

Cellular Modelling Using P Systems and Process Algebra

Francisco J. Romero-Campero¹, Marian Gheorghe²
Gabriel Ciobanu³, John M. Auld², Mario J. Pérez-Jiménez¹

¹Research Group on Natural Computing

Department of Computer Science and Artificial Intelligence
University of Seville, Avda. Reina Mercedes, 41012 Sevilla, Spain

Email: {fran,marper}@us.es

²Department of Computer Science, The University of Sheffield

Regent Court, Portobello Street, Sheffield S1 4DP, UK

Email: {M.Gheorghe@dcs.shef.ac.uk, john.auld@bcs.org}

³Romanian Academy, Institute of Computer Science

Blvd. Carol I nr.11, 700505, Iasi, Romania

Email: gabriel@itt.tuiasi.ro

Abstract

In this paper various molecular chemical interactions are modelled under different computational paradigms. P systems and π -calculus are used to describe intra-cellular reactions like protein-protein interactions and gene regulation control.

1 Introduction

The more we learn about cell systems, the less, paradoxically, we seem to understand. Most systems in the living cell involve the interaction of multiple components and subsystems. This complexity makes increasingly difficult to visualise any system as a whole and requires models that can be used as tools to disentangle these apparent messy interactions.

Models, abstractions of the reality onto a mathematical/computational formalism, should not be seen or presented as representations of the truth, but instead as a statement of our current knowledge of the phenomenon under research. A model is often more useful when proved to be wrong, since it shows that our current understanding of the phenomenon studied does not match the reality. Thus, it helps experimentalists as a way to decide which experiments are necessary to advance understanding [5].

A good model should at least have four properties: relevance, computability, understandability and extensibility [19]. A model must be relevant capturing the essential properties of the phenomenon investigated; and computable so it can allow the simulation of its dynamic behaviour, and the qualitative and quantitative reasoning about its properties. An understandable model will correspond well to the informal concepts and ideas of molecular biology. Finally, a good model should be extensible to higher levels of organisations, like tissues, organs, organisms, etc. in which molecular systems play a key role. There are primarily three uses of models in science: understanding, integration of partial knowledge and discovery of new features; prediction, capability to estimate the dynamics of a system; and control, to constrain or manipulate a system to produce a desirable output or behaviour. These objectives can be more easily obtained if there is a close structural connection between biological data and the model itself. By structural we mean that the model itself reflects the structure of the system being studied [5].

The classical approach to modelling is based on mathematical theory of differential equations. Differential equations have been used successfully to model kinetics of conventional macroscopic chemical reactions. Nevertheless there is an implicit assumption of continuously varying chemical concentration and deterministic dynamics. Two critical characteristics of this approach are that the number of molecules of each type in the reaction mix is large and that for each type of reaction in the system, the number of reactions is large within each observation interval, that is reactions are fast.

When the number of particles of the reacting species are low and reactions are slow, which is frequently the case in gene expression control in bacteria and viruses, both of the previous presumptions are invalid and the deterministic continuous approach to chemical kinetics is questionable. Instead one has to recognise that the individual chemical reaction steps occur discretely and are separated by time intervals of random length.

Previous attempts to model cellular systems from a computational point of view include Petri nets, agent-based approach, L-systems, state charts, process algebra, etc. While each of these approaches captures some of the information regarding pathways and their molecular components, none fully integrates quantitative dynamics, interactions among molecular entities and structural organisation of cells.

Membrane computing is an emergent branch of natural computing introduced by G. Păun in [13]. This new model of computation starts from the assumption that the processes taking place in the compartmental structure of a living cell can be interpreted as computations. The devices of this model are called P systems. Roughly speaking, a P system consists of a cell-like membrane structure, in the compartments of which one places multisets of objects which evolve according to given rules.

Although most research in membrane computing concentrates on the computational power and efficiency of the devices involved, lately they have been used to model biological phenomena within the framework of computational systems biology being complementary and an alternative to more classical approaches

like ordinary differential equations (ODEs) and to some recent computational approaches like Petri nets and π -calculus. In this case P systems are not used as a computing paradigm, but rather as a formalism for describing the behaviour of the system to be modelled. In this respect several P system models have been proposed to describe oscillatory systems [8], signal transduction [15], gene regulation control [16], quorum sensing [21] and metapopulations [14]. These models differ in the syntax (type of the rewriting rules, membrane structure) and the semantic (strategy applied to run the rules in the compartments defined by membranes). Some of these models using *metabolic algorithm* [2], *dynamical probabilistic P systems* [14] and *(multicompartmental) Gillespie Algorithm* [15] have been applied in certain case studies [23].

