0, \dots, q-1\}$  and  $\alpha \in \{0, +, -\}$ , if  $r_1, \dots, r_z$  are the rules from  $R_{\Pi}$  whose left-hand side is  $u[v]_i^\alpha$ , then  $\sum_{j=1}^z f_{r_j}(t) = 1$ , for each  $t$ ,  $1 \leq t \leq T$ .

- For each  $j$  ( $1 \leq j \leq m$ ),  $\mathcal{M}_{0j}, \dots, \mathcal{M}_{q-1,j}$  are strings over  $\Gamma$ , describing the multisets of objects initially placed in the  $q$  regions of  $\mu$ , within the environment  $e_j$ .

In other words, a system  $\Delta_{(q,m)}$  as described in the definition can be viewed as a set of  $m$  environments  $e_1, \dots, e_m$  linked between them by the arcs from the directed graph  $G$ . Each environment  $e_j$  contains a functional probabilistic P system with active membranes of degree  $q$ , each of them with the *same* skeleton,  $\Pi$ , and such that  $\mathcal{M}_{0j}, \dots, \mathcal{M}_{q-1,j}$  describe their initial multisets.

When a communication rule between environments

$$(x)_{e_j} \xrightarrow{p(x,j,j_1,\dots,j_h)} (y_1)_{e_{j_1}} \dots (y_h)_{e_{j_h}}$$

is applied, object  $x$  passes from  $e_j$  to  $e_{j_1}, \dots, e_{j_h}$  possibly modified into objects  $y_1, \dots, y_h$ , respectively. In any moment  $t$ ,  $1 \leq t \leq T$ , for each object  $x$  in environment  $e_j$ , if there exist communication rules with  $(x)_{e_j}$  on their left-hand side, then one of these rules will be applied. If more than one communication rule can be applied to an object, the system selects one randomly, according to their probability which is given by  $p(x,j,j_1,\dots,j_h)(t)$ .

We assume that a global clock exists, marking the time for the whole system (for its compartments), that is, all membranes and the application of all rules are synchronized.

The tuple of multisets of objects present at any moment in the  $m$  environments and at each of the regions of the P systems located within them, and the polarizations of the membranes in each P system, constitutes a *configuration* of the system at that moment. At the initial configuration of the system we assume that all environments are empty and all membranes have a neutral polarization.

The P system can pass from one configuration to another by using the rules from  $R = R_E \cup \bigcup_{j=1}^m R_{\Pi_j}$  as follows: at each transition step, the rules to be applied are selected according to the probabilities assigned to them, and all applicable rules are simultaneously applied.

## III. A P SYSTEM BASED MODEL OF TRITROPHIC INTERACTIONS

Feeding, reproduction and mortality are basic processes for all living organisms. These natural processes are those considered in the model of tritrophic interactions presented here. The model is a simplification of a real ecosystem and many details have been skipped. It should be considered as a simple approximation facilitating a better understanding of the methodology that allows to model actual ecosystems by means of P systems (see e.g. [2] and [3] where two existing ecosystems of great ecological interest are modeled).

The model scheme is composed by 5 modules as shown in Figure 1. A complete cycle is executed in 9 steps of computation and it represents one year in the ecosystem.

We will divide the ecosystem into 10 areas with different weather, orography, and soil conditions. The animals only move between areas if there is lack of resources.

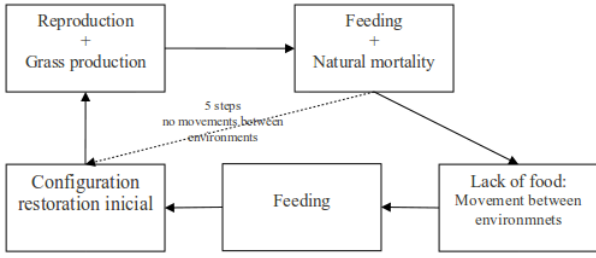


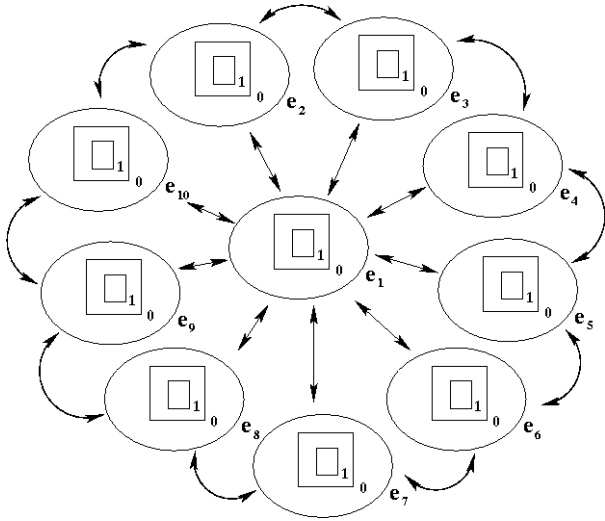
Fig. 1. Modules scheme

The ecosystem is modeled by using a multienvironment probabilistic functional extended P system with active membranes of degree (10, 2) taking  $T$  time units:

$$(G, \Gamma, \Sigma, \Pi, R_E, \{f_{r,j} : r \in R_{\Pi}, 1 \leq j \leq 10\}, \{\mathcal{M}_{ij} : 0 \leq i \leq 1, 1 \leq j \leq 10\})$$

where:

