Membrane systems as a rule based modelling framework for multi-compartmental stochastic and discrete systems

Agustín Riscos-Núñez

Research Group on Natural Computing
Department of Computer Science and Artificial Intelligence
University of Seville

June 6-10, Turku, Finland

- P Systems → modeling framework
  - Ecosystems
- Randomness → probabilistic strategies

Simulation algorithms

- Reproduce the behaviour of the models
- Validation
- Virtual experimentation

Software

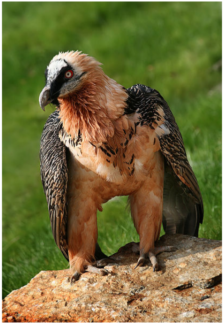
- Implements the algorithms
- GUI for the end-user

Population dynamics

- Complexity of the processes involved
  - Number of species
  - Life cycle
  - Environment
  - Human activities
- Modeling with classical methods
  - Limitations
- Relevance of computational models

Modeling real-life ecosystems

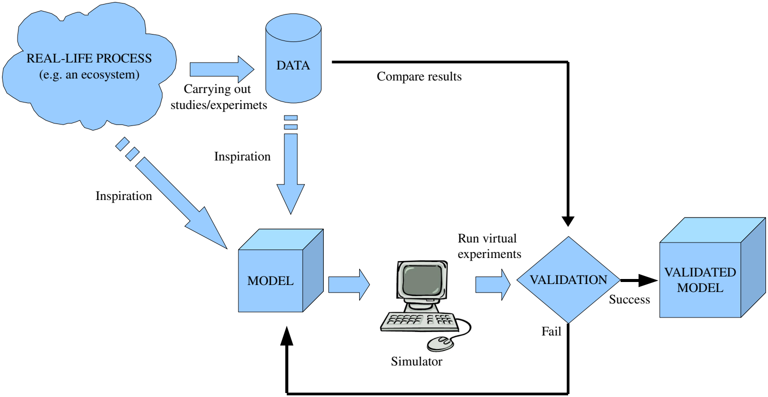
Some studies within the RGNC



- **Modeling Ecosystems using P systems: The Bearded Vulture, a case study**. Cardona et al. *LNCS*, 2009. Vol IV, 137–156.
- **P System Based Model of an Ecosystem of the Scavenger Birds**. Cardona et al. *LNCS*, 2010. Vol IV, 182–195.
- **A Computational Modeling for real Ecosystems based on P systems**. Cardona et al. *Natural Computing*, 2010. on-line version.

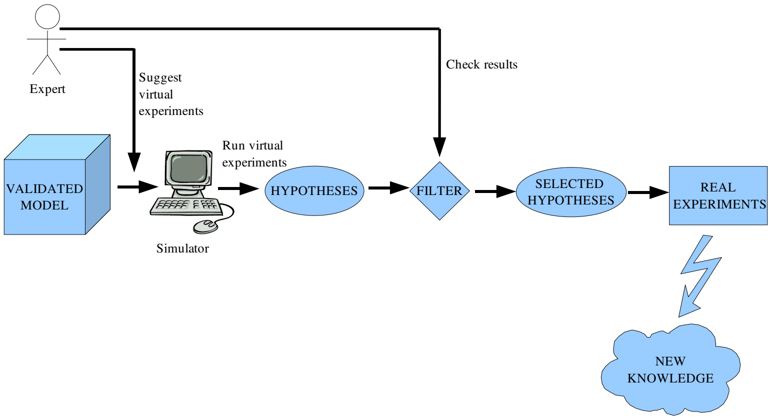
Modeling ecosystems

Validation process



Modeling ecosystems

Virtual Experiments



Need to define a new variant of P Systems

- Cooperation
- Randomness
- Communication between environments
- Membrane polarization