The π -calculus was introduced by Milner, Parrow, and Walker as a formal language to describe mobile concurrent processes [12]. It is now a widely accepted model for interacting systems with dynamically evolving communication topology. The π -calculus allows channels to be passed as data along other channels, and this fact provides a channel mobility. This mobility is an important feature and increases the expressive power. The π -calculus has a simple semantics and a tractable algebraic theory. Starting with atomic actions and simpler processes, complex processes can be constructed in many ways. The process expressions are defined by guarded processes, parallel composition $P|Q$, non-deterministic choice $P + Q$, replication $!P$, and a restriction operator $(\nu x)P$ creating a local fresh channel x for a process P . A structural congruence relation providing a static semantics is defined over the set of processes. The evolution of a process is described in π -calculus by a reduction relation over processes. This relation contains those transitions which can be inferred from a set of rules. Different variants have been used to model molecular interactions [18], gene networks, and to integrate molecular and gene networks [4].

In the last years there have been attempts to relate π -calculus to membrane systems. In [7], the transfer mechanisms involving Na-K pump are described step by step and formal verification tools are used to check the validity of the modelling approach. In [1], the functioning of the same pump is described and analysed in the formal framework of P systems. New features such as variable membrane labelling, activation conditions for rules, membrane bilayer and specific communication rules are defined and utilised to model specific transfer approaches.

In this paper we will present models developed within the framework of membrane computing and process algebra and addressing molecular interactions. The aim of the approach is to show differences between the above mentioned modelling strategies and their suitability to express local molecular interactions. The P system approach relies on Gillespie's Algorithm [9] and uses the tool developed in this respect [22].

An extended version of this paper presents also inter-cellular interactions and a suitable graphical representation for the processes involved that is supported by a standard notation [22].

2 Modelling Single Compartment Systems

Most models of cellular processes so far consists of systems of molecules which interact through a set of chemical reactions taking place inside a single fixed volume or compartment which is assumed to be a well-stirred solution.

There are primarily two different kinds of interactions inside a single compartment: those known as protein-protein interactions which comprise basically transformation, degradation, complex formation and dissociation (dimerisation); and gene regulation interactions comprising fundamentally constitutive expression, positive and negative regulation.

In what follows we will present how these different types of reactions are modelled in P systems and π -calculus. Results will be contrasted with those obtained using ODE. A general reference for biological concepts used in this paper is [11].

Protein-protein interactions:

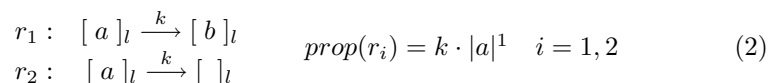
- Transformation and degradation:

A molecule a can react to produce another molecule b or it can be degraded by the cell machinery. The dynamics of these reactions can be modelled using the *exponential decay law*. This law states that the rate of the reaction or its propensity is proportional to the number of molecules of the reactant a .

Classically, these reactions have been modelled using ODEs where the decay of concentration of the substance a , represented by a variable A , is described by the differential equation in (1).

$$\frac{dA}{dt} = -kA \quad (1)$$

In P systems, transformation and degradation are represented using rewriting rules where the object a is replaced with the object b or is simply removed in the case of degradation. The compartment where the molecules are located is also specified using square brackets with a label l which identifies the corresponding compartment. A constant k is associated with the rule so that its propensity can be computed:



A process is used to specify each molecule of type a and b in π -calculus. First order reactions are specified using stochastic delays whereas higher order reactions are specified using communication channels through which the processes representing the reactant molecules interact. In transformation and degradation, a stochastic delay τ_k is associated with the process a . After this delay the

¹ $|a|$ stands for the number of objects a in the case of P systems or the number of processes a in the case of process algebra.

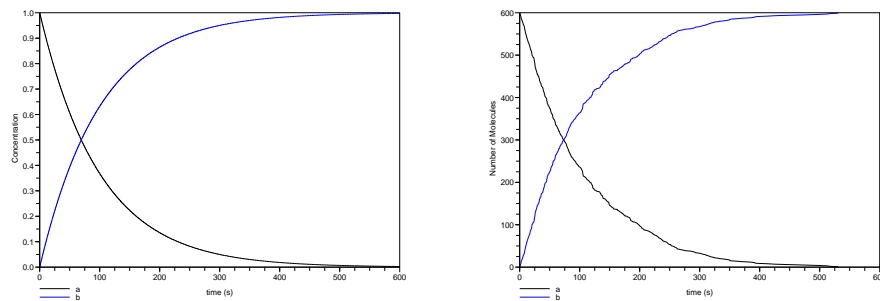


Figure 1: Transformation dynamics for $k = 10^{-3} s^{-1}$.

process a behaves either as the process b , in the case of transformation; or as the empty process in the case of degradation.

$$\begin{aligned} a &:= \tau_k \cdot b \\ a &:= \tau_k \cdot 0 \end{aligned} \quad \text{prop}(\tau_k) = k \cdot |a| \quad (3)$$

Figure 1 shows the time courses obtain using the differential equation in (1), left, and the P systems rules in (2), right. Observe the characteristic exponential decay and increase of the reactant a and the product b .

