Modelling real-life phenomena

Mario J. Pérez Jiménez

Research Group on Natural Computing
Dpt. Computer Science and Artificial Intelligence
University of Seville, Spain
marper@us.es

New Frontiers in Informatics – Barcelona, July 18, 2014

Outline

- Mathematical models.
- Classical approaches.
- Membrane Computing as a bioinspired computing modelling framework.
  - Stochastic approach: **Multicompartmental P systems**.
  - Probabilistic approach: **Population Dynamics P systems**.
- Applications

Mathematical models

Mathematical models

- Abstractions of the real world onto a mathematical domain.

Mathematical models

- Abstractions of the real world onto a mathematical domain.
- Useful in *what if* studies.
- A modern tool for scientific investigation.
- It allows us to analyze and predict.

Mathematical models

- Abstractions of the real world onto a mathematical domain.
- Useful in *what if* studies.
- A modern tool for scientific investigation.
- It allows us to analyze and predict.

Properties of a "good" mathematical model

- *Relevance*.

Mathematical models

- Abstractions of the real world onto a mathematical domain.
- Useful in *what if* studies.
- A modern tool for scientific investigation.
- It allows us to analyze and predict.

Properties of a "good" mathematical model

- *Relevance*.
- *Understandability*.

Mathematical models

- Abstractions of the real world onto a mathematical domain.
- Useful in *what if* studies.
- A modern tool for scientific investigation.
- It allows us to analyze and predict.

Properties of a "good" mathematical model

- *Relevance*.
- *Understandability*.
- *Extensibility*.

Mathematical models

- Abstractions of the real world onto a mathematical domain.
- Useful in *what if* studies.
- A modern tool for scientific investigation.
- It allows us to analyze and predict.

Properties of a "good" mathematical model

- *Relevance*.
- *Understandability*.
- *Extensibility*.
- *Computability and Mathematical tractability*.

RGNC

Classical approach

- Modelling based on ordinary/partial differential equations (ODEs/PDEs)
  At cellular level, it is based on two assumptions:

  1. **Cells** are assumed to be well stirred and homogeneous volumes so that **concentrations do not change with respect to space**.
  2. **Chemical concentrations** vary continuously over time in a deterministic way.

Classical approach

- Modelling based on ordinary/partial differential equations (ODEs/PDEs)
  At cellular level, it is based on two assumptions:

  1. **Cells** are assumed to be well stirred and homogeneous volumes so that **concentrations do not change with respect to space**.
  2. **Chemical concentrations** vary continuously over time in a deterministic way.

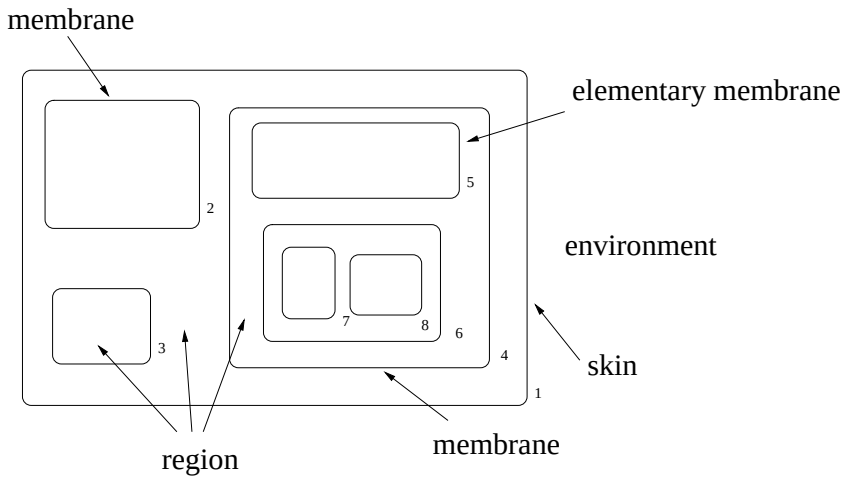
- Many computational frameworks have been used to model cellular systems like Petri nets, process algebra, π–calculus, agents, etc.

Membrane Computing (Gh. Păun, 1998)

Basic *P systems*.

Syntactical ingredients:

1. A cell-like membrane structure: a rooted tree.
2. Multisets of objects and strings placed inside the compartments delimited by membranes.
3. Rewriting rules associated with specific compartments.

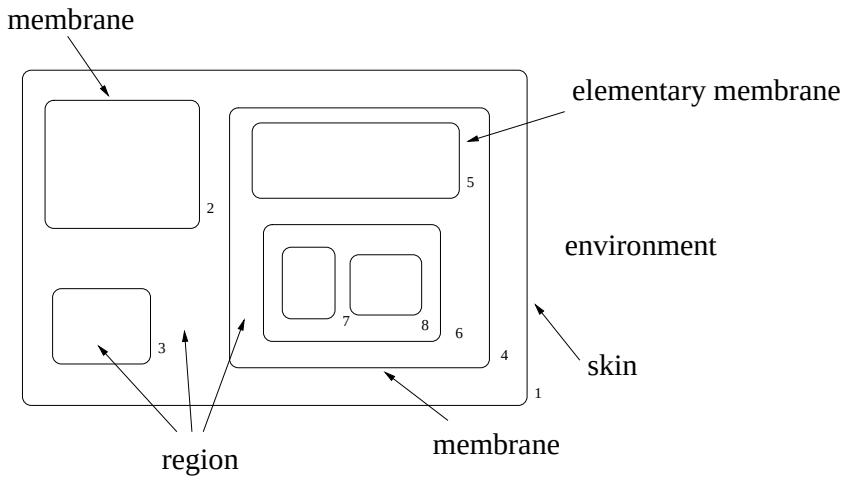


Membrane Computing (Gh. Păun, 1998)

Basic *P systems*.

Syntactical ingredients:

1. A cell-like membrane structure: a rooted tree.
2. Multisets of objects and strings placed inside the compartments delimited by membranes.
3. Rewriting rules associated with specific compartments.



Semantics ingredients:

- Configuration + Transition step + Computation.
- Non-determinism and maximall parallelism.

Multienvironment P systems (I)

A multienvironment P system of degree (m, n, q) taking T time units: $(G, \Gamma, \Sigma, T, \mathcal{R}_E, \mu, \mathcal{R}, \Pi_1, \ldots, \Pi_n)$

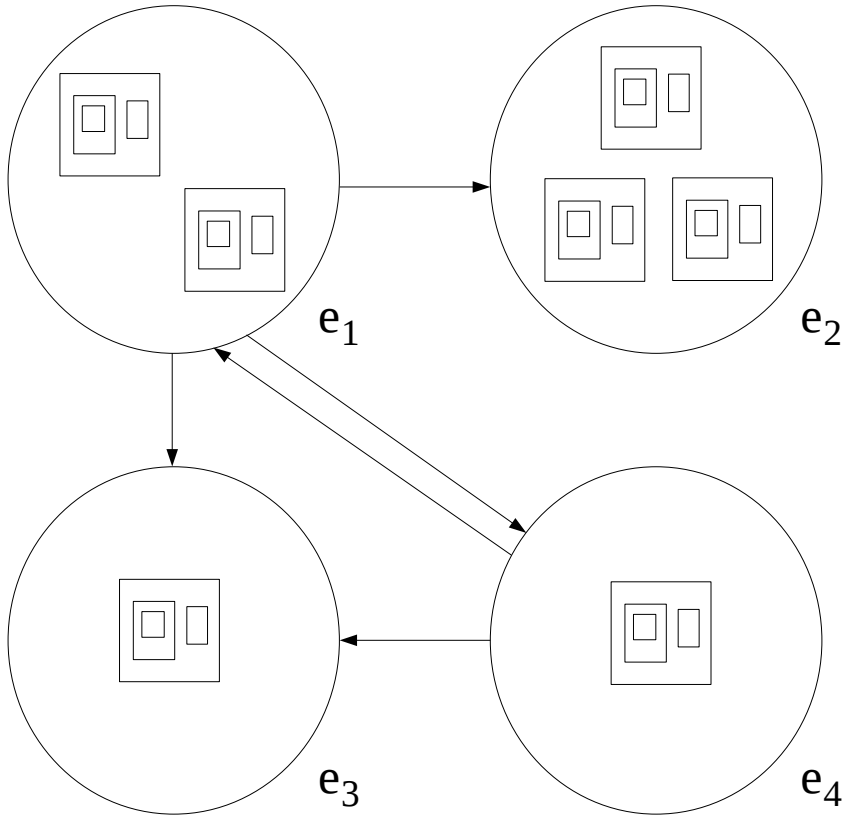
- $G = (V, S)$ is a directed graph. Let $V = \{e_1, \ldots, e_m\}$ whose elements are called environments;
- Γ is the working alphabet and $\Sigma \subsetneqq \Gamma$.
- T is a natural number that represents the simulation time of the system;
- $\mathcal{R}_E$ is a finite set of communication rules between environments of the following forms