A P system based modeling framework

- A skeleton of an extended P system with active membranes of degree q $\geq$ 1,

$$(\Gamma, \mu, R)$$

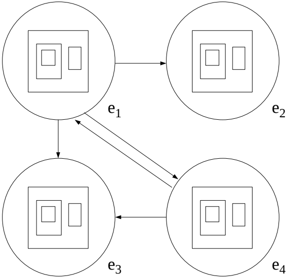
- A probabilistic functional extended P system with active membranes of degree q $\geq$ 1, taking T time units,

$$\Pi = (\Gamma, \mu, R, T, \{f_r : r \in R\}, M_0, \ldots, M_{q-1})$$

- A multienvironment probabilistic functional extended P system with active membranes of degree (m, q) taking T time units,

$$(\Sigma, G, R_E, \Gamma, \mu, R, T, \{f_{rj} : r \in R_\Pi, 1 \leq j \leq m\}, M_{ij} : 0 \leq i \leq q-1, 1 \leq j \leq m)$$

A P system based modeling framework



Skeleton rules

$$u\ [\ v\ ]_h^{\alpha} \xrightarrow{f_r} u'\ [\ v'\ ]_h^{\beta}$$

Environment rules

$$(a)_{e_j} \xrightarrow{f_r} (b)_{e_k}$$

Simulators for P systems

Simulation vs Implementation

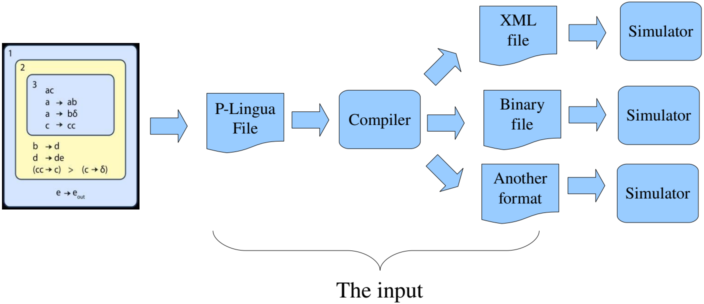
- P systems have not been implemented yet
- It is necessary software/hardware to simulate P system computations

Applications of simulators

- Pedagogical tools
- Support researching in Membrane Computing
- Simulation, validation and virtual experimentation over models of real-life phenomena

Command-line compilation tool

Interoperability

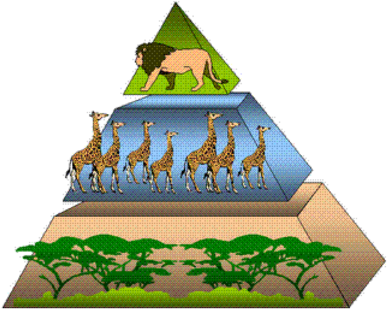


pLinguaCore

Java library for parsing, exporting and simulating

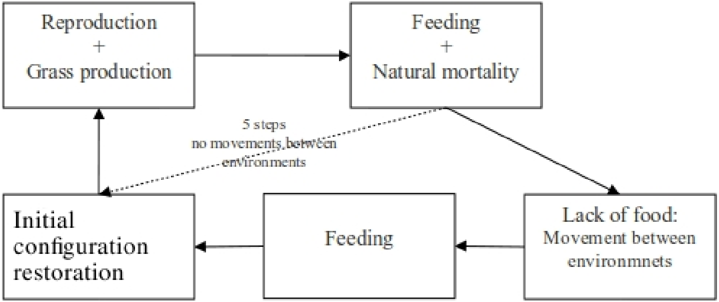
- Free software (GNU GPL license)
- It reads P-Lingua files
- It implements several simulation algorithms
- It exports to other file formats
- Text interface
- It can be used in other Java applications
- It can be extended
- Web page: `http://www.p-lingua.org`

Example: Tritrophic Interactions



- Simplification of a real ecosystem
- Three trophic levels
  - (3) Carnivores
  - (2) Herbivores
  - (1) Grass
- The model
  - 5 modules
  - 9 steps per cycle
  - 10 areas