- Complex formation and dissociation:

Two molecules, a and b , can collide and stick together through noncovalent interactions to produce a complex c . This complex in turn can dissociate back into a and b . In biochemistry, these reactions are referred as complex formation, more specifically heterodimer formation when $a \neq b$ and homodimer formation when $a = b$; and complex dissociation.

Many important cellular processes depend on complex formation and dissociation, since the binding of a molecule to another one can alter (regulate) the activity of the complex which can be completely different from the activity of the single molecules.

The dynamics of these reactions follow the *mass action law*, which states that the rate or propensity of the reaction is directly proportional to the product of the number of the reactant molecules. Thus, two constants k_1 and k_2 are associated with the complex formation and dissociation reactions respectively so their rates or propensities can be computed.

This two chemical reactions lead to the three differential equations in (4), one for the concentration of each chemical species a , b and c . Following the *mass action law* the concentration of the reactants a and b , represented by the variables A and B , decrease according to the term $k_1 AB$ (complex formation); whereas they increase according to the term $k_2 C$ (complex dissociation), where

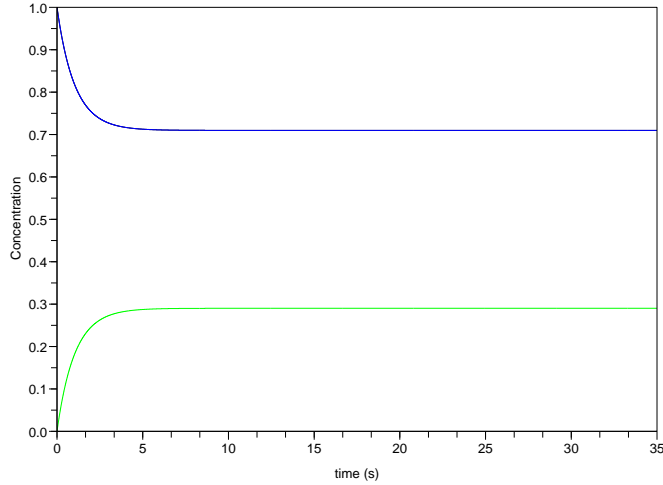


Figure 2: Time course of dimerisation.

C represents the concentration of c . On the contrary, the concentration of the complex c is increased by k_1AB and decreased by k_2C .

$$\begin{aligned}
 \frac{dA}{dt} &= -k_1AB + k_2C \\
 \frac{dB}{dt} &= -k_1AB + k_2C \\
 \frac{dC}{dt} &= k_1AB - k_2C
 \end{aligned}
 \tag{4}$$

The graph in Figure 2 represents the time course of dimerisation obtained after solving the systems of ODEs in (4) for $k_1 = 0.3\mu M^{-1}s^{-1}$, $k_2 = 0.5s^{-1}$ and initial conditions $[A] = [B] = 1\mu M$ and $[C] = 0\mu M$.

In P systems, the complex formation and dissociation reactions are specified using the rewriting rules in (5) which take the same name as the reactions they represent. In the complex formation rule the objects a and b , representing the corresponding molecules, are replaced with the object c , representing the complex. In the same way, in the complex dissociation rule, the object c is replaced with the objects a and b . The compartment in which the reactions take place is specified using square brackets and a label l .

The mesoscopic constants $k'_1 = 0.048molec^{-1}s^{-1}$ and $k_2 = 0.5s^{-1}$, computed from the macroscopic ones, k_1 and k_2 , according to [9], are associated with the corresponding rules in order to compute their propensities.

$$\begin{aligned}
r_1 : [a + b]_l &\xrightarrow{k'_1} [c]_l & \text{prop}(r_1) &= \begin{cases} k'_1 \cdot |a||b| & \text{if } a \neq b \\ k'_1 \cdot \frac{|a|(|a|-1)}{2} & \text{if } a = b \end{cases} & (5) \\
r_2 : [c]_l &\xrightarrow{k_2} [a + b]_l & \text{prop}(r_2) &= k_2 \cdot |c|
\end{aligned}$$

The processes a , b and c specify the behaviour of the corresponding molecules in the π -calculus formalism in (6). The communication channel $bind_{k'_1}$ represents the complex formation reaction; whereas the stochastic delay τ_{k_2} represents the complex dissociation reaction.