$$(x)_{e_j} \longrightarrow (y_1)_{e_{j_1}} \ldots (y_h)_{e_{j_h}} \quad \text{and} \quad (\Pi_k)_{e_j} \longrightarrow (\Pi_k)_{e_{j'}}$$

- μ is a rooted tree with q nodes.
- $\mathcal{R}$ is a finite set of rules of the type $u[v]_i^{\alpha} \longrightarrow u'[v']_i^{\beta}$
- No rules from $\mathcal{R}$ and $\mathcal{R}_E$ compete for objects.
- $\Pi_k = (\Gamma, \mu, \mathcal{M}_{1,k}, \ldots, \mathcal{M}_{q,k}, \mathcal{R})$ is a basic P system of degree q.
- Each rule of the system has associated a computable function whose domain is $\{0, \ldots, T\}$.

Multienvironment P systems (II)

- A set of m environments.
- A set of n basic P systems (with the same skeleton).
- A set of communication rules among environments.
- Each rule of the system has associated a computable function (depending on the environment).



Multienvironment P systems versus ODEs/PDEs

- Advantages of multienvironment P systems with respect to ODEs/PDEs.

Multienvironment P systems versus ODEs/PDEs

- Advantages of multienvironment P systems with respect to ODEs/PDEs.
  - ⋆ They use a language closer to experts than ODEs/PDEs

Multienvironment P systems versus ODEs/PDEs

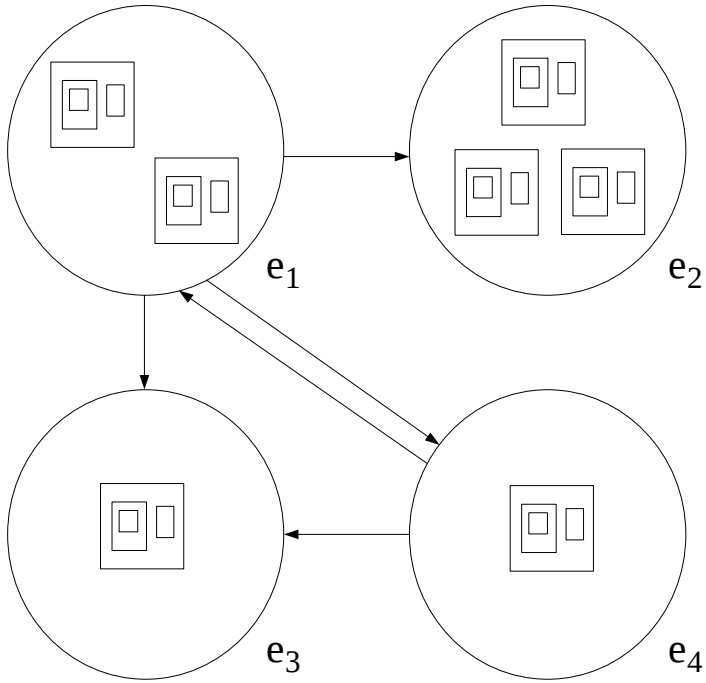
- Advantages of multienvironment P systems with respect to ODEs/PDEs.
  - ⋆ They use a language closer to experts than ODEs/PDEs
  - ⋆ They are not affected by the usual constraints present when defining ODEs/PDEs based models.

Multienvironment P systems versus ODEs/PDEs

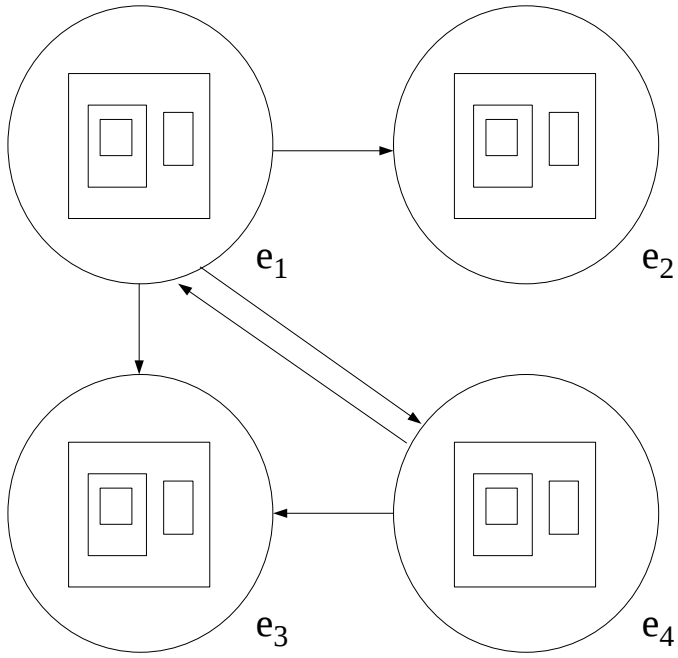
- Advantages of multienvironment P systems with respect to ODEs/PDEs.
  - ⋆ They use a language closer to experts than ODEs/PDEs
  - ⋆ They are not affected by the usual constraints present when defining ODEs/PDEs based models.
  - ⋆ They are modular:
    - Small changes in the system $\rightarrow$ small changes in the model.
    - When using ODEs/PDEs most of times we have to start from scratch.

Stochastic approach: Multicompartmental P systems

- The computable functions associated with the rules are **propensities**.
- Initially, the basic P systems are randomly distributed among the environments of the system.



Probabilistic approach: Population Dynamics P systems



- The computable functions associated with the rules are *probabilities*.
- Initially, each environment contains exactly a basic P system with the same structure.
- There are only rules among environments of the form $(x)_{e_j} \xrightarrow{pr} (y_1)_{e_{j_1}} \cdots (y_h)_{e_{j_h}}$

A Semantics for Multicompartmental P Systems

Our strategy will be based on Gillespie theory of stochastic kinetics.

Classical Gillespie Algorithm

Input: A well mixed and fixed volume (m substances subected to chemical reactions $r_1, \ldots, r_q$).

1. Compute for each rule in r_j its propensity, p_j,
2. Compute the sum of all propensities: $p_0 = \sum_{j=1}^{q} p_j$.
3. Generate two random numbers r_1 and r_2 from the uniform distribution in the unit-interval.
4. Compute the waiting time for the next reacion $\tau = \frac{1}{p_0} \ln\left(\frac{1}{r_1}\right)$
5. Select number j_0 that verifies $\sum_{k=1}^{j_0-1} p_k < r_2 \cdot p_0 \leq \sum_{k=1}^{j_0} p_k$.

Output: The next reaction to be applied and the waiting time for this application.

Multicompartmental Gillespie Algorithm

Input: A multicompartmental P system.

