Noname manuscript No.
(will be inserted by the editor)

Deterministic and Stochastic P Systems for Modelling Cellular Processes

Marian Gheorghe · Vincenzo Manca · Francisco J. Romero-Campero*

Received: date / Accepted: date

Abstract This paper presents two approaches based on metabolic and stochastic P systems, together with their associated analysis methods, for modelling biological systems and illustrates their use through two case studies.

Keywords Membrane Computing · P systems · Modelling · Systems Biology · Synthetic Biology

1 Introduction

Membrane computing is a new and vigorous nature inspired computational paradigm which brings to computer science and associated areas a set of concepts, principles and information from cellular biology with the aim of producing a coherent, robust and efficient computational mechanism, called P system, that mimics the behaviour of some cellular processes [26].

This paradigm was initially introduced in theoretical computer science as an abstract computational mechanism [25] and it soon became apparent that this approach relying on hierachically organised compartments, complex multisets of objects and

M. Gheorghe
Department of Computer Science
The University of Sheffield
Regent Court, Portobello Street, Sheffield S1 4DP, UK
E-mail: M.Gheorghe@dcs.shef.ac.uk

V. Manca
Department of Computer Science
The University of Verona
Strada Le Grazie 15, 37 134, Verona, Italy
E-mail: Vincenzo.Manca@univr.it

F. J. Romero-Campero
School of Computer Science
The University of Nottingham
Jubilee Campus, Nottingham NG8 1BB, UK
E-mail: fxc@cs.nott.ac.uk
* The authors have equal contributions

diverse rewriting sets of rules, would impact on areas far beyond computer science. Applications in biology, economics, graphics, artificial life, self-assembly, are just few from a large list of existing and potential branches of research domains utilising P systems [26].

Of a particular interest are the applications in biology, as this line of research emerged in this context and utilises concepts and phenomena specific to it. Most of the theoretical and computational studies considered so far have imposed variants of P systems relying on an evolution strategy whereby in each step the rules are applied in a non-deterministic and maximally parallel manner. By contrast, most of the applications of P systems in biology have used different execution strategies based on deterministic or probabilistic/stochastic approaches. These variants of P systems, by introducing a certain control over the strategy of applying the rules, allow to perform qualitative and approximate or exact quantitative studies.

In this paper we present two of the most successful modelling approaches based on P systems and largely utilised to describe biological systems. One of these approaches, based on metabolic P systems, shortly MP systems, uses a deterministic evolution ruled by a set of functions, called flux maps, that govern the execution of the system. The other approach uses stochastic P systems, a method relying on a particular scheduling mechanism based on Gillespie algorithm [13]. Two case studies will illustrate the use of these approaches in modelling the behaviour of biological systems and the interplay between these models and other tools utilised in the analysis of biological systems.

2 Preliminaries

2.1 Metabolic P Systems: Basic Concepts and Definitions

In metabolic systems, matter inside a reactor is partitioned in a certain number of substances transformed in time by some reactions. If we consider the system, along a discrete number of steps, at some specific time interval, we observe that the reactions transform the substance matter, but also that matter (of given types) is introduced from the external environment, or expelled outside. Therefore, abstractly, reactions are agents performing matter transformations. Avogadro's principle, which is fundamental in chemistry, rules the behaviour of any reaction r, at any step. In fact r "moves" (consumes/produces) multiples of the same number of objects (molecules), which we call *reaction unit* or flux of r. The "stoichiometry" of r establishes its "reactants" (consumed substances) and its "products" (produced substances). We call this kind of transformation mechanism a *molar multiset rewriting*. It differs from the usual multiset rewriting of P systems. In fact, in the classical case, a rule $ddc \to b$ replaces two d and one c by only one b (the order does not matter), and the application of such a rule is *individual*. On the contrary, in a molar multiset rewriting perspective, an occurrence of d, c, and b in a rule means a population of d, c, and b respectively, and the size of this population is just the reaction unit of the reaction (a value depending on the state of the system). The place of substances, with respect to the arrow, $\to$, of the rule, means increment or decrement (production or consumption). This perspective implies that the time of the system is not the microscopic time of reaction kinetics, but the macroscopic time of the observer. Therefore, it is enough to know the substance variations between two consecutive observation instants. The additivity of the effects of all reactions correspond to another chemical principle, referred to as Dalton's principle.

According to it, if we compute the reaction unit of each reaction, then by adding the effects of all reactions we can compute the next state of the system. This strategy, we call it the *matter partition*, is the essence of the MP dynamics [17–19, 21–24].

In simple words, a computation step in the dynamics of an MP is obtainable in the following way: *i) compute the reaction units; ii) apply the reactions, according to the reaction units they transform; iii) replace the matter they consume with the matter they produce.*

In the following, Greek letters $\alpha, \beta, \ldots$ (with possible subscripts) denote finite multisets (strings where symbol order is irrelevant) over an alphabet X of substances. A reaction r is also indicated by $\alpha_r \to \beta_r$, where α_r, β_r are the reactants and products of r. For a multiset α, we denote by $|\alpha|_x$ the multiplicity of x in α and by $|\alpha|$ the sum $\sum_{x\in\alpha} |\alpha|_x$, where $x \in \alpha$ means that the multiplicity of x in α is different from zero.

Definition 1 The **stoichiometric matrix** $\mathbb{A}$ of a set R of reactions over a set X of substances is $\mathbb{A} = (\mathbb{A}_{x,r} \mid x \in X, r \in R)$ where $\mathbb{A}_{x,r} = |\beta_r|_x - |\alpha_r|_x$. The set of reactions having the substance x as a reactant is $R_\alpha(x) = \{r \in R \mid |\alpha_r|_x > 0\}$ and the set of rules consuming or producing x is $R(x) = \{r \in R \mid \mathbb{A}_{x,r} \neq 0\}$. Two reactions r_1, r_2 **compete** for some substance $x \in X$ if $r_1, r_2 \in R_\alpha(x)$.

A discrete dynamical system is given by a set of states and by a discrete dynamics on them, that is, by a function from the set $\mathbb{N}$ of natural numbers to the states of the system. In this context, the natural numbers which are arguments of the dynamics are called *instants* or *steps*. This general notion of dynamical system is the common basis of the types of MP systems we will subsequently define. The following definition introduces a class of MP systems, which are called MP systems with flux maps, or shortly MPF systems (F may be omitted).