The processes a and b have complementary communication channels, $bind_{k'_1}?$ and $bind_{k'_1}!$, through which they interact with the rate characterised by the constant k'_1 ; after the communication the process a behaves as the process c , which represents the complex, and the process b is replaced with the empty process. The complex dissociation reaction is specified in the process c using the stochastic delay τ_{k_2} . Once this delay is completed the process c is replaced with the processes a and b representing the unbound molecules.

$$\begin{aligned}
a &:= bind_{k'_1}?.c & \text{prop}(bind_{k'_1}) &= k'_1 \cdot |a||b| \\
b &:= bind_{k'_1}!.0 & & \\
c &:= \tau_{k_2}.(a|b) & \text{prop}(\tau_{k_2}) &= k_2 \cdot |c|
\end{aligned} \tag{6}$$

The case of homodimer formation (complex formation with two molecules of the same chemical specie) is treated in π -calculus in a different way. The processes a , representing molecules of type a , have both complementary channels $bind_{\frac{k'_1}{2}}?$ and $bind_{\frac{k'_1}{2}}!$. Observe that, in this case the constant is halved since the natural encoding of heterodimer formation in π -calculus produces an artificial doubling of interactions $|a|(|a|-1)$ instead of $\frac{|a|(|a|-1)}{2}$, hence by halving the constant we compensate this replication in order to get the expected propensity, $k'_1 \cdot \frac{|a|(|a|-1)}{2}$.

$$\begin{aligned}
a &:= bind_{\frac{k'_1}{2}}?.c + bind_{\frac{k'_1}{2}}!.0 \\
c &:= \tau_{k_2}.(a|a)
\end{aligned} \tag{7}$$

In Figure 3 we study the effect on the dynamics of dimerisation of the volume of the compartment where the reactions take place. The volumes considered are that of a golgi body $\approx 10^{-16}l$ and yeast cell $\approx 10^{-14}l$. The constant associated to the complex formation rule for the different volumes are $k'_1 = 4.8 \times 10^{-3} molec^{-1} s^{-1}$ and $k'_1 = 4.8 \times 10^{-5} molec^{-1} s^{-1}$; the constant associated to the complex dissociation does not depends on the volume of the compartment and so it does no change according to [9].

Note that the smaller the volume is the higher the level of noise. Observe the noisy behaviour in the golgi body (Figure 3 left) and how the dynamics gets smoother in a yeast cell as the volume increases and so the number of molecules (Figure 3 right). In the case of infinitive volume (considered by ODE models), and large number of molecules, the time series given by the ODEs in

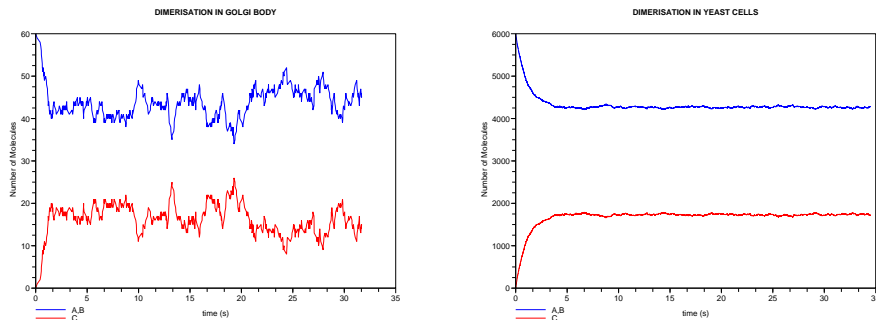


Figure 3: Evolution of the number of molecules over time for complex formation and dissociation dynamics in a golgi body (left) and yeast cell (right).

(4) and plotted in Figure 2 will describe faithfully the dynamics in dimerisation. Nevertheless, for small volumes, as it is the case in the organella of eukaryotic cells, the approach based on ODEs is questionable.

Although the level of noise differs considerably in different volumes, the equilibrium is reached approximately at the same point in time.

Finally, highlight that these results support the use of different Gillespie's algorithms (multicompartmental Gillespie's algorithm [15]) in the compartments of a cell system, since the difference in volume highly influences the level of noise of the system.