- The graph of the system is  $G = (V, S)$ , where  $V = \{e_1, \dots, e_{10}\}$  and the arcs are  $(v, v)$  for each  $v \in V$ , adding the arcs represented in the next figure:



- $\Gamma = \{X_1, G\} \cup \{X_i, X'_i, Y_i : 2 \leq i \leq 7\} \cup \{R_i : 0 \leq i \leq 6\}$ .

The symbols  $X_i, X'_i$  and  $Y_i$  with  $2 \leq i \leq 7$  represent animals of species  $i$ .  $X_1$  is associated with 1 hectare (He) of land. The symbol  $G$  represents 1 Kg of grass and  $R_i$  with  $0 \leq i \leq 6$  is an auxiliary synchronization counter.

- $\Sigma = \{\bar{X}_i, \bar{X}'_i : 2 \leq i \leq 7\}$ .
- $\Pi = (\Gamma, [[ ]_1]_0, R_{\Pi})$ .
- $R_E$  is the following set of rules:

$$\{(X_i)_{e_k} \xrightarrow{p_{k,s,i}} (X'_i)_{e_s} : 1 \leq k, s \leq 10, 2 \leq i \leq 7\}$$

- $\{f_{r,j} : r \in R_{\Pi}, 1 \leq j \leq 10\}$  is a set of constant functions whose range is contained in  $[0, 1]$ .

- $\{\mathcal{M}_{0j}, \mathcal{M}_{1j} : 1 \leq j \leq 10\}$  are strings over  $\Gamma$ , describing the multisets of objects initially placed in the two regions of  $\mu$  for each  $\Pi_j : 1 \leq i \leq 10$ .

- $\mathcal{M}_{0j} = \{R_0, X_1^{q_{1,j}}\}$ , for  $1 \leq j \leq 10$ .

- $\mathcal{M}_{1j} = \{X_2^{q_{2,j}}, \dots, X_7^{q_{7,j}}\}$ , for  $1 \leq j \leq 10$ .

- $R = R_E \cup \bigcup_{j=1}^m R_{\Pi_j}$  is a set of rules composed by the rules described below:

- **Module 1:**

(Grass production and species reproduction)

\* Grass production.

$$r_{1,j} \equiv X_1 [ ]_1^0 \xrightarrow{m_j} [X_1, G^{h_j}]_1^+, 1 \leq j \leq 3$$

The environmental conditions affect the amount of produced grass per unit of surface (He). For the sake of simplicity only three scenarios are considered: low, normal and high production (with probabilities  $m_1, m_2$  and  $m_3$ , respectively).

\* Females which reproduce and generate  $d_i$  descendants.

$$r_{2,i} \equiv [X_i]_1^0 \xrightarrow{k_{i,1} \cdot 0.5} [X_i^{1+d_i}]_1^+, 2 \leq i \leq 7$$

\* Females and males which do not reproduce.

$$r_{3,i} \equiv [X_i]_1^0 \xrightarrow{1-k_{i,1} \cdot 0.5} [X_i]_1^+, 2 \leq i \leq 7$$

\* P system synchronization.

$$r_4 \equiv R_0 [ ]_1^0 \rightarrow [R_0]_1^+$$

A 1:1 ratio between females and males is assumed. Moreover, all the females are able to reproduce. The first of these assumptions is fulfilled by most of animal species, but not the second one, since fertility depends on the age.

- **Module 2:**

(Feeding and mortality)

\* Animals which feed and survive.

$$r_5 \equiv [X_i G^{f_i}]_1^+ \xrightarrow{1-k_{i,2}} [Y_i]_1^-, 2 \leq i \leq 6$$

$$r_6 \equiv [X_7 X_i^{f_7}]_1^+ \xrightarrow{1-k_{7,2}} [Y_7]_1^-, 2 \leq i \leq 6$$

\* Animals which feed and do not survive.

$$r_7 \equiv [X_i G^{f_i}]_1^+ \xrightarrow{k_{i,2}} [ ]_1^-, 2 \leq i \leq 6$$

$$r_8 \equiv [X_7 X_i^{f_7}]_1^+ \xrightarrow{k_{7,2}} [ ]_1^-, 2 \leq i \leq 6$$

\* P system synchronization.

$$r_9 \equiv [R_0]_1^+ \rightarrow [R_0]_1^-$$

The feeding and mortality processes are considered at the same step. When an animal dies, it has already eaten the necessary amount of food to survive for a year. This assumption is clearly unrealistic, since animals may die at any period of the year. Note that only mortality due to natural causes has been considered in this module.