Tritrophic Interactions

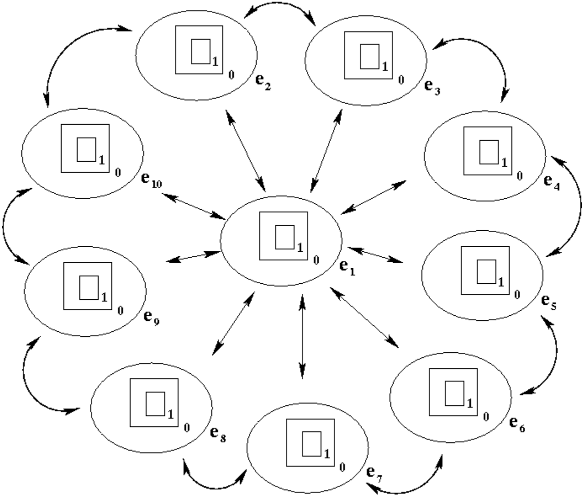


Tritrophic Interactions

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

- $G = (V, S)$
- $\Gamma = \{X_i : 1 \leq i \leq 7\} \cup \{X'_i, Y_i : 2 \leq i \leq 7\} \cup \{R_i : 0 \leq i \leq 6\} \cup G.$
- $\Sigma = \{X_i, X'_i : 2 \leq i \leq 7\}.$
- $\Pi = (\Gamma, [[\,]_1]_0, R_\Pi).$
- $R_E = \{(X_i)_{e_k} \xrightarrow{p_{k,s,i}} (X'_i)_{e_s} : 1 \leq k \leq 10, 1 \leq s \leq 10, 2 \leq i \leq 7\}$
- $\{f_{r,j} : r \in R_\Pi, 1 \leq j \leq 10\}$
- $\{\mathcal{M}_{ij} : 0 \leq i \leq 1, 1 \leq j \leq 10\}$
  - $\mathcal{M}_{0j} = \{X_1^{q_{1,j}}, R_0 : 1 \leq j \leq 10\}.$
  - $\mathcal{M}_{1j} = \{X_2^{q_{2,j}}, \ldots, X_7^{q_{7,j}} : 1 \leq j \leq 10\}.$
- $R = R_E \cup R_\Pi$

Tritrophic Interactions



- Grass production

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

- Females which reproduce and generate d_i offsprings.

  $r_{2,i} \equiv [X_i]_1^0 \xrightarrow{k_{i,1}\cdot 0{,}5} [X_i^{1+d_i}]_1^+, 2 \leq i \leq 7$

- Females and males which don't 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.

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

- Animals which feed and survive.

  $r_{5,i} \equiv [X_i G^{f_i}]_1^+ \xrightarrow{1-k_{i,2}} [Y_i]_1^-,\ 2 \leq i \leq 6$

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

- Animals which feed and don't survive.

  $r_{7,i} \equiv [X_i G^{f_i}]_1^+ \xrightarrow{k_{i,2}} [\ ]_1^-,\ 2 \leq i \leq 6$

  $r_{8,i} \equiv [X_7 X_i^{f_7}]_1^+ \xrightarrow{k_{7,2}} [\ ]_1^-,\ 2 \leq i \leq 6$

- P system synchronization.

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

Tritrophic Interactions

Lack of food: movement between 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.

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

- The object X' goes into the skin membrane.

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

- The object X' goes into the inner membrane.

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

- P system synchronization.

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

resources in the new area $\rightarrow$ possibility to feed and survive.

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

$$r_{17} \equiv [X_7' X'^{f_7}_i]_1^- \xrightarrow{1-k_{7,2}} [Y_7]_1^0, 2 \leq i \leq 6$$

$$r_{18} \equiv [X_7' Y_i^{f_7}]_1^- \xrightarrow{1-k_{7,2}} [Y_7]_1^0, 2 \leq i \leq 6$$

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

$$r_{20} \equiv [X_7' X'^{f_7}_i]_1^- \xrightarrow{k_{7,2}} [\ ]_1^0, 2 \leq i \leq 6$$

$$r_{21} \equiv [X_7' Y_i^{f_7}]_1^- \xrightarrow{k_{7,2}} [\ ]_1^0, 2 \leq i \leq 6$$

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

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

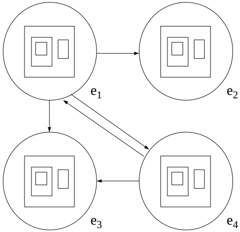
$$r_{24} \equiv [R_6]_1^0 \longrightarrow [R_0]_1^0$$

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

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

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

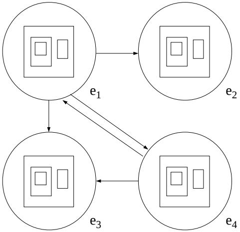
Multienvironment probabilistic P system



Semantics of the system:

- **Binomial block based simulation algorithm**
- **DNDP algorithm**.