- **Initialization**
  - set time of the simulation $t = 0$;
  - for each membrane i compute a pair (t_i, r_{j_i}) by using the Gillespie algorithm;
  - construct a list containing all such pairs;
  - sort this list in increasing order according to t_i;
- **Iteration**
  - extract the first pair, $(t_{i_0}, r_{j_{i_0}})$ from the list;
  - set time of the simulation $t = t + t_{i_0}$;
  - update the waiting time for the rest of the triples in the list by subtracting t_{i_0};
  - apply the rule $r_{j_{i_0}}$ in membrane i only once;
  - for each membrane i' affected by the application of the rule remove the corresponding pair $(t_{i'}, r_{j_{i'}})$ from the list;
  - for each membrane i' affected by the application of the rule $r_{j_{i_0}}$ re-run the Gillespie algorithm for the new context in i' to obtain $(t'_{i'}, r_{j'_{i'}})$;
  - add the new pairs $(t'_{i'}, r_{j'_{i'}})$ in the list and sort this list according to each waiting time and iterate the process.
- **Termination**
  - Terminate simulation when time of the simulation t reaches or exceeds a preset maximal time of simulation.

Output: The next reaction to be applied and the waiting time for this application.

A Semantics for PDP systems: DNDP algorithm (I)

Direct non-deterministic distribution algorithm with probabilities (DNDP)

Input: A PDP system of degree (m, n, q) taking T time units, $T \geq 1$.

$C_0 \leftarrow$ initial configuration of the system

for $t \leftarrow 0$ to $T - 1$ **do**

$C'_t \leftarrow C_t$

Initialization

First selection phase: generates a multiset of *consistent* applicable rules.

Second selection phase: generates a multiset of *maximal consistent* applicable rules.

Execution of selected rules.

$C_{t+1} \leftarrow C'_t$

end for

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
  $LC_j \leftarrow$ 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 $\langle h, \alpha \rangle \in LC_j$ (following the considered order) **do**
    $B_j \leftarrow B_j \cup$ ordered set of rules $u[v]_h^\alpha \rightarrow u'[v']_h^\beta$ from R_Π whose probability at the moment t is greater than 0 for the environment j
  end for
end for

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
  $LC_j \leftarrow$ 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 $\langle h, \alpha \rangle \in LC_j$ (following the considered order) **do**
    $B_j \leftarrow B_j \cup$ ordered set of rules $u[v]_h^\alpha \rightarrow u'[v']_h^\beta$ from R_Π whose probability at the moment t is greater than 0 for the environment j
  end for
end for

First selection phase (*consistency*)

for $j \leftarrow 1$ to m **do**
  $R_j \leftarrow$ *the empty multiset*
  $D_j \leftarrow A_j \cup B_j$ with a *random order*
  for each $r \in D_j$ (following the considered order) **do**
    $M \leftarrow$ maximum number of times that r is applicable to C'_t
    if r is *consistent* with the rules in $R_j^1 \wedge M > 0$ **then**
      $N \leftarrow$ maximum number of times that r is applicable to C_t
      $n \leftarrow \min\{M, F_b(N, p_{r,j}(t))\}$
      $C'_t \leftarrow C'_t - n \cdot LHS(r)$
      $R_j \leftarrow R_j \cup \{< r, n >\}$
    end if
  end if
end for

Second selection phase (*maximality*)

for $j \leftarrow 1$ to m **do**
  $R_j \leftarrow R_j$ *with an order by the rule probabilities, from highest to lowest*
  for each $< r, n > \in R_j$ (following the selected order) **do**
    if $n > 0 \vee (r$ is *consistent* with the rules in $R_j^1)$ **then**
      $M \leftarrow$ maximum number of times that r is applicable to C_t'
      if $M > 0$ **then**
        $R_j \leftarrow R_j \cup \{< r, M >\}$
        $C_t' \leftarrow C_t' - M \cdot LHS(r)$
      end if
    end if
  end for
end for

Second selection phase (*maximality*)

for $j \leftarrow 1$ to m **do**
  $R_j \leftarrow R_j$ *with an order by the rule probabilities, from highest to lowest*
  for each $< r, n > \in R_j$ (following the selected order) **do**
    if $n > 0 \vee (r$ is *consistent* with the rules in $R_j^1)$ **then**
      $M \leftarrow$ maximum number of times that r is applicable to C_t'
      if $M > 0$ **then**
        $R_j \leftarrow R_j \cup \{< r, M >\}$
        $C_t' \leftarrow C_t' - M \cdot LHS(r)$
      end if
    end if
  end for
end for

Execution of selected rules

for each $< r, n > \in R_j$, $n > 0$ **do**

  $C_t' \leftarrow C_t' + n \cdot RHS(r)$

  Update the electrical charges of C_t' according to $RHS(r)$

end for

Applications of Multicompartmental P systems

- Signalling pathways:
  - Epimermal Growth Factor Receptor.
  - FAS-induced apoptosis.

Applications of Multicompartmental P systems

- Signalling pathways:
  - Epimermal Growth Factor Receptor.
  - FAS-induced apoptosis.
- Gene expression control in Lac Operon.

Applications of Multicompartmental P systems

- Signalling pathways:
  - Epimermal Growth Factor Receptor.
  - FAS-induced apoptosis.
- Gene expression control in Lac Operon.
- Quorum sensing in Vibrio Fischeri.

Applications of PDP systems

- Population dynamics of ecosystems with ungulates and scavengers:
  - ⋆ Catalan Pyrenees (Spain): 14 species.
  - ⋆ Navarra (Spain): 10 species.
  - ⋆ Swaziland (South Africa): 30 species.

Applications of PDP systems

- Population dynamics of ecosystems with ungulates and scavengers:
  - ⋆ Catalan Pyrenees (Spain): 14 species.
  - ⋆ Navarra (Spain): 10 species.
  - ⋆ Swaziland (South Africa): 30 species.
- Population dynamics of the zebra mussel in the dam of Ribaroja (Spain).

Applications of PDP systems

- Population dynamics of ecosystems with ungulates and scavengers:
  - ⋆ Catalan Pyrenees (Spain): 14 species.
  - ⋆ Navarra (Spain): 10 species.
  - ⋆ Swaziland (South Africa): 30 species.
- Population dynamics of the zebra mussel in the dam of Ribaroja (Spain).
- Pyrenean newt population dynamics in the Sierra del Cadí (Spain).
- The Hazel Grouse reintroduction in the Pyrenees (Spain).
- Effect of pesti-virus dynamics chamois (Spain).
- Development and growth of amphibians in natural ponds (Spain).
- Effect of management in the whole process of raising pigs (Spain).

Applications of PDP systems

- Population dynamics of ecosystems with ungulates and scavengers:
  - ⋆ Catalan Pyrenees (Spain): 14 species.
  - ⋆ Navarra (Spain): 10 species.
  - ⋆ Swaziland (South Africa): 30 species.
- Population dynamics of the zebra mussel in the dam of Ribaroja (Spain).
- Pyrenean newt population dynamics in the Sierra del Cadí (Spain).
- The Hazel Grouse reintroduction in the Pyrenees (Spain).
- Effect of pesti-virus dynamics chamois (Spain).
- Development and growth of amphibians in natural ponds (Spain).
- Effect of management in the whole process of raising pigs (Spain).
- Logic networks (special classes of gene regulatory network)