Definition 2 (MPF System) An MP system with flux regulation maps, shortly an MPF system, is a discrete dynamical system given by a construct

$$M = (X, R, V, Q, \Phi, \nu, \mu, \tau, q_0, \delta)$$

where X, R, V are finite disjoint sets, and the following conditions hold, with $n, m, k \in \mathbb{N}$:

- $X = \{x_1, x_2, \ldots, x_n\}$ is the set of **substances** (the types of molecules);
- $R = \{r_1, r_2, \ldots, r_m\}$ is the set of **reactions** over X, that is, pairs (in arrow notation) of type $\alpha \to \beta$ with α, β strings over the alphabet X (sometimes concatenation is denoted by $+$ to stress the commutativity implicit in the string notation of multisets);
- $V = \{v_1, v_2, \ldots, v_k\}$ is the set of **parameters** (such as pressure, temperature, volume, pH, ...) equipped with a set $\{h_v : \mathbb{N} \to \mathbb{R} \mid v \in V\}$ of **parameter evolution functions**, the elements of $X \cup V$ are called **magnitudes**;
- Q is the set of **states**, that is, of the functions $q : X \cup V \to \mathbb{R}$, from magnitudes to real numbers.
- $\Phi = \{\varphi_r \mid r \in R\}$ is a set of **flux (regulation) maps**, where the function $\varphi_r : Q \to \mathbb{R}$ states the amount (moles) which is consumed/produced, in the state q, for every occurrence of a reactant/product of r. We set by $U(q) = (\varphi_r(q) | r \in R)$ the **flux vector** at state q;
- ν is a natural number which refers to the number of molecules of a (conventional) mole of M, as its **population unit**;

- μ is a function which assigns, to each $x \in X$, the **mass** $\mu(x)$ of a mole of x (with respect to some measure unit);
- τ is the **temporal interval** between two consecutive observation steps;
- $q_0 \in Q$ is the **initial state**;
- $\delta : \mathbb{N} \to Q$ is the dynamics of the system. At the initial instant it provides the initial state, that is, $\delta(0) = q_0$. Moreover, for any parameter $v \in V$ and for any $i \geq 0$, $(\delta(i))(v) = h_v(i)$, while the function δ provides the evolution of substance quantities by means of stoichiometric matrix $\mathbb{A}$ and vectors U. In fact, let us set, for any magnitude $w \in X \cup V$ and $i \geq 0$

$$(\delta(i))(w) = w[i]$$

and

$$X[i] = (x[i] \mid x \in X)$$

then the dynamics of substances is given by the following recurrent vector equation also called **Equational Metabolic Algorithm** (**EMA**), where $\times$ is the usual matrix multiplication, and the sum between vectors is the usual component-wise sum (analogously, vector difference and division are component-wise difference and division):

$$X[i+1] = \mathbb{A} \times U(\delta(i)) + X[i] \tag{1}$$

where $\mathbb{A}$ is the stoichiometric matrix of R over X (according to dynamics δ, the value $\delta(i)$ identifies some state $q \in Q$).

If $EMA[i]$ is the system at step i, given the vectors of $U(\delta(i))$ and $X[i]$, we can obtain the vector $X[i+1]$ by evaluating the right member of the equation above.

All the components of an MP system, apart from the set Q of states (deducible from the other components), and the dynamics, constitute an **MP graph**, easily representable in graphical form (see Fig. 1). When in a MP graph the elements τ, ν, μ are omitted, then we call it an **MP grammar**, that is, a multiset rewriting grammar where rules are regulated by functions (usually rational algebraic expressions). Such a grammar is completely defined by: i) reactions, ii) flux maps (substances are the elements occurring in the reactions, and parameters are the arguments of flux maps different from substances), iii) parameter evolution maps, and iv) initial values of magnitudes. Parameter evolution maps or initial values may be omitted when only the MP grammar structure is specified. A kind of equivalence between MPF systems and hybrid Petri nets was proved in [8], where it is shown that MP formulation provides logical and computational advantages.

MP systems proved to be relevant in the analysis of dynamics of metabolic processes. Their structure clearly distinguishes a reaction level and a regulation level. We showed that an essential component of the regulation level can be deduced by applying the log-gain theory to data that can be collected from observations of the system. The log-gain method can deduce, in a given metabolic system, a time series of (approximate) flux unit vectors $U[i]$ (i ranging in time instants), from a time series of observed states (substance and parameter values). This method [19,21], uses a suitable formulation of the *allometric principle* [6], which along a process of adaptation, by means of the specific biological information about the model under investigation, provides a square, univocally solvable linear system, called $OLGA[i]$ involving substances, parameters and flux units for each step i of an observation time series. The solution of this

system provides the flux units in that state. In this manner a time series of flux units is recovered, and from it, by using regression and optimization techniques, the MP flux regulation maps can be deduced. This is a crucial aspect for identifying models when no other model is available and microscopic determination of kinetic rates are difficult, or even impossible, to perform. This method was successfully applied in several cases. In particular a complex model of a photosynthetic process was recently obtained [24] entirely by this procedure.

2.2 Stochastic P Systems: Basic Concepts and Definitions

The probabilistic and stochastic approaches discussed in this paper rely on a basic cell-like P system structure [26]. This system, although it relies on the same basic concept of P system, as metabolic P system, has many different characteristics. It is a stochastic mechanism, it has, in general, more than one compartment, exhibits a different behaviour due to a special scheduling procedure to execute the rules. Consequently, some notations below, even those that refer to the same basic components might be different from those used by MP systems.

Definition 3 A *stochastic P system* is a construct:

$$\Pi_S = (V, \mu, ms^1, \ldots, ms^n, R_1, \ldots, R_n)$$

where:

- V is the finite alphabet of *simple objects*;
- μ is the *membrane structure*, formally represented, as a rooted tree, where the nodes are called membranes, and the relationship of a membrane being inside another one is specified as the corresponding node being the descendant of the other one;
- ms^i, $1 \leq i \leq n$, is the *initial state* of the compartment i, consisting of a multiset of objects from V;
- R_i, $1 \leq i \leq n$, is the finite set of *evolution and communication rules* on multisets of objects. Each evolution-communication rule from R_i is represented by a generic rewriting rule, called boundary rule [5], and has the following form:

$$r: \; ms_1 \, [\, ms_2 \,]_i \xrightarrow{c} ms_1' \, [\, ms_2' \,]_i$$

These multiset rewriting rules operate on both sides of the membrane defining the compartment i, i.e., a multiset ms_1 outside the membrane i and a multiset ms_2 from the compartment i are simultaneously replaced by the multisets ms_1' and ms_2', respectively. When ms_1, ms_1' are not present then one has an evolution rule, whereas when either $ms_1 = ms_2'$ and ms_2, ms_1' are absent or $ms_2 = ms_1'$ and ms_1, ms_2' are not present, then r denotes a communication rule. The constant c associated with this rule is called *kinetic stochastic constant* and is used together with the number of objects occurring on their left-hand side to compute the probabilities associated to them.

The simple objects, elements of V, denote simple molecular species; one can also describe by strings over V more complex molecules, like DNA's, RNA's, proteins, but these are not discussed here. A multiset, ms, is represented as $ms = e_1 + \ldots + e_p$, with e_i denoting simple objects. The symbol $+$ might be omitted when writting a multiset.