Multienvironment probabilistic P system



Semantics of the system:

- **Binomial block based simulation algorithm**
- **DNDP algorithm**.

- Strategy based on the binomial distribution
- Blocks of rules with the same left-hand side
  - Probabilities summing 1
  - Consistent charges in right-hand side
- Each simulation step is composed by

  (1) Selection micro-step
  (2) Execution micro-step

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

- It needs to classify the rules by its 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.

Input: A multienvironment functional P system with active membranes of degree (q, m) with $q \geq 1,\ m \geq 1$, 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**

Simulation algorithms

DNDP Algorithm: Initialization

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

Simulation algorithms

DNDP Algorithm: First selection phase (*consistency*)

for $j \leftarrow 1$ to m **do**
    $R^1_{sel,j} \leftarrow$ *the empty multiset*
    $R^0_{sel,j} \leftarrow$ *the empty multiset*
    for $k \leftarrow 1$ to K **do**
        $D_j \leftarrow A_j \cup B_j$ with a *random order*
        for each $r \in D_j$ (following the considered order) **do**
            if r is *consistent* with the rules in $R^1_{sel,j}$ **then**
                $n \leftarrow$ applications(r,j)
                if $n > 0$ **then**
                    $C'_t \leftarrow C'_t - n \cdot l(r)$
                    $R^1_{sel,j} \leftarrow R^1_{sel,j} \cup \{< r, n >\}$
                else
                    $R^0_{sel,j} \leftarrow R^0_{sel,j} \cup \{< r, n >\}$
                end if
            end if
        end for
    end for
end for

Simulation algorithms

DNDP Algorithm: First selection phase (*applications* function)

$n \leftarrow 0$
$N' \leftarrow$ max{number of times that r is applicable to C'_t}
if $N' > 0$ **then**
    if $p_{r,j}(t) = 1$ **then**
        $n \leftarrow F_b(N', 0{,}5)$
    else
        $N \leftarrow$ max{number of times that r is applicable to C_t}
        $n \leftarrow F_b(N, p_{r,j}(t))$
        if $n > N'$ **then**
            $n \leftarrow N'$
        end if
    end if
end if
return n

First selection phase: comments

In order for a rule r to be applicable

- It must be consistent with the set R_j of previously selected rules.
- The number of possibles aplications into C'_t must be $M > 0$.

Then

- A number of applications n is obtained for r according to the probability function (binomial distribution).
- Such a number n cannot be greater than M.
- Even if $n = 0$, the rule is added to R_j, with associated multiplicity 0.

We denote

- $R_j^0 = \{r \in D_j : \langle r, 0\rangle \in R_j\}$.
- $R_j^1 = \{r \in D_j : \langle r, n\rangle \in R_j, \ n > 0\}$.

C'_t is obtained from C_t by eliminating the corresponding left-hand sides of the selected rules (with their multiplicity).

Simulation algorithms

DNDP Algorithm: Second phase of rules selection (*maximality*)

for $j \leftarrow 1$ to m **do**
    $R_{sel,j} \leftarrow R^1_{sel,j} + R^0_{sel,j}$ *with an order by the rule probabilities, from highest to lowest*
    for each $< r, n > \in R_{sel,j}$ (following the selected order) **do**
        if $n > 0 \vee$ (r is *consistent* with the rules in $R^1_{sel,j}$) **then**
            $N' \leftarrow \max\{$number of times that r is applicable to $C'_t\}$
            if $N' > 0$ **then**
                $R^1_{sel,j} \leftarrow R^1_{sel,j} \cup \{< r, N' >\}$
                $C'_t \leftarrow C'_t - N' \cdot l(r)$
            end if
        end if
    end for
end for

Second phase of rules selection: comments

All rules from R_j selected in the previous phase are checked again in order to get maximality.