Some publications

- J.M. Cecilia, J.M. García, G.D. Guerrero, M.A. Martínez, I. Pérez-Hurtado, M.J. Pérez-Jiménez. Simulation of P systems with active membranes on CUDA. Briefings in Bioinformatics, 11, 3 (2010), 313-322.
- J.M. Cecilia, J.M. García, G.D. Guerrero, M.A. Martínez, M.J. Pérez-Jiménez, M. Ujaldón. The GPU on the simulation of cellular computing models. SoftComputing, 16, 2 (2012), 231-246.
- M.A. Colomer, A. Margalida, M.J. Pérez-Jiménez. Population Dynamics P System (PDP) Models: A Standardized Protocol for Describing and Applying Novel Bio-Inspired Computing Tools. PLOS ONE, **8**, 4:e60698 (2013), 1-13 (doi: 10.1371/journal.pone.0060698).
- M.A. Colomer, A. Margalida, D. Sanuy, M.J. Pérez-Jiménez. A bio-inspired computing model as a new tool for modeling ecosystems: The avian scavengers as a case study. Ecological modelling, **222**, 1 (2011), 33-47.
- M.A. Colomer, I. Pérez-Hurtado, M.J. Pérez-Jiménez, A. Riscos. Comparing simulation algorithms for multienvironment probabilistic P system over a standard virtual ecosystem. Natural Computing, **11** (2012), 369-379.
- M.J. Pérez-Jiménez, F.J. Romero. A study of the robustness of the EGFR signalling cascade using continuous membrane systems. Lecture Notes in Computer Science, 3561 (2005), 268-278.
- 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.
- 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.
- F.J. Romero, M.J. Pérez-Jiménez. A model of the Quorum Sensing System in Vibrio Fischeri using P systems. Artificial Life, 14, 1 (2008), 95-109.
- L. Valencia-Cabrera, M. García-Quismondo, M.J. Pérez-Jiménez, Y. Su, H. Yu, L. Pan. Modeling Logic Gene Networks by Means of Probabilistic Dynamic P Systems. International Journal of Unconventional Computing, 9, 5-6 (2013), 445-464.
- G. Zhang, M. Gheorghe, L. Pan, M.J. Pérez-Jiménez. Evolutionary membrane computing: a comprehensive survey and new results. Information Sciences, 279 (2014), 528-551

THANK YOU

FOR YOUR ATTENTION!

Quorum sensing in Vibrio Fischeri (I)

- Bacteria are generally considered to be independent unicellular organisms.

Quorum sensing in Vibrio Fischeri (I)

- Bacteria are generally considered to be independent unicellular organisms.
- *Vibrio fischeri* exists naturally either in a planktonic state or as a symbiont of certain luminescent squid.

Quorum sensing in Vibrio Fischeri (I)

- Bacteria are generally considered to be independent unicellular organisms.
- *Vibrio fischeri* exists naturally either in a planktonic state or as a symbiont of certain luminescent squid.

- *Vibrio fischeri* exhibit coordinated behaviour which allows an entire population of bacteria to regulate the expression of certain or specific genes in a coordinated way depending on the size of the population.
- *Quorum sensing*: cell density dependent gene regulation system.
- This phenomenon was first investigated in the marine bacterium *Vibrio fischeri*.

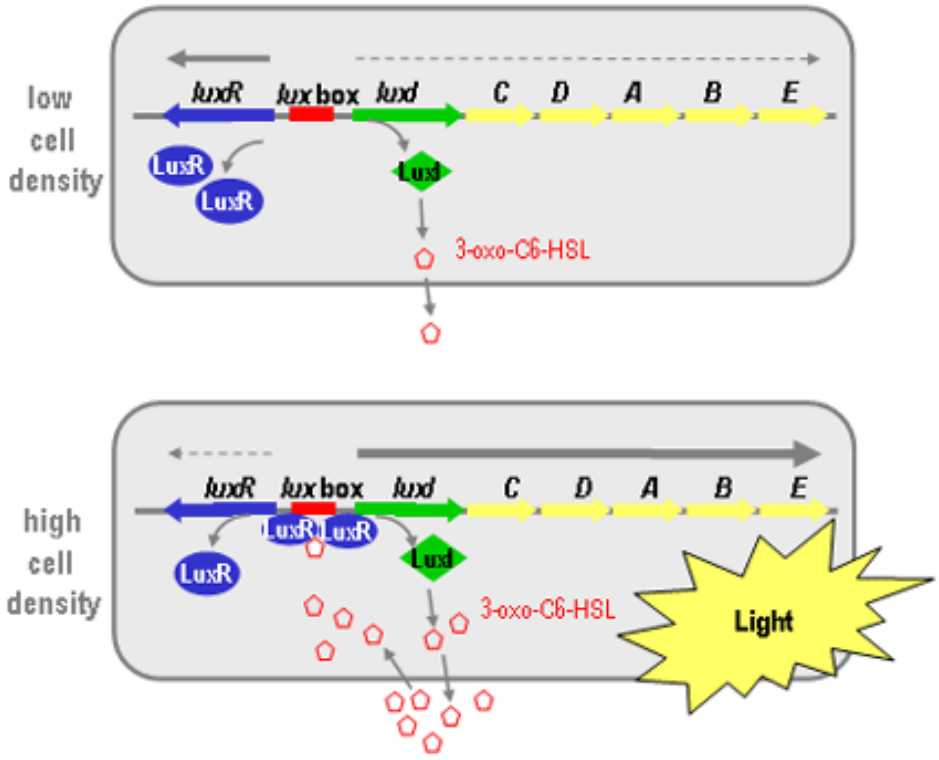
Quorum sensing in Vibrio Fischeri (I)

- Bacteria are generally considered to be independent unicellular organisms.
- *Vibrio fischeri* exists naturally either in a planktonic state or as a symbiont of certain luminescent squid.

- *Vibrio fischeri* exhibit coordinated behaviour which allows an entire population of bacteria to regulate the expression of certain or specific genes in a coordinated way depending on the size of the population.
- *Quorum sensing*: cell density dependent gene regulation system.
- This phenomenon was first investigated in the marine bacterium *Vibrio fischeri*.
- The bacteria colonise specialised light organs in the squid which cause it to luminesce.
- The bacteria only luminesce when colonising the light organs and do not emit light when in the free-living state
- Luminescence in the squid is involved in the attraction of prey, camouflage and communication between different individuals.

Molecular mechanisms of Quorum sensing

(K.H. Nealson y J.W. Hasting, 1979; K.L Visic et al., 2000)



- The process start when **Lux Box** produces proteins **LuxR** and **LuxI** at low/basal level.
- Protein **LuxI** transcribes the signal **OHHL**.
- Signals **OHHL** diffuse out of the bacterial cells and into the surrounding environment.
- At high cell density, the signal is able to interact with the **LuxR** protein to form the complex **LuxR-OHHL**.
- This complex blinds with **Lux Box** making it produces **LuxR** and **LuxI** at high level.
- Complex **LuxR-OHHL** causes the transcription of the luminescence genes: a cluster of 5 genes, **luxCDABE**.

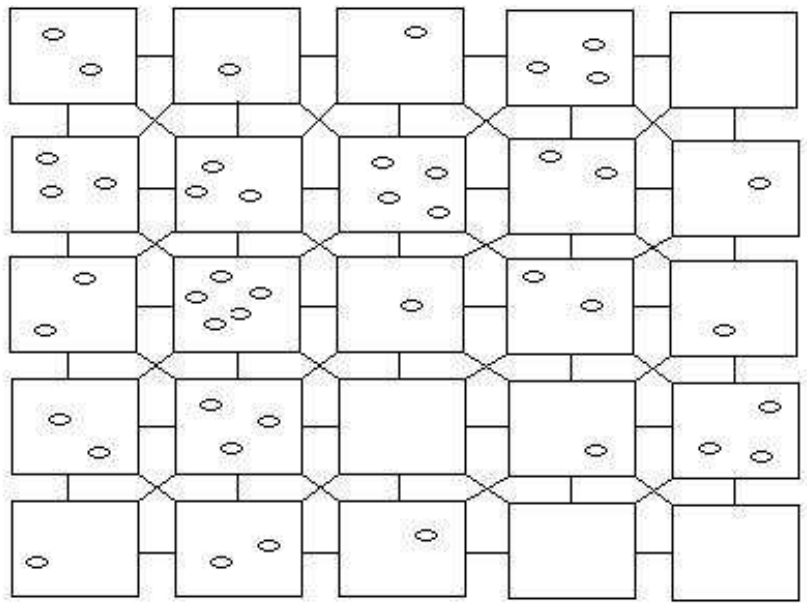
Modelling Quorum Sensing in Vibrio fischeri (I)

We study the behaviour of a population of N bacteria placed inside a multienvironmen P system of degree $(25, 1, N)$.

$$\mathbf{ME} = (G, \Gamma, \Sigma, T, \mathcal{R}_E, \mu, \mathcal{R}, \Pi_1, \ldots, \Pi_N)$$

where:

- $G = (V = \{e_1, \ldots, e_{25}\}, S)$ is the following directed graph.