- **Module 3:**

(Movement of species through environments)

\* The objects related to species which have not eaten go to the skin membrane.

$$r_{10,i} \equiv [X_i]_1^- \longrightarrow X_i [ ]_1^-, 2 \leq i \leq 7$$

\* The objects in the skin go to the environment.

$$r_{11,i} \equiv [X_i]_0^0 \longrightarrow [X_i]_0^0, 2 \leq i \leq 7$$

\* Movement of objects between environments.

$$r_{12,k,s,i} \equiv (X_i)_{e_k} \xrightarrow{p_{k,s,i}} (X_i)_{e_s}, \\ 1 \leq k \leq 10, 1 \leq s \leq 10, 2 \leq i \leq 7$$

\* Objects  $X_i'$  go into the skin membrane.

$$r_{13,i} \equiv X_i' [ ]_0^0 \longrightarrow [X_i']_0^0, 2 \leq i \leq 7$$

\* Objects  $X_i'$  go into the inner membrane.

$$r_{14,i} \equiv X_i' [ ]_1^- \longrightarrow [X_i']_1^-, 2 \leq i \leq 7$$

\* P system synchronization.

$$r_{15,l} \equiv [R_l]_1^- \longrightarrow [R_{l+1}]_1^-, 0 \leq l \leq 4$$

If there is lack of food, there is a possibility for animals to migrate to a neighbor area looking for resources.

#### – Module 4:

(Feeding)

If there are resources in the new area, then animals that just arrived have a chance to feed and survive.

$$r_{16} \equiv [X_i' G^f]_1^- \xrightarrow{1-k_{i,2}} [Y_i]_1^0, 2 \leq i \leq 6$$

$$r_{17} \equiv [X_7' X_i'^{f7}]_1^- \xrightarrow{1-k_{7,2}} [Y_7]_1^0, 2 \leq i \leq 6$$

$$r_{18} \equiv [X_7' Y_i'^{f7}]_1^- \xrightarrow{1-k_{7,2}} [Y_7]_1^0, 2 \leq i \leq 6$$

$$r_{19} \equiv [X_i' G^f]_1^- \xrightarrow{k_{i,2}} [ ]_1^0, 2 \leq i \leq 6$$

$$r_{20} \equiv [X_7' X_i'^{f7}]_1^- \xrightarrow{k_{7,2}} [ ]_1^0, 2 \leq i \leq 6$$

$$r_{21} \equiv [X_7' Y_i'^{f7}]_1^- \xrightarrow{k_{7,2}} [ ]_1^0, 2 \leq i \leq 6$$

$$r_{22} \equiv [R_5]_1^- \longrightarrow [R_6]_1^0$$

At this stage of the cycle, objects representing animals that species 7 can predate on may be found in the P system as  $X_i$  or  $Y_i$ , for  $2 \leq i \leq 6$ .

#### – Module 5:

(Reinitialization of the cycle)

$$r_{23,i} \equiv [Y_i]_1^0 \longrightarrow [X_i]_1^0, 2 \leq i \leq 7$$

$$r_{24} \equiv [R_6]_1^0 \longrightarrow [R_0]_1^0$$

$$r_{25} \equiv [X_1]_1^0 \longrightarrow X_1 [ ]_1^0$$

$$r_{26,i} \equiv [X_i]_1^0 \longrightarrow [ ]_1^0, 2 \leq i \leq 7$$

$$r_{27} \equiv [G]_1^0 \longrightarrow [ ]_1^0$$

Following is the list of necessary initial parameters:

#### • Animal related parameters:

- $k_{i,1}$ , ( $2 \leq i \leq 7$ ): Fertility ratio for species  $i$  (females).
- $k_{i,2}$ , ( $2 \leq i \leq 7$ ): Survival ratio for species  $i$ .
- $f_i$ , ( $2 \leq i \leq 7$ ): Amount of food units consumed by animal for species  $i$ .
- $d_i$ , ( $2 \leq i \leq 7$ ): Number of descendants per female and birth for species  $i$ .
- $q_{i,j}$ , ( $2 \leq i \leq 7, 1 \leq j \leq 10$ ): Initial number of individuals of species  $i$  in area  $j$
- $p_{k,s,i}$ , ( $2 \leq i \leq 7, 1 \leq k, s \leq 10$ ): Probability of movement of species  $i$  from the area  $k$  to  $s$ .

#### • Grass related parameters:

- $h_i$ ,  $1 \leq i \leq 3$ : Amount of produced grass per hectare in different conditions ( $i = 1$  unfavorable

conditions,  $i = 2$  normal conditions and  $i = 3$  favorable conditions).

- $m_i$ ,  $1 \leq i \leq 3$ : Probability of the corresponding condition affecting the production of grass.
- $q_{1,j}$ , ( $1 \leq j \leq 10$ ): Number of land hectares in area  $j$ .