- We take over R_j a decreasing order looking at the probabilities of the rules.
- If $\langle r, n \rangle \in R_j$, then we apply it the maximum number of times possible $M > 0$ (in the new C'_t).
- C'_t is updated.

for each $< r, n > \in R^1_{sel,j}$ **do**
  $C'_t \leftarrow C'_t + n \cdot r(r)$
  Update the electrical charges of C'_t according to $r(r)$
end for

Comments:

- All rules from R^1_j will be applied to C'_t, meaning that we just add their right-hand sides.
- The obtained configuration will be C_{t+1}.

- P-Lingua: programming language to define P systems

  `http://www.p-lingua.org`

- pLinguaCore: Java library $\rightarrow$ P-Lingua parser + simulation algorithms
- A specific Java GUI over pLinguaCore
  - Input
    - Initial ecosystem parameters
    - Number of years (complete cycles) to simulate
    - Number of simulations per year
  - Output
    - Evolution of the populations
    - Tables and graphs

Simulation results

Number of animals of each species and grass surface

Tritrophic Interactions [beta]

Ecosystem Edit Model Simulation Help

Parameters | Simulation | Debug console

Animals | Grass | Populations | Movements

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

P SYSTEM USER **(dndp2 simulation algorithm K=1, check left-hand-rule sides)**

Data: C:\Users\Usuario\Desktop\tritrophicSimulador\escenari1.ec2 (unsaved)

Model: C:\Users\Usuario\Desktop\tritrophicSimulador\Tritrophic6especies.pli

Simulated years: 100

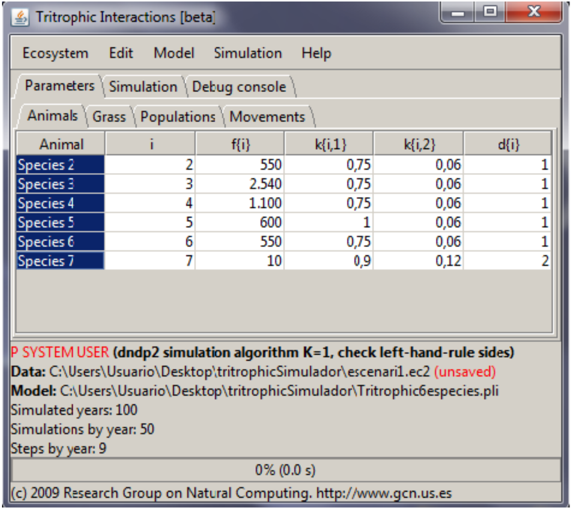
Simulations by year: 50

Steps by year: 9

0% (0.0 s)

Simulation results

Biological parameters

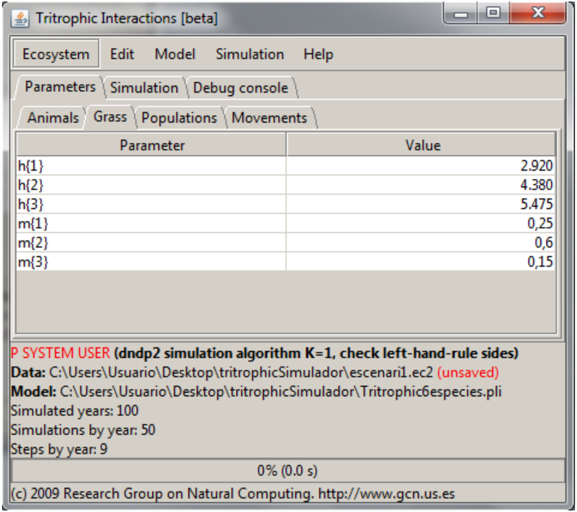


Animal	i	f{i}	k{i,1}	k{i,2}	d{i}
Species 2	2	550	0,75	0,06	1
Species 3	3	2.540	0,75	0,06	1
Species 4	4	1.100	0,75	0,06	1
Species 5	5	600	1	0,06	1
Species 6	6	550	0,75	0,06	1
Species 7	7	10	0,9	0,12	2