Modelling Quorum Sensing in Vibrio fischeri (II)

- $\Gamma = \{\text{LuxR},\ \text{LuxR.OHHL},\ \text{LuxBox},\ \text{LuxR.OHHL.LuxBox},\ \text{OHHL}\}$.
- $\Sigma = \{\text{OHHL}\}$.
- $T \geq 1$.
- Rules from $\mathcal{R}_E$.

$$r_1 : (\ \text{OHHL}\ )_{e_i} \xrightarrow{c_1} (\ \ )_{e_i}$$

$$r_2 : (\ \text{OHHL}\ )_{e_i} \xrightarrow{c_2} (\ \text{OHHL}\ )_{e_j}$$

$$r_3 : (\ \Pi_k\ )_{e_i} \xrightarrow{c_3} [\ \Pi_k\ )_{e_j}$$

- $\mu = [\ \ ]_b$.

Modelling Quorum Sensing in Vibrio fischeri (III)

- Rules from $\mathcal{R}$:

$$
\begin{aligned}
r_4 &: \text{OHHL}\,[\ ]_b \xrightarrow{c_4} [\ \text{OHHL}\ ]_b \\
r_5 &: [\ \text{LuxBox}\ ]_b \xrightarrow{c_5} [\ \text{LuxBox} + \text{OHHL}\ ]_b \\
r_6 &: [\ \text{LuxBox}\ ]_b \xrightarrow{c_6} [\ \text{LuxBox} + \text{LuxR}\ ]_b \\
r_7 &: [\ \text{LuxR} + \text{OHHL}\ ]_b \xrightarrow{c_7} [\ \text{LuxR.OHHL}\ ]_b \\
r_8 &: [\ \text{LuxR.OHHL}\ ]_b \xrightarrow{c_8} [\ \text{LuxR} + \text{OHHL}\ ]_b \\
r_9 &: [\ \text{LuxR.OHHL} + \text{LuxBox}\ ]_b \xrightarrow{c_9} [\ \text{LuxR.OHHL.LuxBox}\ ]_b \\
r_{10} &: [\ \text{LuxR.OHHL.LuxBox}\ ]_b \xrightarrow{c_{10}} [\ \text{LuxR.OHHL} + \text{LuxBox}\ ]_b \\
r_{11} &: [\ \text{LuxR.OHHL.LuxBox}\ ]_b \xrightarrow{c_{11}} [\ \text{LuxR.OHHL.LuxBox} + \text{OHHL}\ ]_b \\
r_{12} &: [\ \text{LuxR.OHHL.LuxBox}\ ]_b \xrightarrow{c_{12}} [\ \text{LuxR.OHHL.LuxBox} + \text{LuxR}\ ]_b \\
r_{13} &: [\ \text{OHHL}\ ]_b \xrightarrow{c_{13}} \text{OHHL}\,[\ ]_b \\
r_{14} &: [\ \text{OHHL}\ ]_b \xrightarrow{c_{14}} [\ ]_b \\
r_{15} &: [\ \text{LuxR}\ ]_b \xrightarrow{c_{15}} [\ ]_b \\
r_{16} &: [\ \text{LuxR.OHHL}\ ]_b \xrightarrow{c_{16}} [\ ]_b
\end{aligned}
$$

Modelling Quorum Sensing in Vibrio fischeri (IV)

- $\Pi_k = (\Sigma, L, \mu, M_1, \mathcal{R})$, $1 \leq k \leq N$, where:
  - $\Sigma = \{\text{OHHL}\}$.
  - $L = \{b\}$.
  - $\mu = [\ ]$.
  - $M_1 = \{LuxBox\}$.

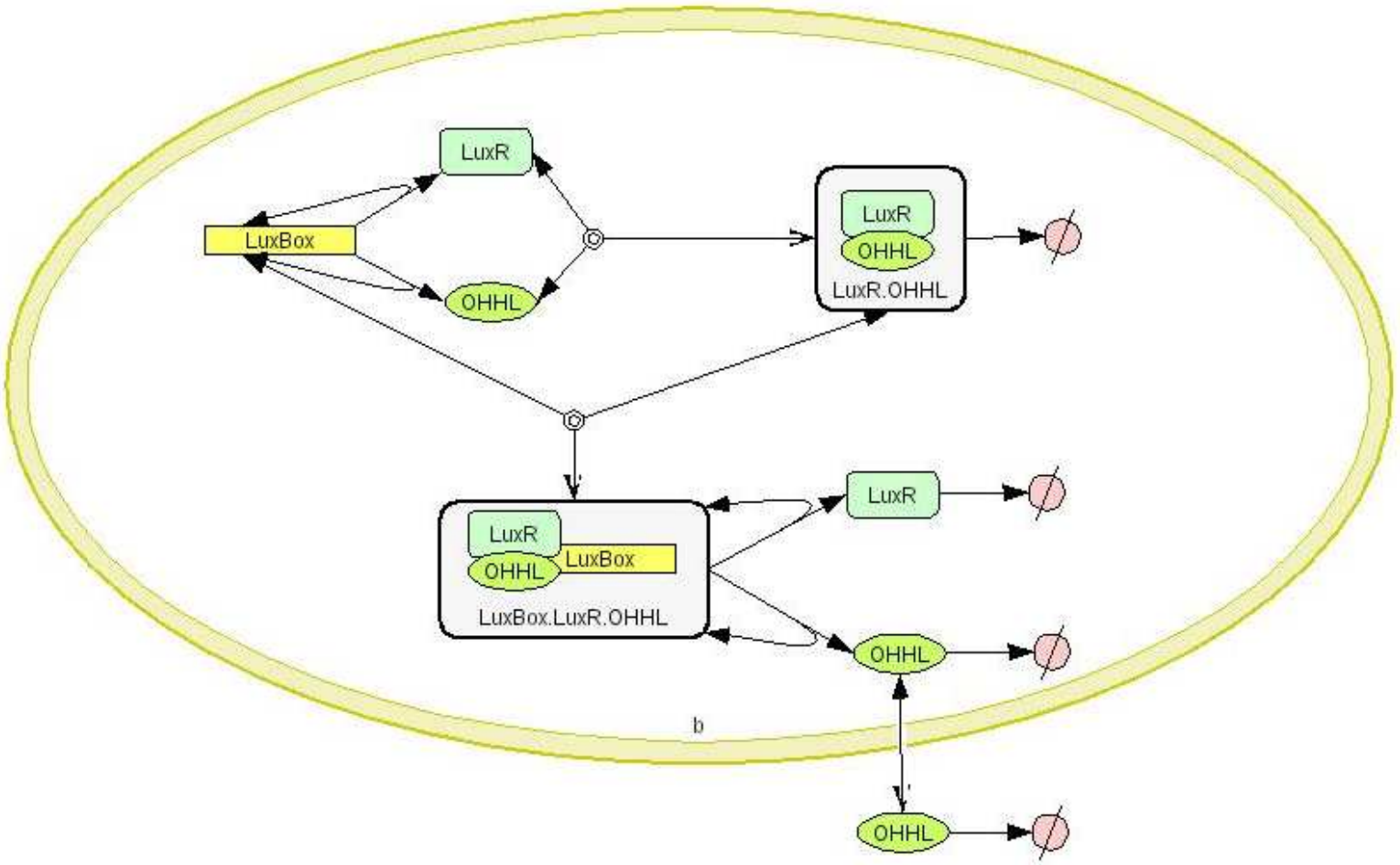
Modelling Quorum Sensing in Vibrio fischeri (IV)

- $\Pi_k = (\Sigma, L, \mu, M_1, \mathcal{R})$, $1 \leq k \leq N$, where:
  - $\Sigma = \{\text{OHHL}\}$.
  - $L = \{b\}$.
  - $\mu = [\ ]$.
  - $M_1 = \{LuxBox\}$.