## IV. BINOMIAL BLOCK BASED SIMULATION ALGORITHM

In this section we describe the first simulation algorithm developed for multienvironment probabilistic P systems [3]. It follows a strategy based on the binomial distribution and blocks of rules with the same left-hand side.

In general, each simulation step is divided into two main stages: *selection* and *execution*. In the first one, the algorithm decides which rules will be applied, and the number of applications for each one (taking into account their left-hand sides and the available objects in the current configuration). In the second stage, the selected rules are applied, consuming the multisets of the rules' left-hand sides and adding the multisets of the rule's right-hand sides the selected number of times, and possibly changing the polarization of membranes.

Next we describe the selection stage.

**Input:** A multienvironment functional P system with active membranes of degree  $(q, m, n)$  with  $q \geq 1$ ,  $m \geq 1$ ,  $n \geq 1$ , taking  $T$  time units,  $T \geq 1$ .

- 1: Rules are classified into sets (blocks) so that all the rules belonging to a block have the same left-hand side. Note that rules from different blocks may have overlapping left-hand sides.
- 2: Let  $F_b(N, p)$  be a function that returns a discrete random number within the binomial distribution  $B(N, p)$ .
- 3: **for** each step of simulation **do**
- 4: A random order on the family of blocks is considered.
- 5: **for all** blocks, according to the considered order **do**
- 6: A random order on the rules of the block,  $\{r_1, \dots, r_t\}$ , is selected.
- 7: Let us suppose that the common left-hand side is  $u [v]_i^\alpha$  and their respective probabilistic constants are  $c_{r_1}, \dots, c_{r_t}$ .
- 8: The highest number  $N$  is computed so that  $u^N$  appears in the parent membrane of  $i$  and  $v^N$  appears in membrane  $i$ .
- 9: let  $d = 1$
- 10: **for all**  $k$  ( $1 \leq k \leq t - 1$ ), according to the selected order **do**
- 11: let  $c_{r_k}$  be  $\frac{c_{r_k}}{d}$
- 12: let  $n_{r_k}$  be  $F(N, c_{r_k})$
- 13: let  $N$  be  $N - n_{r_k}$
- 14: let  $q$  be  $1 - c_{r_k}$
- 15: let  $d$  be  $d * q$
- 16: **end for**
- 17: let  $n_{r_t}$  be  $N$
- 18: The pair  $\langle r, n_r \rangle$  is added to  $R_{sel}$ , which means that each rule  $r$  is applied  $n_r$  times.
- 19: **end for**
- 20: **end for**

This simulation algorithm is useful for the most of cases but it has the next disadvantages:

- It needs to classify the rules by their left-hand side.
- It does not handle rules with intersections on their left-hand sides.
- It does not check the consistency of charges in the selection of rules.
- It does not evaluate probabilistic functions related to rules.

#### V. DIRECT NON-DETERMINISTIC DISTRIBUTION ALGORITHM WITH PROBABILITIES (DNDP)

The following is a more efficient simulation algorithm for multienvironment probabilistic P systems inspired by [7] that overcomes the weak points of the simulation algorithm described above.

**Input:** A multienvironment functional P system with active membranes of degree  $(q, m, n)$  with  $q \geq 1$ ,  $m \geq 1$ ,  $n \geq 1$ , taking  $T$  time units,  $T \geq 1$ , and a natural number  $K \geq 1$ .

- 1: **for**  $t \leftarrow 0$  to  $T - 1$  **do**
- 2:  $C_t \leftarrow$  configuration of the system at the moment  $t$
- 3:  $C'_t \leftarrow C_t$
- 4: initialization
- 5: First selection phase. It generates a multiset of *consistent* applicable rules.
- 6: Second selection phase. It generates a multiset of *maximally consistent* applicable rules.
- 7: Execution of selected rules.
- 8:  $C_{t+1} \leftarrow C'_t$
- 9: **end for**

#### Initialization

- 1:  $R_{\Pi} \leftarrow$  ordered set of rules of  $\Pi$
- 2: **for**  $j \leftarrow 1$  to  $m$  **do**
- 3:  $R_{E,j} \leftarrow$  ordered set of rules from  $R_E$  related to the environment  $j$
- 4:  $A_j \leftarrow$  ordered set of rules from  $R_{E,j}$  whose probability at the moment  $t$  is  $> 0$
- 5:  $M_j \leftarrow$  ordered set of pairs  $\langle \text{label}, \text{charge} \rangle$  for all the membranes from  $C_t$  contained in the environment  $j$
- 6:  $B_j \leftarrow \emptyset$
- 7: **for each**  $\langle h, \alpha \rangle \in M_j$  (following the considered order) **do**
- 8:  $B_j \leftarrow B_j \cup$  ordered set of rules  $u[v]_h^\alpha \leftarrow u'[v']_h^\beta$  from  $R_{\Pi}$  whose probability at the moment  $t$  is  $> 0$  for the environment  $j$
- 9: **end for**
- 10: **end for**