A computation in a stochastic P system, Π_S, starts from the initial multisets by setting the simulation time to 0. In each iteration and in every compartment, Gillespie's algorithm [13] is used to compute a rule that will be next executed and the time needed to apply it. The rule with the smallest time value is selected to be executed, the simulation time and the waiting time of the rules selected from the other compartments are updated. In the next step, only the compartment(s) affected by the application of the last rule is (are) involved in identifying the rule(s) and its (their) associated execution time(s) according to Gillespie method. This algorithm is very precise and widely used in stochastic simulations. However, it is slow due to a sequential execution of the rules across the entire system.

Cellular systems are highly complex structures with many compartments and numerous molecules interacting in various ways. The compartments are, in general, tightly inter-related and their contents change over time. In order to faithfully reflect the complex processes that take place, it is necessary to establish a set of modelling principles. In any formal framework a set of implicit or explicit robust methods and techniques are employed in order to automate the process of generating the model, especially for highly complex systems with many components and interactions between them. This approach is also used to devise sound principles of mapping between different formalisms and to formally verify and test certain properties of the dynamics of the system. A mapping between cellular regions and compartments, molecules, molecular and inter-cellular interactions and various parts of a stochastic P system has been identified according to the basic principles of P systems [26] and some modelling requirements [30]. According to this mapping, biological compartments are described as P systems regions, chemical molecules are defined by objects and interactions by evolution-communication rules. In this respect a broad range of chemical interactions - complex formation and dissociation, binding and debinding, diffusion, degradation etc., can be represented in a uniform and rigorous way.

A first attempt to model molecular interactions based on P systems is given in [15] and a stochastic approach has been provided in [30]. Multicellular systems can also be represented in this context by using a tissue-like paradigm, whereby the hierachical structure of compartments is replaced by a network of components and hierarchical interactions are replaced by interactions between neighbours. The approach has been used to abstractly represent and model colonies of bacteria [3] and quorum sensing systems in the marine bacterium *Vibrio fischeri* from an artificial life perspective [32]. A similar approach to stochastic P systems uses the concept of dynamical probabilistic P systems [28]; in this case the maximal parallelism feature combined with a probabilistic way of selecting the rules is applied to model biological systems.

Any modelling approach, either continuous or discrete, deterministic or stochastic, provides through its semantics a way to execute and consequently to obtain simulations for the systems analysed. On the other hand it brings a rich palette of methods and tools to analyse a system and to verify certain properties of the dynamics of the system. One of the most popular methods to verify such properties relies on the use of adequate model checkers. For systems with a stochastic behaviour such a model checker is PRISM (abbreviation for Probabilistic and Symbolic Model Checker), which offers both a specification language, allowing the representation of the data and processes involved and a query language based on temporal logic, which is used to verify various properties of the model [16]. The main parts of a P system definition, components and objects, are mapped into modules and variables, respectively, and the evolution and communication rules are transformed into commands simulating them [4,31].

A PRISM specification of the model allows to simulate it and generate information regarding the bahaviour of the system and collect data related to the concentration level of various molecular species for given initial values. Using the temporal logic query language certain properties of the model can be also verified; for instance, whether certain molecules reached given concentration levels, equilibrium and/or steady states etc. [4,31]). To help identifying the right properties that are verified, a methodology based on reverse-engineering specifications from software systems [11], in terms of invariants or rules that hold true at particular points of a program, has been established and successfully applied for some systems [4].

Very often a model is quite complex and might contain unknown elements or noisy data. We face these problems for P system models as well. Although the structure of a P system is inherently divided into hierarchically organised compartments, in certain cases this might not be enough to manage the high complexity of the system. Through modularity design [33] a certain level of finer granularity is obtained, but also one may get a greater generality by reusing some of the designed components. In order to tackle the unknown structural information or to estimate various parameters that occur in these systems, a certain methodology has been successfully implemented [29].

3 Modelling a *Mitotic Oscilator* with Metabolic P Systems

Now we consider a classical example of biological modelling, which concerns an important case study of the mitotic oscillator, reported in [14]. Mitotic oscillations are mechanisms exploited by nature to regulate the onset of mitosis, that is, the process of cell division aimed at producing two identical daughter cells from a single parent cell. More precisely, mitotic oscillations concern the fluctuation in the activation state of a protein produced by *cdc2* gene in fission yeasts or by homologous genes in other eukaryotes. The model here considered focuses on the simplest form of this mechanism, as it is found in early amphibian embryos. Here cyclin is synthesized at a constant rate and triggers the transformation of inactive (M^+) into active (M) cdc2 kinase, by enhancing the rate of a phoshatase. A second kinase reverts this modification. On the other hand a third kinase elicits the transformation from the inactive (X^+) to the active (X) form of a protease that degrades cyclin, and this activation is reverted by a phoshatase. Magnitudes V_i, V_d denote the speed of cyclin production and degradation respectively, while V_2, V_4 are prameters defined in Table 1, and V_1, V_3 are parameters defined in Table 4.

The activation of *cdc2 kinase* provides the formation of a complex known as M-phase promoting factor (or *MPF*). The complex triggers mitosis and the degradation of cyclin leads to the inactivation of the cdc2 kinase that brings the cell back to the initial conditions in which a new division cycle can take place. In yeasts and in somatic cells the cell cycle is subject to the control of many checkpoints, but the mechanism based on the activation-inactivation of cdc2 kinase remains the same [1].

Table 1 Golbeter's values of constants.

$K1 = 0.005$	$K2 = 0.005$	$K3 = 0.005$	$K4 = 0.005$
$VM1 = 3\ min^{-1}$	$VM3 = 1\ min^{-1}$	$V2 = 1.5\ min^{-1}$	$V4 = 0.5\ min^{-1}$
$V_i = 0.025\ \mu M \cdot min^{-1}$	$V_d = 0.25\ \mu M \cdot min^{-1}$	$Kc = 0.5\ \mu M$	$KKd = 0.02\ \mu M$
$Kd = 0.01\ min^{-1}$	$S = 0.001$		

Table 2 Initial values of substances.

$C = 0.01$	$M = 0.01$	$Mp = 0.99$	$X = 0.01$	$Xp = 0.99$

Table 3 MP reactions and flux maps of Golbeter's mitotic oscillator (λ is the empty multiset).