Stochastic Constants associated with the rules:

$c_1 = 5, c_2 = 8, c_3 = 2, c_4 = 1, c_5 = 2, c_6 = 2, c_7 = 9, c_8 = 1$
$c_9 = 10, c_{10} = 2, c_{11} = 250, c_{12} = 200, c_{13} = 50, c_{14} = 30, c_{15} = 20, c_{16} = 20,.$

Modelling Quorum Sensing in Vibrio fischeri (V)

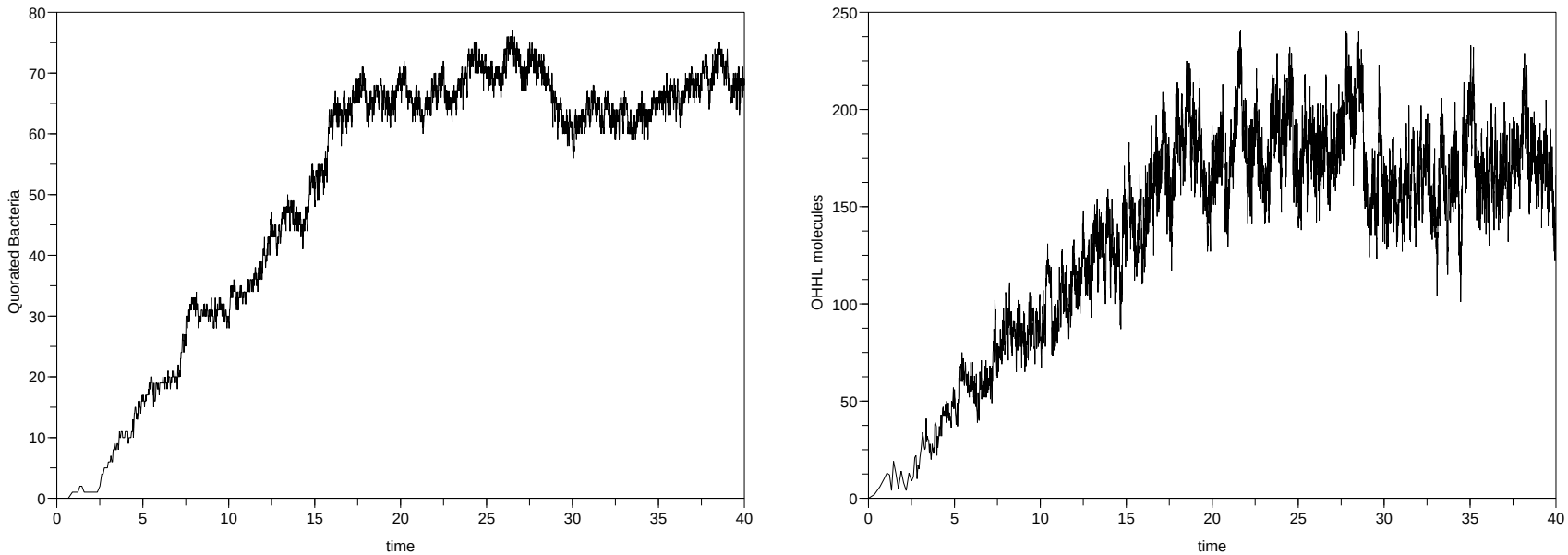


Results and Discussions

- The model has been represented in SBML (Systems Biology Markup Language).
- The SBML code was generated using CellDesigner.
- The semantics has been captured by the multicompartmental Gillespie algorithm.
- We have run our simulations using a program written in C with input file the SBML file specifying our model.
- The emergent behaviour of the system has been studied for three populations of different size.

A population of 100 bacteria (I)

Evolution over time of the number of quorated bacteria [1] and the number of signals (OHHL) in the environment.



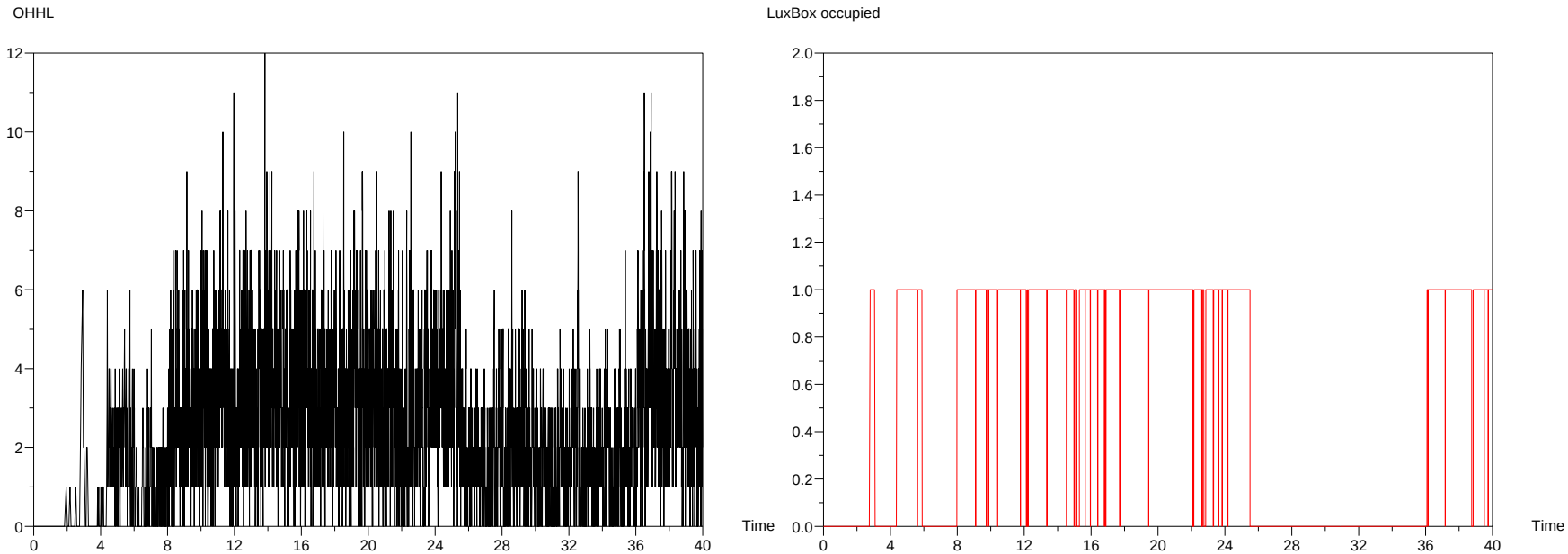
Number of quorated bacteria (left) and signals in the environment (right)

[1] A bacterium is quorated if the LuxBox in this bacterium is occupied by the complex LuxR-OHHL.

A population of 100 bacteria (II)

The behaviour of each individual in the population can be tracked.

- Correlation between the number of signals inside one bacterium (left) and the occupation of the LuxBox by the complex **LuxR**-**OHHL** (right).