#### First selection phase (consistency)

- 1: **for**  $j \leftarrow 1$  to  $m$  **do**
- 2:  $R_{sel,j}^1 \leftarrow$  the empty multiset
- 3:  $R_{sel,j}^2 \leftarrow$  the empty multiset
- 4: **for**  $k \leftarrow 1$  to  $K$  **do**
- 5:  $D_j \leftarrow A_j \cup B_j$  with a *random order*
- 6: **for each**  $r \in D_j$  (following the considered order) **do**
- 7: **if**  $r$  is consistent with the rules in  $R_{sel,j}^1$  **then**

- 8:  $N' \leftarrow \max\{\text{number of times that } r \text{ is applicable to } C'_t\}$
- 9: **if**  $N' > 0$  **then**
- 10: **if**  $p_{r,j}(t) = 1$  **then**
- 11:  $n \leftarrow F_b(N', 0.5)$
- 12: **else**
- 13:  $N \leftarrow \max\{\text{number of times that } r \text{ is applicable to } C_t\}$
- 14:  $n \leftarrow F_b(N, p_{r,j}(t))$
- 15: **if**  $n > N'$  **then**
- 16:  $n \leftarrow N'$
- 17: **end if**
- 18: **end if**
- 19: **if**  $n > 0$  **then**
- 20:  $C'_t \leftarrow C'_t - n \cdot l(r)$
- 21:  $R_{sel,j}^1 \leftarrow R_{sel,j}^1 \cup \{ \langle r, n \rangle \}$
- 22: **else**
- 23:  $R_{sel,j}^2 \leftarrow R_{sel,j}^2 \cup \{ \langle r, n \rangle \}$
- 24: **end if**
- 25: **end if**
- 26: **end if**
- 27: **end for**
- 28: **end for**
- 29: **end for**

#### Second phase of rules selection (maximality)

- 1: **for**  $j \leftarrow 1$  to  $m$  **do**
- 2:  $R_{sel,j} \leftarrow R_{sel,j}^1 + R_{sel,j}^2$  with an order by the rule probabilities, from highest to lowest
- 3: **for each**  $\langle r, n \rangle \in R_{sel,j}$  (following the selected order) **do**
- 4: **if**  $n > 0 \vee (r \text{ is consistent with the rules in } R_{sel,j}^1)$  **then**
- 5:  $N' \leftarrow \max\{\text{number of times that } r \text{ is applicable to } C'_t\}$
- 6: **if**  $N' > 0$  **then**
- 7:  $R_{sel,j}^1 \leftarrow R_{sel,j}^1 \cup \{ \langle r, N' \rangle \}$
- 8:  $C'_t \leftarrow C'_t - N' \cdot l(r)$
- 9: **end if**
- 10: **end if**
- 11: **end for**
- 12: **end for**

#### Execution of selected rules

- 1: **for each**  $\langle r, n \rangle \in R_{sel,j}^1$  **do**
- 2:  $C'_t \leftarrow C'_t + n \cdot r(r)$
- 3: Update the electrical charges of  $C'_t$  according to  $r(r)$
- 4: **end for**
- Note 1:  $l(r)$  is the left-hand side of the rule  $r$ , moreover  $r(r)$  is the right-hand side of the rule  $r$ .
- Note 2: Two rules  $r$  and  $r'$  with the same labels and electrical charges for  $l(r)$  and  $l(r')$  are *consistent* if  $r(r)$  and  $r(r')$  have the *same* electrical charges.
- Note 3:  $p_{r,j}(t)$  indicates the probability associated to rule  $r$  for the P system situated in the environment  $j$ , at the moment  $t$ .

- Note 4: The used multisets in this algorithm can contain pairs  $\leftarrow r, 0 \rightarrow$

## VI. SIMULATION RESULTS

The inherent randomness in the dynamics of ecosystems makes it unfeasible to address the formal validation of models that attempt to reproduce their behavior. It is therefore necessary to carry out an experimental validation, by comparing results generated by simulation tools against experimental data obtained directly from the real ecosystem. Moreover, once a model is (experimentally) validated, it is possible to use the software tool to analyze the dynamics of the real-life process for different virtual scenarios that could be interesting for the experts in order to formulate plausible hypotheses.

We have used the P-Lingua [5], [14] programming language and the *pLinguaCore* [5], [15] library in order to simulate the model. P-Lingua is a programming language to define P systems in an easy-to-learn, parametric and modular way, and *pLinguaCore* is a Java library under GNU GPL license [13] which implements several simulation algorithms for P systems. In particular, the current release of *pLinguaCore* includes implementations of the two algorithms described in the previous sections.