$R1 : \lambda \to C$	$F1 = S \cdot Vi$
$R2 : C \to \lambda$	$F2 = S \cdot K_d \cdot C$
$R3 : C\ Mp \to C\ M$	$F3 = (S \cdot V1 \cdot Mp)/(K1 + Mp)$
$R4 : C\ X \to X$	$F4 = (S \cdot Vd \cdot X \cdot C)/(KKd + C)$
$R5 : M \to Mp$	$F5 = (S \cdot V2 \cdot M)/(K2 + M)$
$R6 : Xp\ M \to X\ M$	$F6 = (S \cdot V3 \cdot Xp)/(K3 + Xp)$
$R7 : X \to Xp$	$F7 = (S \cdot V4 \cdot X)/(K4 + X)$

The following equations provide Golbeter's model:

$$\begin{aligned} \frac{dC}{dt} &= V_i - V_d X \frac{C}{KK_d + C} - K_d C \\ \frac{dM}{dt} &= V_1 \frac{(1-M)}{K_1 + (1-M)} - V_2 \frac{M}{K_2 + M} \\ \frac{dX}{dt} &= V_3 \frac{(1-X)}{K_3 + (1-X)} - V_4 \frac{X}{K_4 + X} \end{aligned} \tag{2}$$

The corresponding MP model is identified, according to a procedure similar to one introduced in [12]. In this case, differently than [19], dynamics was given by EMA (equation (1) of Definition 2), with a more natural procedure for inferring flux regulation maps from the Golbeter's differential model. Namely, from the first equation of (2) we deduce the fluxes of reactions consuming and producing C ($R1$ producing, and $R2.R4$ consuming). They are multiplied by a variable S representing the speed of reactions at each step (in our simulation $S = 0.001$). Analogously, from the second equation fluxes for reactions $R3$ and $R5$ were deduced (producing and consuming M), and finally, from the third equation fluxes of reactions $R6$ and $R7$ were deduced (producing and consuming X).

Tables 1, 2, 3, 4, and Figures 1, 2 refer to this MP model and its dynamics, which was generated by the software $MetaPlab$ (available at http://mplab.sci.univr.it which implements many tools of design and analysis of MP systems). The curves of our MP dynamics coincide perfectly with the classical curves of Golbeter's model.

Table 4 Parameters evolutions and initial values.

$V1:$	Initial values: 0.0588	Evolution: $(C \cdot VM1)/(Kc + C)$
$V3:$	Initial values: 0.01	Evolution: $M \cdot VM3$

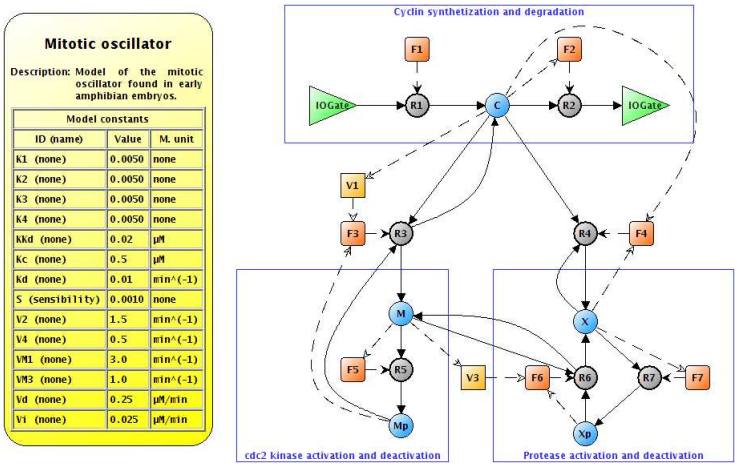

Fig. 1 The MP graph of the mitotic oscillator according to the graphical notation introduced in [22].

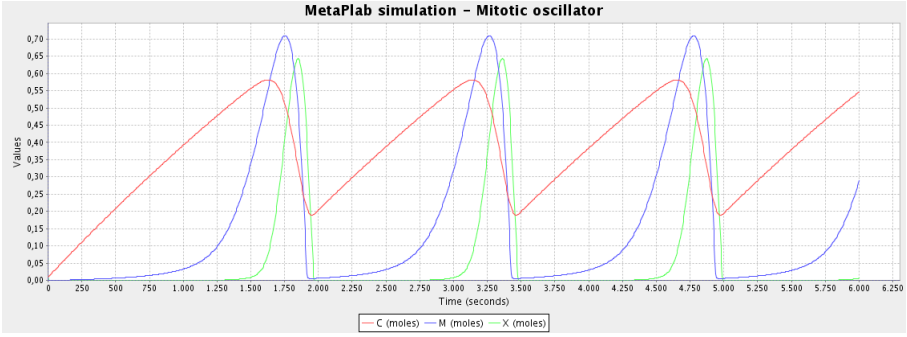

Fig. 2 The dynamics of the mitotic oscillator generated by the MetaPlab software [7].

4 Modelling the *Repressilator* with Stochastic P Systems

Transcriptional regulatory networks are responsible for the functioning of essential cellular systems. Nevertheless, the *design principles* of the mechanisms involved in these processes are not fully understood. A recent approach to advance the understanding of their functioning consists in the design and construction of synthetic transcriptional networks exhibiting desirable behaviour, *synthetic biology* [2].

One of the main goals in synthetic biology is the characterisation of gene regulation modular patterns exhibiting certain functionalities. These molecular modules should be, to some extent, separable or orthogonal to the rest of the network where they are embedded, in such a way that their functioning is preserved when they are removed from their original network and synthetically wired to a new one.

Stochastic P systems aim at providing an efficient modelling methodology for synthetic biology by introducing the concept of *P system module* [33]. A basic library of P system modules describing the best known and characterised gene regulation modular patterns was introduced in [33]. An evolutionary algorithm able to automatically generate gene regulatory networks exhibiting a prefixed behaviour by combining P systems modules was developed in [29].

The *Repressilator* is one of the first synthetic genetic circuits [10]. It consists of three genes, *lacI*, the operon lactose repressor, *tetR*, the repressor from the tetracycline transposon, and *cI*, a repressor from the λ phage virus. The network is built in such a way that lacI gene represses cooperatively the tetR gene, which in turn represses coopertively cI. Finally in order to close the cycle the gene cI represses lacI.

In what follows we introduce a P system modular specification of the Repressilator which could help understanding its rational design. Two different P system modules are used.