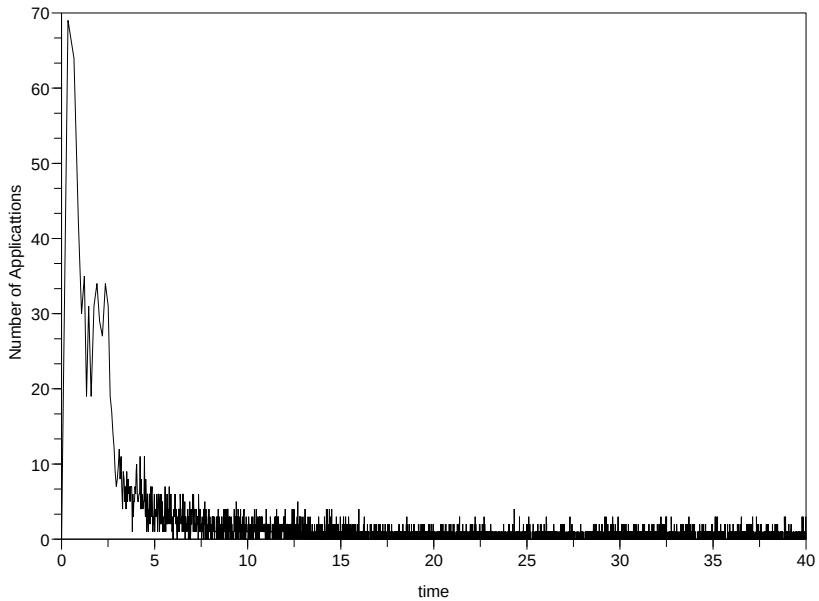


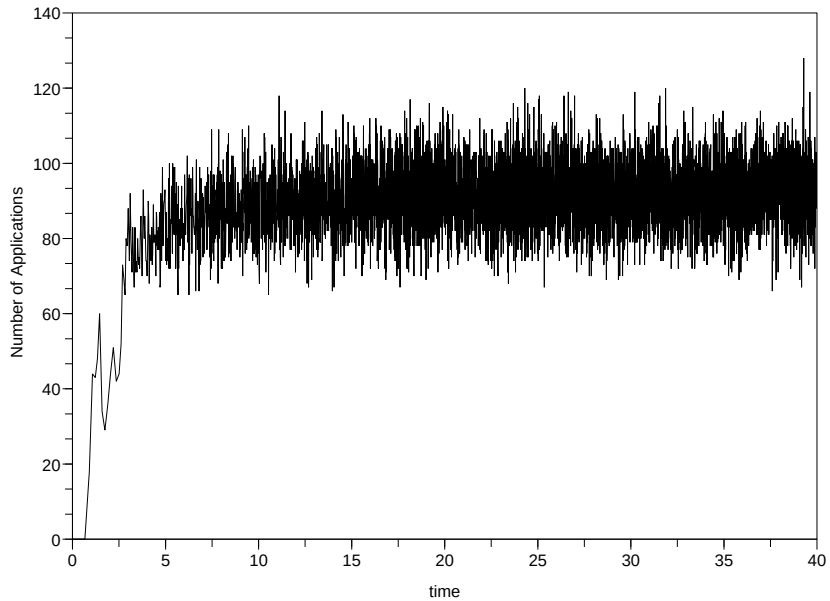
Number of signals and occupation of the LuxBox in a bacterium

A population of 300 bacteria

We can also study how rules are applied across the evolution of the system.

- Number of applications of the rule representing the basal production (left) and the rule representing the massive production of the signal.





Number of applications of rules r_5 and r_{11} in a population of 300 bacteria

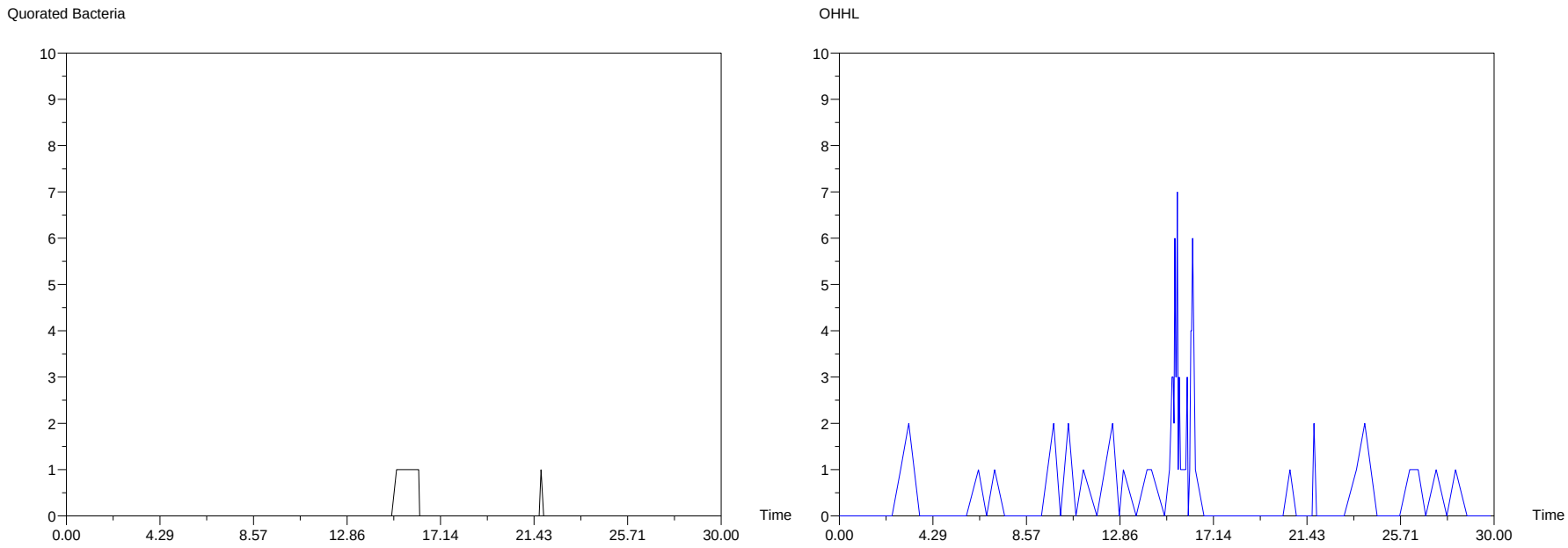
r_5 : $[\ \text{LuxBox}\ ]_b \xrightarrow{c_1} [\ \text{LuxBox} + \text{OHHL}\ ]_b$

r_{11} : $[\ \text{LuxR.OHHL.LuxBox}\ ]_b \xrightarrow{c_7} [\ \text{LuxR.OHHL.LuxBox} + \text{OHHL}\ ]_b$

A population of 10 bacteria (I)

Finally, we examine the behaviour of a population of only 10 bacteria.

In this case no recruitment process takes place and the signal does not accumulate in the environment.