A new graphic user interface (GUI) have been developed over *pLinguaCore*. It allows the edition of the initial parameters of the ecosystem and the collection of simulation results. Figures 2- 5 show how the parameters are introduced by using the GUI. This GUI and all the files related to the simulator can be downloaded from <http://www.p-lingua.org/>

In order to compare the two previously described algorithms, we have considered a theoretical ecosystem with three trophic levels composed by 6 species of animals (5 ungulates and 1 carnivore that predates on them). The ecosystem is geographically divided into 10 areas. Figure 2 shows the initial number of animals of each species and the surface of each of the 10 areas (Grass)

Species	Area 1	Area 2	Area 3	Area 4	Area 5	Area 6	Area 7	Area 8	Area 9	Area 10
Grass	5,000	20,000	5,000	15,000	7,000	5,500	4,000	15,000	9,000	22,000
Species 2	500	1,000	200	2,000	1,000	1,250	980	575	580	1,450
Species 3	500	400	100	450	375	400	300	250	375	200
Species 4	500	250	200	150	400	675	225	350	450	200
Species 5	500	600	400	500	780	550	800	775	525	675
Species 6	25	30	50	60	50	25	12	35	75	100
Species 7	2	0	0	0	4	0	0	0	0	0

Fig. 2. Number of animals of each species and grass surface

The following assumptions have been accepted:

- The base diet of ungulates is the grass.
- The carnivorous species feeds on ungulates.
- Grass production depends on environmental conditions.

- When ungulates do not have enough food in the area where they reside at the moment, there is a possibility that they will move to another adjacent area. The carnivore can move to any of the other areas with equal probability.

The ungulates are herbivores, and usually when reproducing, each fertile female has one descendent. However, carnivores in general (as might be the Wolf) have more than one descendent, usually 2-3 or more. Ungulates are supposed to belong to different species having different weights, and therefore the amount of biomass provided by an ungulate for the carnivores to eat should depend on the species. In this hypothetical example and to simplify the problem we have assumed that one carnivore needs ten ungulates each year for survival. The values taken for the biological parameters are shown in Figure 3.

Animal	i	f(i)	k <sub>i,1</sub>	k <sub>i,2</sub>	d(i)
Species 2	2	550	0,75	0,06	1
Species 3	3	2.540	0,75	0,06	1
Species 4	4	1.100	0,75	0,06	1
Species 5	5	600	1	0,06	1
Species 6	6	550	0,75	0,06	1
Species 7	7	10	0,9	0,12	2

Fig. 3. Biological parameters

The base diet of ungulates is the grass produced by the ecosystem itself. Depending on climatic conditions production can vary, and in this example three different situations are considered. The first situation corresponds to a year with low grass production, the second represents the common (average) case, and the third would be a year with production values higher than usual. Each of these three situations may occur randomly each year, according to associated probabilities. Figure 4 shows the values taken in this case.

When there is shortage of resources on a given environment, the model gives to the animals a chance to migrate to another environment. In the case of ungulates, they can only move to adjacent areas (eventually they may stay where they are). On the contrary, carnivorous species are always supposed to move, and they can move to any other area. Figure 5 shows the screen of the simulator with some of these values.

We have simulated the evolution of this ecosystem for 100 years, and for each year 50 repetitions were made. The simulator provides not only graphical output, but also numerical results. The results from the point of view of ecology have been the same irrespectively of the implemented algorithm.

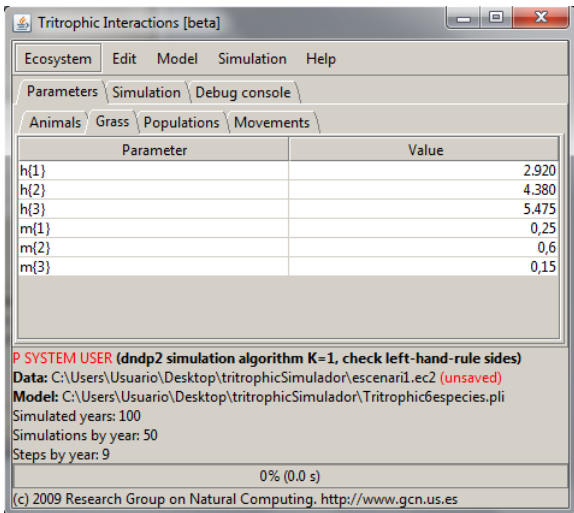


Fig. 4. Parameters related to grass

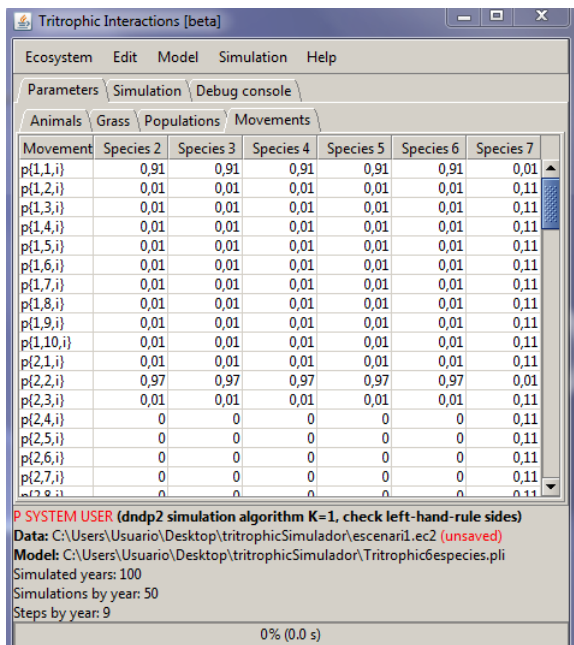


Fig. 5. Probability to move species

Figure 6 shows results for the area 1, where the initial number of individuals was the same for four of the five ungulates. Moreover, for three of these ungulates the same biological parameters were introduced, while in the other case (species 5) the probability of reproduction was higher. This difference is reflected in the number of animals to which each species is stabilized.