The first one, $UnReg(\{G, R, P\}, \{c_1, c_2, c_3, c_4\}, \{l\})$, describes the unregulated expression of a gene G into its mRNA R, rule r_1 in (3); which is in turn translated into its corresponding protein P, rule r_2 in (3). Degradation of the mRNA and protein are also modelled by rules r_3 and r_4 in (3).

$$UnReg(\{G, R, P\}, \{c_1, c_2, c_3, c_4\}, \{l\}) = \left\{ \begin{array}{ll} r_1: & [\ G\ ]_l \xrightarrow{c_1} [\ G + R\ ]_l, \\ r_2: & [\ R\ ]_l \xrightarrow{c_2} [\ R + P\ ]_l, \\ r_3: & [\ R\ ]_l \xrightarrow{c_3} [\ \ ]_l, \\ r_4: & [\ P\ ]_l \xrightarrow{c_4} [\ \ ]_l \end{array} \right\} \tag{3}$$

The second P system module, $CoopRepr(\{Rep, G, R\}, \{c_1, \ldots, c_6\}, \{l\})$, describes the repression mechanism of a repressor *Rep* over a gene G whereby a first repressor molecule binds to the promoter of the gene with a low affinity, rules r_1 and r_2 in (4). In contrast this first binding helps a second repressor to bind to the promoter of the gene with a high affinity, rules r_3 and r_4 in (4). A leakiness in transcription is taken into account even when the repressors are bound to the promoter of the gene, rules r_5 and r_6 in (4).

$$CoopRepr(\{Rep, G, R\}, \{c_1, \ldots, c_6\}, \{l\}) = \left\{ \begin{array}{ll} r_1: & [\ Rep + G\ ]_l \xrightarrow{c_1} [\ Rep.G\ ]_l, \\ r_2: & [\ Rep.G\ ]_l \xrightarrow{c_2} [\ Rep + G\ ]_l, \\ r_3: & [\ Rep + Rep.G\ ]_l \xrightarrow{c_3} [\ Rep_2.G\ ]_l, \\ r_4: & [\ Rep_2.G\ ]_l \xrightarrow{c_4} [\ Rep + Rep.G\ ]_l, \\ r_5: & [\ Rep.G\ ]_l \xrightarrow{c_5} [\ Rep.G + R\ ]_l, \\ r_6: & [\ Rep_2.G\ ]_l \xrightarrow{c_6} [\ Rep_2.G + R\ ]_l, \end{array} \right\} \tag{4}$$

The design of the Repressilator is obtained by instantiating these modules with the objects describing the corresponding genes, mRNAs and protein products in Table 5 and with the stochastic constants in Table 6 according to the reported estimates in [10]. This P system modular specification of the Repressilator is presented in Table 7, its corresponding expansion using only elementary P system rules is shown in Table 8 and a graphical representation of the circuit is presented in Figure 3.

Our modular P system specification of the Repressilator was executed according to the stochastic semantics based on *Gillespie algorithm* associated with stochastic P systems [27] in order to obtain different simulations of the behaviour of the Repressilator, see Figure 4. Our results show that the circuit produces oscillations of the three

Table 5 Specification of the molecular species

Objects	Molecular Species
$lacI, mlacI, LacI$	lacI gene, mRNA and protein
$CI.lacI, CI_2.lacI$	CI repressors bound to the lacI promoters
$tetR, mtetr, TetR$	tetR gene, mRNA and protein
$LacI.tetR, LacI_2.tetR$	LacI repressors bound to the tetR promoters
cI, mcI, CI	cI gene, mRNA and protein
$TetR.cI, TetR_2.cI$	TetR repressors bound to the cI promoters

Table 6 Stochastic Parameters

Stochastic Constant	Description
$c_1 = 1\ s^{-1}$	Repressor binding to the promoters
$c_2 = 224\ s^{-1}$	Repressor debinding from first occupied promoter
$c_3 = 9\ s^{-1}$	Repressor debinding from second occupied promoter
$c_4 = 0.5\ s^{-1}$	Transcription from unoccupied promoters
$c_5 = 5 \times 10^{-4}\ s^{-1}$	Transcription from occupied promoters
$c_6 = 0.167\ s^{-1}$	Translation rate
$c_7 = 5.78 \times 10^{-3}\ s^{-1}$	Messenger RNA degradation
$c_8 = 1.16 \times 10^{-3}\ s^{-1}$	Protein degradation

Table 7 Modular Specification of the Repressilator

Module	Description
$UnReg(\{lacI, mlacI, LacI\}, \{c_4, c_6, c_7, c_8\}, \{b\})$	Unregulated expression of $lacI$
$CoopRepr(\{LacI, tetR, mtetR\}, \{c_1, c_2, c_1, c_3, c_5, c_5\}, \{b\})$	LacI cooperative repression over $tetR$
$UnReg(\{tetR, mtetR, TetR\}, \{c_4, c_6, c_7, c_8\}, \{b\})$	Unregulated expression of $tetR$
$CoopRepr(\{TetR, cI, mcI\}, \{c_1, c_2, c_1, c_3, c_5, c_5\}, \{b\})$	TetR cooperative repression over cI
$UnReg(\{cI, mcI, CI\}, \{c_4, c_6, c_7, c_8\}, \{b\})$	Unregulated expression of cI
$CoopRepr(\{CI, lacI, mlacI\}, \{c_1, c_2, c_1, c_3, c_5, c_5\}, \{b\})$	CI cooperative repression over $lacI$

repressor molecules according to the order they are connected in the network; that is TetR, CI and LacI. The oscillations are very noisy during the first 100 minutes to subsequently become more even. Nevertheless, the amplitude and frequency of the oscillations vary considerably during the simulations due to stochastic effects, see Figure 4.

Our methodology based on stochastic P systems is not only limited to the generation of simulations. We also support exhaustive and formal analysis of P system models using the probabilistic and symbolic model checker PRISM [4,31]. In our study of the Repressilator we analysed the advantage of using cooperative versus non-cooperative repression. In cooperative repression, the mechanism used in the Repressilator, two repressors molecules are needed to fully repress transcription. First a repressor molecule recognises the promoter of a gene with a low affinity and subsequently it helps another repressor to bind the promoter with a higher affinity. In contrast, in non-cooperative repression the binding of a single repressor molecule to the promoter of a gene is enough to stop transcription.

We have compared both mechanisms according to their *repressor efficiency*, the expected occupancy time of the gene promoter in full repression for different number of repressors. This was achieved by associating a reward to each state of our model corresponding to the presence of the object which represents the fully repressed gene. A temporal logic query was formulated to compute the expected cumulative reward