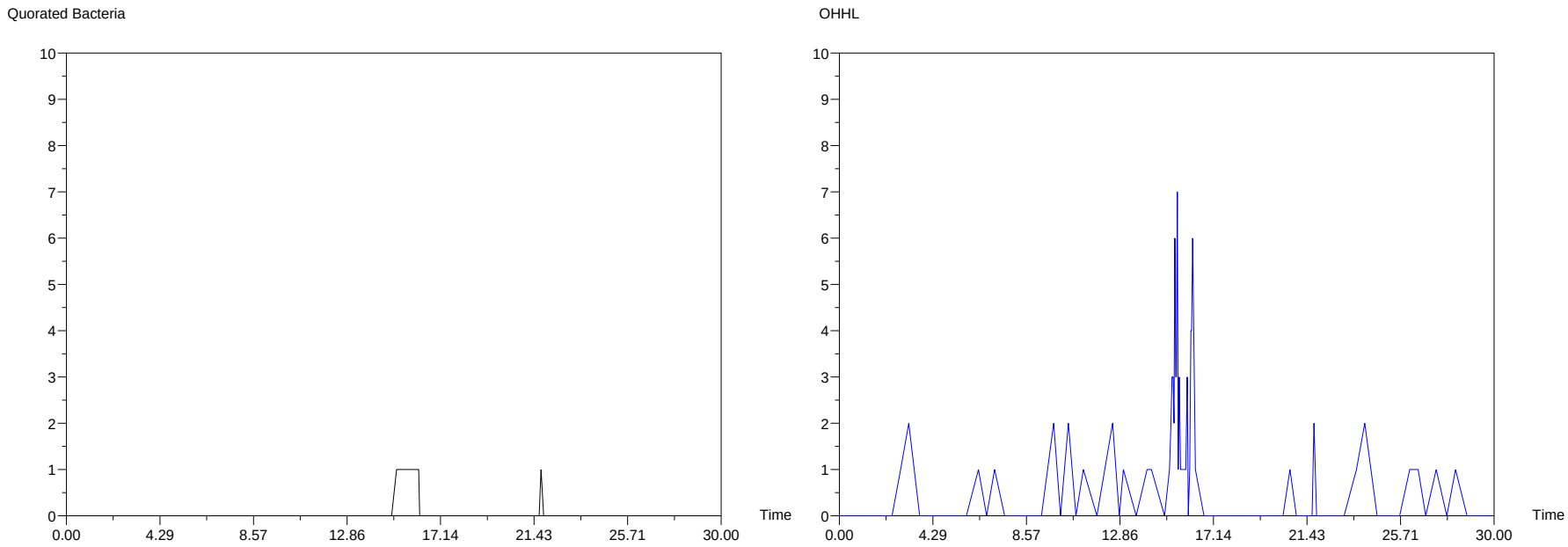
Quorated bacteria and signals in the environment in a population of 10 bacteria.

BUT ...

A population of 10 bacteria (I)

Finally, we examine the behaviour of a population of only 10 bacteria.

In this case no recruitment process takes place and the signal does not accumulate in the environment.



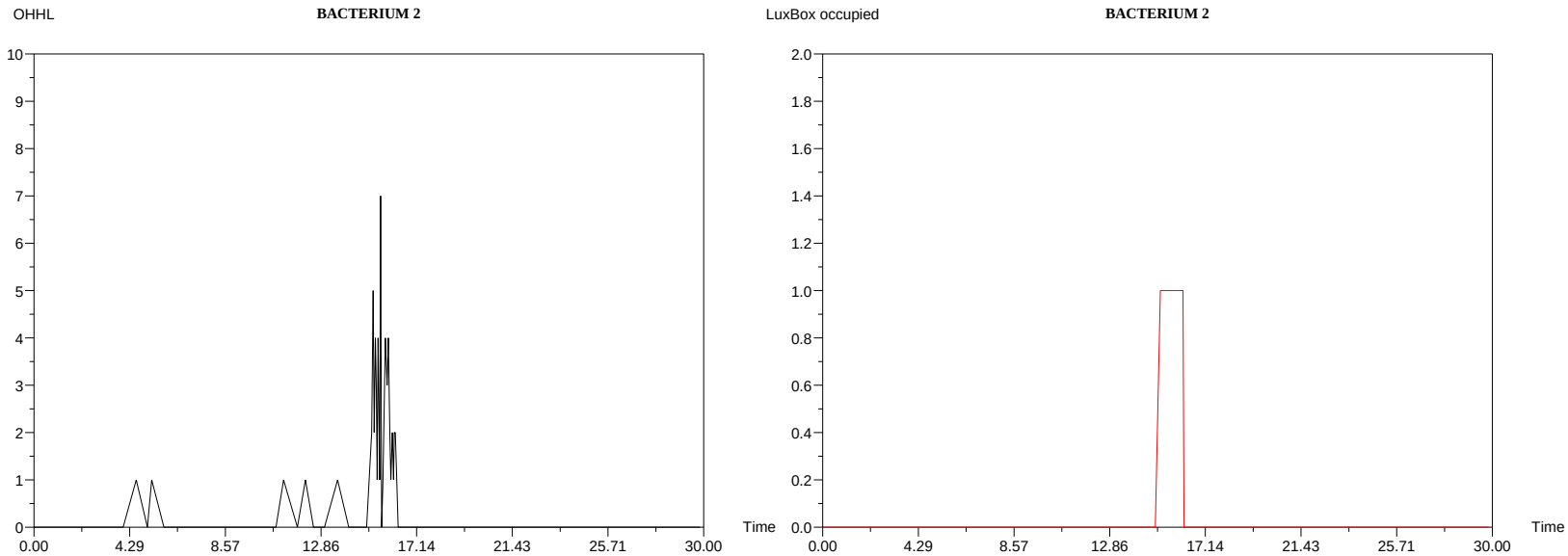
Quorated bacteria and signals in the environment in a population of 10 bacteria.

BUT ... one of the bacteria guessed wrong the size of the population and got upregulated.

A population of 10 bacteria (II)

But then, after sensing that the signal did not accumulate in the environment, it switched off its system.

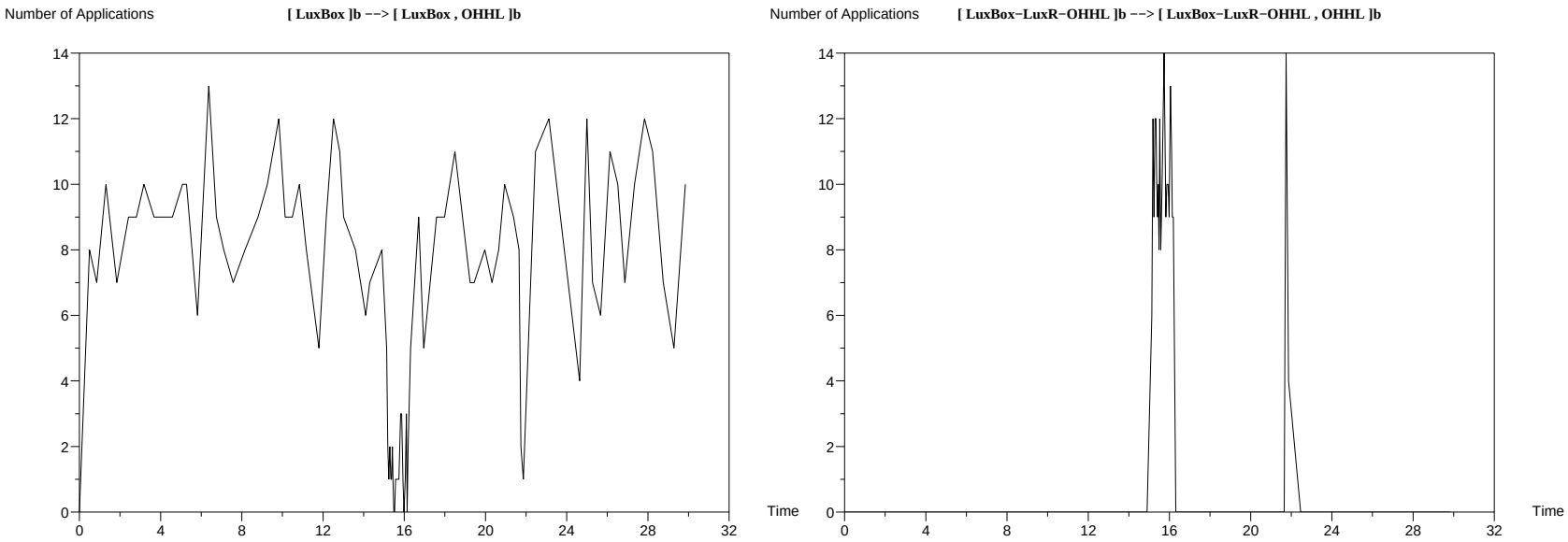
Next figure depicts the behaviour of the bacterium that got quorated.



Behaviour of a bacterium in a population of 10 bacteria.

A population of 10 bacteria (III)

Finally, we observe that for only 10 bacteria the system remains in an downregulated state.



Number of applications of rules r_5 and r_{11} in a populaton of 10 bacteria.