P SYSTEM USER **(dndp2 simulation algorithm K=1, check left-hand-rule sides)**
Data: C:\Users\Usuario\Desktop\tritrophicSimulador\escenari1.ec2 (unsaved)
Model: C:\Users\Usuario\Desktop\tritrophicSimulador\Tritrophic6especies.pli
Simulated years: 100
Simulations by year: 50
Steps by year: 9
0% (0.0 s)
(c) 2009 Research Group on Natural Computing. http://www.gcn.us.es

Simulation results

Parameters related to grass



Simulation results

Probabilities of species movement

Tritrophic Interactions [beta]

Ecosystem Edit Model Simulation Help

Parameters | Simulation | Debug console

Animals | Grass | Populations | Movements

Movement	Species 2	Species 3	Species 4	Species 5	Species 6	Species 7
p{1,1,i}	0,91	0,91	0,91	0,91	0,91	0,01
p{1,2,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{1,3,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{1,4,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{1,5,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{1,6,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{1,7,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{1,8,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{1,9,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{1,10,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{2,1,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{2,2,i}	0,97	0,97	0,97	0,97	0,97	0,01
p{2,3,i}	0,01	0,01	0,01	0,01	0,01	0,11
p{2,4,i}	0	0	0	0	0	0,11
p{2,5,i}	0	0	0	0	0	0,11
p{2,6,i}	0	0	0	0	0	0,11
p{2,7,i}	0	0	0	0	0	0,11
p{2,8,i}	0	0	0	0	0	0,11

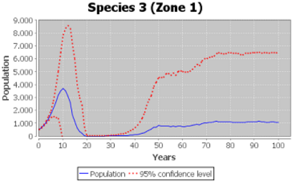
P SYSTEM USER **(dndp2 simulation algorithm K=1, check left-hand-rule sides)**

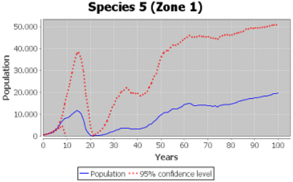
Data: C:\Users\Usuario\Desktop\tritrophicSimulador\escenari1.ec2 (unsaved)

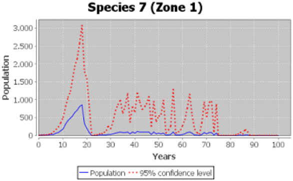
Model: C:\Users\Usuario\Desktop\tritrophicSimulador\Tritrophic6especies.pli

Simulated years: 100

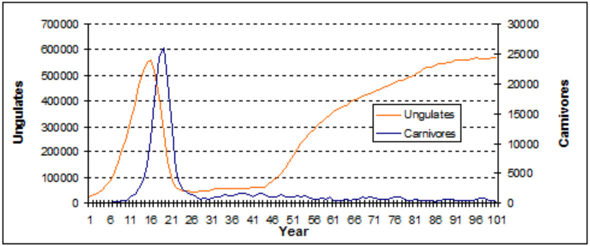
Simulation results



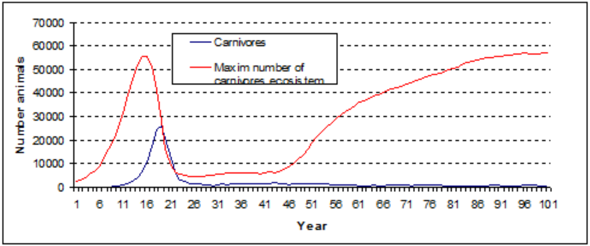




Simulation results



Simulation results



Simulation results

	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
Average	58,27	55,35	62,19	58,06
Deviation	0,38	0,61	0,91	0,67

Conclusions

- P systems provide a high-level modeling framework for ecosystems
- Software tools based on membrane computing can be used to carry out virtual experiments
  - Key role of simulation algorithms
- A software framework based on P-Lingua has been provided
- A virtual ecosystem has been used as an example

Ongoing / Future work

- Design new simulation algorithms
- Develop simulators based on High Performance Computing (GPUs)
- Design a common protocol to communicate simulators and user interfaces
  - Using different platforms for simulators
  - Codyfing P systems on a standard format file
- Extend the software framework to cover more types of P systems
- Design more efficient and standard GUIs for final users