Table 8 Repressilator rules

Number	Rule	Stochastic Constant
r_1 :	$[\ lacI\ ]_b \xrightarrow{c_1} [\ lacI + mlacI\ ]_b$	$c_1 = 0.5\ s^{-1}$
r_2 :	$[\ mlacI\ ]_b \xrightarrow{c_2} [\ ]_b$	$c_2 = 5.78 \times 10^{-3}\ s^{-1}$
r_3 :	$[\ mlacI\ ]_b \xrightarrow{c_3} [\ mlacI + LacI\ ]_b$	$c_3 = 0.167 \times 10^{-4}\ s^{-1}$
r_4 :	$[\ LacI\ ]_b \xrightarrow{c_4} [\ ]_b$	$c_4 = 1.16 \times 10^{-3}\ s^{-1}$
r_5 :	$[\ LacI + tetR\ ]_b \xrightarrow{c_5} [\ LacI.tetR\ ]_b$	$c_5 = 1\ s^{-1}$
r_6 :	$[\ LacI.tetR\ ]_b \xrightarrow{c_6} [\ LacI + tetR\ ]_b$	$c_6 = 224\ s^{-1}$
r_7 :	$[\ LacI + LacI.tetR\ ]_b \xrightarrow{c_7} [\ LacI_2.tetR\ ]_b$	$c_7 = 1\ s^{-1}$
r_8 :	$[\ LacI_2.tetR\ ]_b \xrightarrow{c_8} [\ LacI + LacI.tetR\ ]_b$	$c_8 = 9\ s^{-1}$
r_9 :	$[\ LacI.tetR\ ]_b \xrightarrow{c_9} [\ LacI.tetR + mtetR\ ]_b$	$c_9 = 5 \times 10^{-4}\ s^{-1}$
r_{10} :	$[\ LacI_2.tetR\ ]_b \xrightarrow{c_{10}} [\ LacI_2.tetR + mtetR\ ]_b$	$c_{10} = 5 \times 10^{-4}\ s^{-1}$
r_{11} :	$[\ tetR\ ]_b \xrightarrow{c_{11}} [\ tetR + mtetR\ ]_b$	$c_{11} = 0.5\ s^{-1}$
r_{12} :	$[\ mtetR\ ]_b \xrightarrow{c_{12}} [\ ]_b$	$c_{12} = 5.78 \times 10^{-3}\ s^{-1}$
r_{13} :	$[\ mtetR\ ]_b \xrightarrow{c_{13}} [\ mtetR + TetR\ ]_b$	$c_{13} = 0.167 \times 10^{-4}\ s^{-1}$
r_{14} :	$[\ TetR\ ]_b \xrightarrow{c_{14}} [\ ]_b$	$c_{14} = 1.16 \times 10^{-3}\ s^{-1}$
r_{15} :	$[\ TetR + cI\ ]_b \xrightarrow{c_{15}} [\ TetR.cI\ ]_b$	$c_{15} = 1\ s^{-1}$
r_{16} :	$[\ TetR.cI\ ]_b \xrightarrow{c_{16}} [\ TetR + cI\ ]_b$	$c_{16} = 224\ s^{-1}$
r_{17} :	$[\ TetR + TetR.cI\ ]_b \xrightarrow{c_{17}} [\ TetR_2.cI\ ]_b$	$c_{17} = 1\ s^{-1}$
r_{18} :	$[\ TetR_2.cI\ ]_b \xrightarrow{c_{18}} [\ TetR + TetR.cI\ ]_b$	$c_{18} = 9\ s^{-1}$
r_{19} :	$[\ TetR.cI\ ]_b \xrightarrow{c_{19}} [\ TetR.cI + mcI\ ]_b$	$c_{19} = 5 \times 10^{-4}\ s^{-1}$
r_{20} :	$[\ TetR_2.cI\ ]_b \xrightarrow{c_{20}} [\ TetR_2.cI + mcI\ ]_b$	$c_{20} = 5 \times 10^{-4}\ s^{-1}$
r_{21} :	$[\ cI\ ]_b \xrightarrow{c_{21}} [\ cI + mcI\ ]_b$	$c_{21} = 0.5\ s^{-1}$
r_{22} :	$[\ mcI\ ]_b \xrightarrow{c_{22}} [\ ]_b$	$c_{22} = 5.78 \times 10^{-3}\ s^{-1}$
r_{23} :	$[\ mcI\ ]_b \xrightarrow{c_{23}} [\ mcI + CI\ ]_b$	$c_{23} = 0.167 \times 10^{-4}\ s^{-1}$
r_{24} :	$[\ CI\ ]_b \xrightarrow{c_{24}} [\ ]_b$	$c_{24} = 1.16 \times 10^{-3}\ s^{-1}$
r_{25} :	$[\ CI + lacI\ ]_b \xrightarrow{c_{25}} [\ CI.lacI\ ]_b$	$c_{25} = 1\ s^{-1}$
r_{26} :	$[\ CI.lacI\ ]_b \xrightarrow{c_{26}} [\ CI + lacI\ ]_b$	$c_{26} = 224\ s^{-1}$
r_{27} :	$[\ CI + CI.lacI\ ]_b \xrightarrow{c_{27}} [\ CI_2.lacI\ ]_b$	$c_{27} = 1\ s^{-1}$
r_{28} :	$[\ CI_2.lacI\ ]_b \xrightarrow{c_{28}} [\ CI + CI.lacI\ ]_b$	$c_{28} = 9\ s^{-1}$
r_{29} :	$[\ CI.lacI\ ]_b \xrightarrow{c_{29}} [\ CI.lacI + mlacI\ ]_b$	$c_{29} = 5 \times 10^{-4}\ s^{-1}$
r_{30} :	$[\ CI_2.lacI\ ]_b \xrightarrow{c_{30}} [\ CI_2.lacI + mlacI\ ]_b$	$c_{30} = 5 \times 10^{-4}\ s^{-1}$

during an interval of time for different number of repressors. The percentage of the occupancy time for both mechanisms is shown in Figure 5. These results show that in the case of cooperative repression the system exhibits a switch-like behaviour. On the one hand, for small number of repressors the gene will seldom become fully repressed. On the other hand, as soon as a threshold of around ten repressors is exceeded the gene will be drastically repressed. In contrast in the case of non-cooperative repression the system shows a more linear and sluggish repression efficiency for different number of repressors. Therefore, cooperative repression should be chosen over non-cooperative repression if one wants to engineer a system with switch-like dynamics as it is the case of the *Repressilator*.

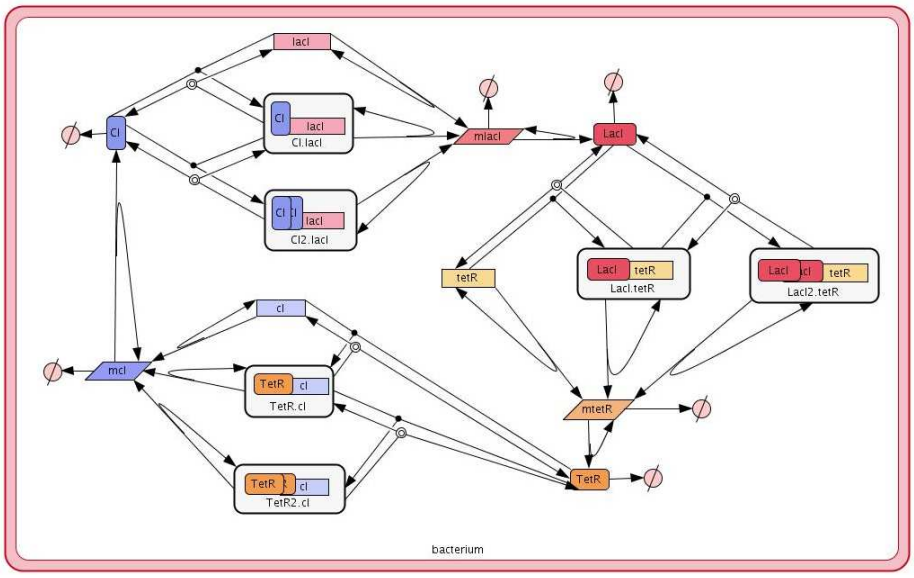

Fig. 3 Graphical representation of the genetic circuit forming the Repressilator. Genes are depicted as rectangles, messenger RNAs as rhomboids and proteins as rounded rectangles. Transcription factor binding and debinding are represented using arrows with filled circles and arrows with concentric circles respectively. Transcription and translation are represented with double headed arrows to describe the fact the reactants are not consumed in the reaction. Finally, the empty set symbol is used to refer to the degradation.