In Figure 7 the total number of ungulates and carnivores in the ecosystem are shown. At first, populations of ungulates and carnivores grow, but one can notice that there is a small offset between the years when the maximum of ungulates and carnivores are reached. At some point the necessary amount of ungulates required to feed the carnivores is larger than the actual number of individuals. Then the decline of ungulates

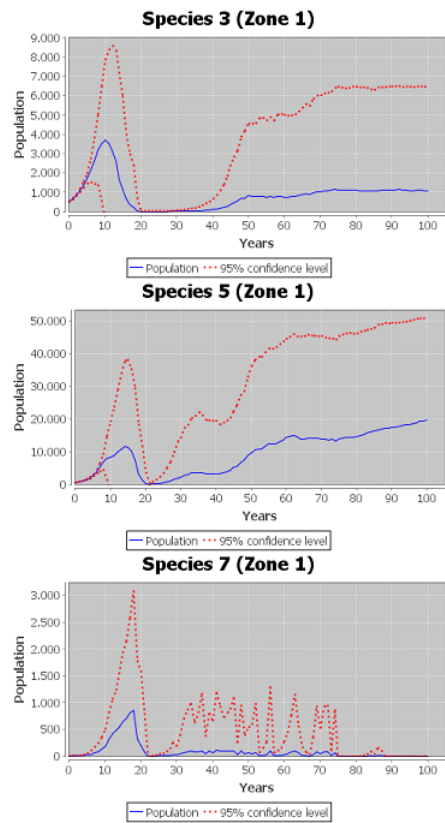


Fig. 6. Evolution of species

starts, and shortly after that of the carnivores, which fails to overcome. Note that carnivores can migrate between different areas in an arbitrary manner when they do not have enough resources, but nevertheless they can go to settings where there is no food and die.

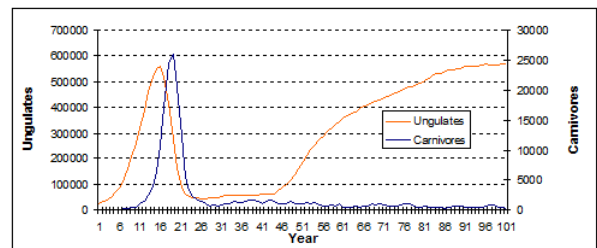


Fig. 7. Total number of ungulates and carnivores for year

Figure 8 shows that between 1919-1923 the number of carnivores exceeds the maximum number of carnivores that could survive according to the available biomass on the ecosystem, and this fact causes their extinction. Obviously, if the number of carnivores decreases to 0, the ungulates will increase their population and thus there will be more biomass on the ecosystem. Consequently the line representing the maximum number of carnivores increases as well.

To study whether the new algorithm, Dndp, improves the previous one, both algorithms were run on two different

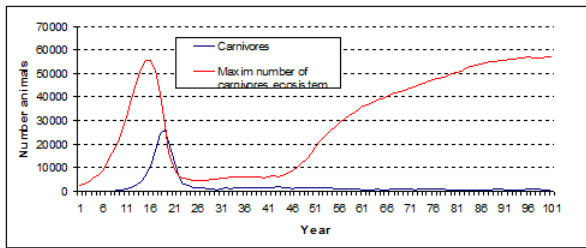


Fig. 8. The pantalla3 GUI

scenarios, measuring the time required to simulate 100 years with 50 repetitions per year. The first scenario consists of 10 environments, but initially there are only animals in area 1. The second scenario used is discussed in the previous results. Table 1 shows the values obtained on seven simulations. On average, Dndp shows a reduction of around 5-6% on the execution time with respect to the Binomial algorithm.

TABLE I  
TIME (SECONDS) OF A 100 YEARS SIMULATION, RUNNING 50  
REPETITIONS PER YEAR

	Scenario 1		Scenario 2	
Algorithm	Binomial	Dndp	Binomial	Dndp
Simulation 1	58,41	54,94	63,62	57,81
Simulation 2	58,57	55,10	61,56	58,58
Simulation 3	58,29	56,05	61,39	57,22
Simulation 4	58,19	56,31	62,81	58,19
Simulation 5	58,75	55,21	61,20	58,75
Simulation 6	57,56	55,19	62,86	57,17
Simulation 7	58,13	54,62	61,92	58,68
<b>Average</b>	58,27	55,35	62,19	58,06
<b>Deviation</b>	0,38	0,61	0,91	0,67

## VII. CONCLUSIONS

The results presented in the previous section show an improvement of a 5-6% of the performance of Dndp algorithm with respect to the Binomial algorithm (as far as the simulation time is concerned). However, the presented tritrophic ecosystem was not designed specifically to be used in comparative studies. In order to draw significant conclusions on the speed-up achieved with Dndp one needs to simulate a more demanding case study.