5 Discussions

In this section are summarised the most important common features of the two models presented and significant differences that make them particularly appropriate for certain applications.

Both metabolic P systems and stochastic P systems rely on the same basic formalism inspired by the structure and functionality of the living cell, the membrane system. In its initial form the membrane system concept utilised a maximal parallel approach as its main scheduling mechanism governing the execution of the system. According to this mechanism, in every single step as many symbol objects as possible are processed, by using rules in a non-deterministic manner. Although this principle proves to be very effective for the computational power and efficiency of P systems, it shows its limits in many circumstances when various problems and applications require quite different scheduling mechanisms. Both approaches presented in this paper have considered different scheduling mechanisms that distinguish them from the main body of research dealing with maximal parallelism and non-determinism.

Another common feature of the variants of P systems considered in this paper is given by the applicability area of them; both surfaced as appropriate models of molecular and cellular interactions. In both cases, together with a comprehensive theoretical basis, a sound methodology to model biological systems is developed in this framework and adequate tools have been produced to help analysing and simulating these systems. Both have shown a powerful capability to interact with existing verification techniques and tools or suggest a great potential in using them.

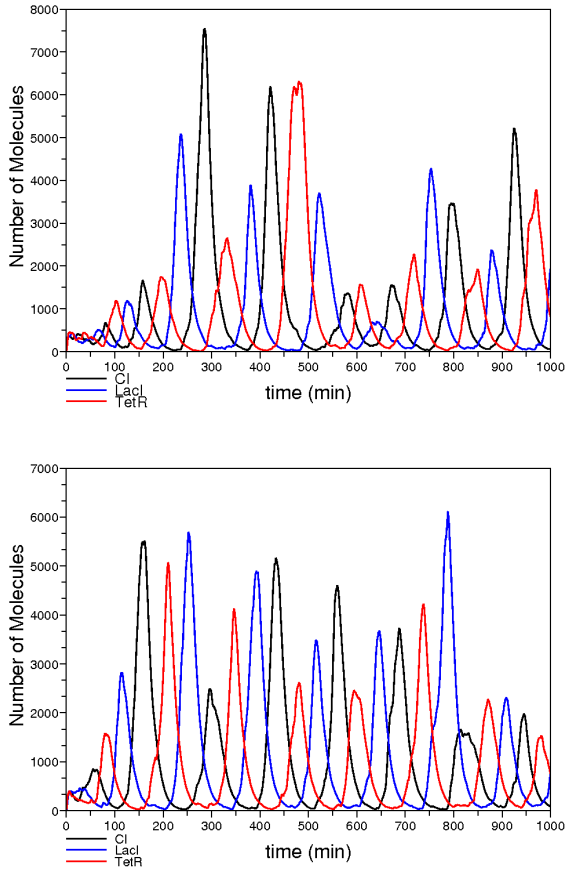

Fig. 4 Two independent simulations of the behaviour of the Repressilator

Despite having common rootes and ancestors, these approaches present clear differences and capabilities for expressing various problems. Metabolic P systems are very appropriate for modelling molecular interactions, exhibiting a deterministic behaviour by making use of appropriate flux (regulation) maps that govern the associated scheduling mechanisms. The expressions of these maps are obtained from other models, like differential equations which, for instance, represent the key modelling paradigm of the biological systems with a deterministic behaviour. One of the most relevant features of MP systems is the log-gain theory, developed in the context of these systems [19,21, 20], by means of which MP models of biological phenomena can be deduced from suitable macroscopic observations of their state time series. On the other hand, stochastic P systems use a well-established stochastic scheduling mechanism based on Gillespie algorithm. They have been successfully involved in modelling systems with a stochastic behaviour, especially gene regulatory networks where the level of noise is high and the degree of complexity is significant.

Stochastic P systems have been linked to various formal approaches in software engineering which become more intensively and extensively used in analysing the be-

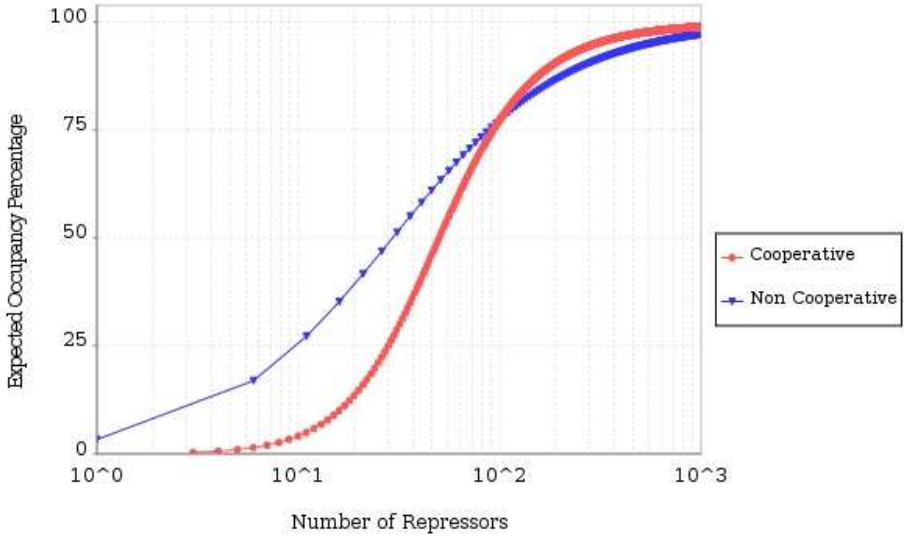

Fig. 5 Expected occupancy percentage of the promoter for different number of repressors.

haviour of biological systems. As the complexity of such biosystems is very high, various modular approaches have been provided for stochastic P systems together with methods using evolutionary and genetic algorithms for the automated design of certain classes of P systems that suit biological systems specifications. The problem of parameter estimation and optimisation has been also investigated.

Metabolic P systems make use of regression analysis and optimisation techniques to set up the flux maps that support the rewriting process. They appear to be particularly appropriate to specify the dynamics of metabolic systems, whereas stochastic P systems represent an adequate tool for modelling hierarchical systems, especially signalling pathways.