For instance, in [6] a parallel implementation of the Dndp algorithm is presented, making use of multi-thread programming in Java. In that paper, the algorithm is tested on a P system having 16 environments and 4000 rules, and in this case the obtained improvement grows up to 84% (again, w.r.t. simulation time).

To summarize, this paper presents a virtual ecosystem that tries to balance simplicity and biological relevance. We believe that it was necessary to define a scenario smaller than real case studies (on the size of the alphabet, number of rules, number of parameters, etc) but complex enough to become an illustrative example for the ingredients of the modeling framework (multienvironment functional probabilistic P system with

active membranes) and for the possible simulation algorithms that could be implemented.

On one hand, this hypothetical ecosystem can be used as a benchmark case study for algorithms (and programs) of this kind, as it was illustrated in the paper. On the other hand, despite its simplicity, it has been tailored keeping as close as possible to ecological reality. Thus, it can be very useful as a first case study to introduce researchers from other areas to the modeling framework in membrane computing.

## REFERENCES

- [1] D. Besozzi, P. Cazzaniga, D. Pescini, G. Mauri. Modelling metapopulations with stochastic membrane systems. *BioSystems*, **91**, 3, (2008) 499-514.
- [2] M. Cardona, M. A. Colomer, I. Pérez-Hurtado, M.J. Pérez-Jiménez, D. Sanuy, A. Margalida. A P system Based Model of an Ecosystem of Some Scavenger Birds. *10th International Workshop, WMC 2009. Lecture Notes in Computer Science*, **5957** (2010), 182-195.
- [3] M. Cardona, M.A. Colomer, A. Margalida, A. Palau, I. Pérez-Hurtado, M.J. Pérez-Jiménez, D. Sanuy. A Computational Modeling for real Ecosystems Based on P Systems. *Natural Computing*, online version (<http://dx.doi.org/10.1007/s11047-010-9191-3>).
- [4] J.P. Crampe, J.M. Gaillard, A. Loison. L'enneigement hivernal: un facteur de variation du recrutement chez l'isard (*Rupicapra pyrenaica pyrenaica*). *Canadian Journal of Zoology*, **80** (2002), 306-312.
- [5] M. García-Quismondo, R. Gutiérrez-Escudero, I. Pérez-Hurtado, M.J. Pérez-Jiménez, A. Riscos-Núñez. An overview of P-Lingua 2.0. *10th International Workshop, WMC 2009. Lecture Notes in Computer Science*, **5957** (2010), 264-288.
- [6] M.A. Martínez-del-Amor, I. Pérez-Hurtado, M.J. Pérez-Jiménez, A. Riscos-Núñez and M.A. Colomer. A new simulation algorithm for multienvironment probabilistic P systems, submitted.
- [7] V. Nguyen, D. Kearney, G. Gioiosa. An algorithm for non-deterministic object distribution in P systems and its implementation in hardware. *9th International Workshop, WMC 2008. Lecture Notes in Computer Science*, **5391** (2009), 325-354.
- [8] Gh. Păun, G. Rozenberg, A. Salomaa (eds.) *The Oxford Handbook of Membrane Computing*, Oxford University Press, 2010.
- [9] M.J. Pérez-Jiménez, F.J. Romero. P systems, a new computational modelling tool for Systems Biology. *Transactions on Computational Systems Biology VI. Lecture Notes in Bioinformatics*, **4220**, (2006) 176-197.
- [10] M. Pioz, A. Loison, P. Gibert, D. Dubray, P. Menaut, B. Le Tallec, M. Artois, E. Gilot-Fromont. Transmission of a pestivirus infection in a population of Pyrenean chamois. *Veterinary Microbiology* **119**, (2007) 19-30.
- [11] F.J. Romero, M.J. Pérez-Jiménez. Modelling gene expression control using P systems: The Lac Operon, a case study. *BioSystems*, **91**, 3, (2008) 438-457.
- [12] L. Valencia-Cabrera, I. Pérez-Hurtado, M.J. Pérez-Jiménez, M.A. Colomer. MecoSim3: A Software Simulator for Systems Biology and Population Dynamics based on Membrane Computing. *IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2010)* submitted.
- [13] GPL license  
<http://www.gnu.org/copyleft/gpl.html>
- [14] The P-Lingua website  
<http://www.p-lingua.org/>
- [15] The pLinguaCore library website  
<http://www.p-lingua.org/wiki/index.php/PLinguaCore>