6 Conclusions

Deterministic metabolic and stochastic P systems both contribute to building a consistent and mature research area relying on a strong theoretical basis rooted in formal languages and automata, with successful modelling paradigms, significant links with other computational methods, like formal verification and analysis, and a rich portfolio of case studies.

In the near future it is expected that more extensive case studies will be developed to illustrate the benefits of using these models in systems in synthetic biology. A deeper connection between these models and various formal verification methods will be developed that will also emphasise the role of reverse engineering in identifying properties and patterns of behaviour from various simulations. More steps will be taken toward studying systems with noisy data or unknown behaviour where the interplay between machine learning techniques, evolutionary approaches and modelling

capabilities of various variants of P systems will show their efficiency and suitability in specifying large biosystems.

Acknowledgement. The authors would like to thank the anonymous referees for their valuable comments. FJRC would like to acknowledge EPSRC grant EP/E017215/1 and BBSRC grants BB/F01855X/1 and BB/D019613/1.

References

1. B. Alberts and M. Raff (1997) Essential cell biology. An introduction to the molecular biology of the cell. Garland Science, New York
2. S.A. Benner, M. Sismour (2005) Synthetic biology, Nature Reviews Genetics, 6, 533–543
3. F. Bernardini, M. Gheorghe, N. Krasnogor (2007) Quorum sensing P systems. Theoretical Computer Sci, 371, 1–2, 20–33
4. F. Bernardini, M. Gheorghe, F.J. Romero-Campero, N. Walkinshaw (2007) A hybrid approach to modeling biological systems. Lecture notes in computer science, vol 4860, Springer, 138–159
5. F. Bernardini, V. Manca (2003) Dynamical aspects of P systems. BioSystems, 70, 2, 85–93
6. L. von Bertalanffy (1967) General systems theory: Foundations, developments, applications. George Braziller Inc., New York
7. A. Castellini, V. Manca (2008) MetaPlab: A computational framework for metabolic P systems. Pre-proceedings WMC 2008, Edinburgh, UK
8. A. Castellini, V. Manca, L. Marchetti (2008) MP systems and hybrid Petri nets. Studies in Computational Intelligence, 129, 53–62
9. G. Ciobanu, Gh. Păun, M. J. Pérez-Jiménez, eds. (2006) Applications of membrane computing. Springer
10. M.B. Elowitz, S. Leibler (2000) A synthetic oscillatory network of transcriptional regulators. Nature, 403, 20, 335–338
11. M. Ernst, J. Cockrell, W. Griswold, D. Notkin (2001) Dynamically discovering likely program invariants to support program evolution. IEEE Transactions on Software Engineering, 27, 2, 99–123
12. F. Fontana, V. Manca (2007) Discrete solutions to differential equations by metabolic P systems. Theoretical Computer Sci, 372, 2–3, 165–182
13. D.T. Gillespie (2007) Stochastic simulation of chemical kinetics. Annual Review of Physical Chemistry, 58, 35–55
14. A. Goldbeter (1991) A minimal cascade model for the mitotic oscillator involving cyclin and cdc2 kinase. PNAS, 88, 9107–9111
15. N. Krasnogor, M. Gheorghe, G. Terrazas, S. Diggle, P. Williams, M. Camara (2005) An appealing computational mechanism drawn from bacterial quorum sensing. Bulletin of the EATCS, 85, February, 135–148
16. M. Kwiatkowska, G. Norman, D. Parker (2005) Probabilistic model checking in practice: Case studies with PRISM. ACM SIGMETRICS Performance Evaluation Review, 32, 4, 16–21
17. V. Manca (2007) Metabolic P systems for biochemical dynamics. Progress in Natural Sciences, 17, 4, 384–391
18. V. Manca (2008) Discrete simulations of biochemical dynamics. Lecture notes in computer science, vol 4848, Springer, pp. 231–235
19. V. Manca (2008) The metabolic algorithm for P systems: Principles and applications. Theoretical Computer Sci, 404, 1-2, 142–157
20. V. Manca (2009) Fundamentals of Metabolic P Systems. In: Gh. Paun, G. Rozenberg, A. Salomaa (eds.), Handbook of Membrane Computing, CHAPTER 16, Oxford University Press, 2009
21. V. Manca (2008) Log-gain principles for metabolic P systems. *Rozenberg's Festschrift*, Natural Computing Series, Springer, in print
22. V. Manca, L. Bianco (2008) Biological networks in metabolic P systems. BioSystems, 91, 3, 489–498
23. V. Manca, L. Bianco, F. Fontana (2005) Evolutions and oscillations of P systems: Applications to biological phenomena. Lecture notes in computer science, vol 3365, Springer, pp. 63–84

24. V. Manca, R. Pagliarini, S. Zorzan (2008) A photosynthetic process modelled by a metabolic P system. Natural Computing, to appear, DOI 10.1007/s11047-008-9104-x
25. Gh. Păun (2000) Computing with membranes. J. Comput. System Sci., 61, 108–143
26. Gh. Păun (2002) Membrane computing. An introduction. Springer, 2002
27. M.J. Pérez-Jiménez, F.J. Romero-Campero (2006) P Systems, a new computational modelling tool for systems biology. Transactions on Computational Systems Biology VI, LNBI, 4220, 176–197
28. D. Pescini, D. Besozzi, G. Mauri, C. Zandron (2006) Dynamical probabilistic P systems. Intern. J. Found. Computer Sci., 17, 183–204
29. F.J. Romero-Campero, H. Cao, M. Cámara, N. Krasnogor (2008) Structure and parameter estimation for cell systems biology models. Proc. of the Genetic and Evolutionary Computation Conference, July 12 - 16, Atlanta, USA, 331–338
30. F.J. Romero-Campero, M. Gheorghe, G. Ciobanu, J.M. Auld, M.J. Pérez-Jiménez (2007) Cellular modelling using P systems and process algebra. Progress in Natural Sci, 17, 4, 375–383
31. F.J. Romero-Campero, M. Gheorghe, G. Ciobanu, L. Bianco, D. Pescini, M.J. Pérez-Jiménez, R. Ceterchi (2006) Towards probabilistic model checking on P systems using PRISM. Lecture notes in computer science, vol 4361, Springer, pp. 477–495
32. F.J. Romero-Campero, M.J. Pérez-Jiménez (2008) A model of the quorum sensing system in Vibrio fischeri using P systems. Artificial Life, 14, 1, 1–15
33. F.J. Romero-Campero, J. Twycross, M. Cámara, M. Bennett, M. Gheorghe, N. Krasnogor (2008) Modular assembly of cell systems biology models using P systems, submitted
34. The P Systems Web Site (2008) `http://ppage.psystems.eu`