Gene regulation control:

We now discuss how gene regulation control in prokaryotes can be specified in different formalisms. We deal only with prokaryotes since we are focussing on single compartment systems. In contrast to prokaryotic cells, in eukaryotes there are two compartments involved in gene regulation, namely, the nucleus, where transcription and more complex processes like RNA splicing take place, and the cytoplasm where, for example, post transcriptional control and translation occur.

In this paper for simplicity transcription and translation are represented as individual reactions. Nevertheless, in living cells transcription and translation involve many interactions between RNA polymerase, DNA, mRNA and ribosomes that take place in a concurrent manner; for example, before a gene has been completely transcribed, ribosomes can start transcribing the growing mRNA. In the framework of P systems and π -calculus, transcription and translation are explicitly represented as taking place in parallel (see [16] and [10] respectively).

The central dogma of molecular biology states that genetic information is stored in the DNA, transcribed into messenger RNA (mRNA), and then trans-

lated into proteins. This picture is much more complicated since certain proteins, called transcription factors, acts as regulators in the transcriptions of genes, either positively or negatively; that is, an increase in the amount of transcription factor leads to either more or less gene expression. This provides a feedback pathway by which genes can regulate the expression of other genes and so of the production of the proteins encoded by them.

- Constitutive expression:

We start by modelling genes whose level of expression does not depend on transcription factors. These genes are called, *constitutive genes* or *housekeeping genes*. This kind of genes are transcribed continually at a relatively constant level compared to *facultative genes*, which are only transcribed when needed.

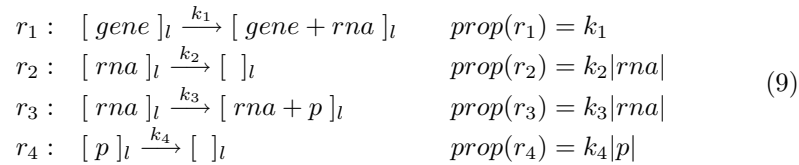
In this case from the gene encoded in the DNA the mRNA is transcribed. This mRNA is then translated into the protein product associated to the gene and the mRNA is also degraded by the cell machinery.

This system is represented by the two differential equations in (8), where the variable RNA represents the concentration of mRNA and P represents the concentration of the protein product encoded by the gene. Transcription is assumed to take place at a constant rate k_1 . Degradation of the mRNA follows, as before, the *exponential decay law* characterised by the term k_2RNA . Translation is assumed to be proportional to the concentration of mRNA, k_3RNA . Finally, protein degradation is modelled in the term k_4P [3].

$$\begin{aligned} \frac{dRNA}{dt} &= k_1 - k_2RNA \\ \frac{dP}{dt} &= k_3RNA - k_4P \end{aligned} \tag{8}$$

Figure 4 (left) shows the solution of the differential equations in (8). Note that the protein concentration reaches equilibrium and the cell keeps it constant to that value. Also observe that the protein concentration is considerably higher than the mRNA concentration.

The processes of transcription, translation and degradation are represented in P systems by rewriting rules with the same names as the processes they represent, see (9). Note that, since all these reactions are assumed to be first order reactions, we can use the same constants as in differential equations according to [9].



The time courses obtained when simulating the P system rules in (9) are represented in Figure 4 (right). Note that these results are in accordance with the

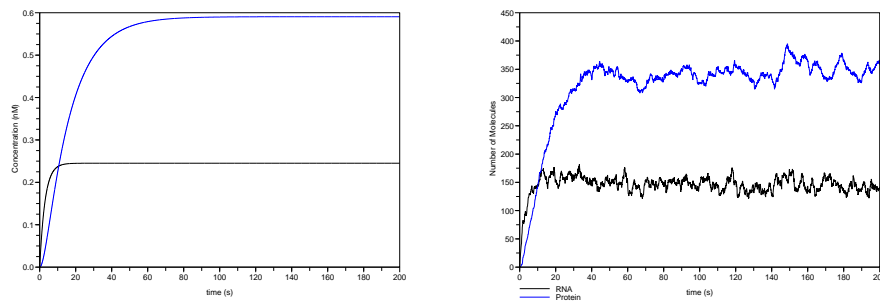


Figure 4: Time course of constitutive gene expression (left) dynamics of constitutive gene expression by using P systems (right).

previous results from differential equations plotted in Figure 4 (left). Nonetheless ODEs do not capture the noise, we can see in Figure 4 (right), due to the low number of molecules. This type of noise is also referred in physics as *shot noise*; this noise occurs when the finite number of particles that carry energy is small enough to give rise to detectable statistical fluctuations in a measurement [20].

In the π -calculus specification in (10), the process *gene* represents transcription by producing the processes *gene* and *rna* after a stochastic delay τ_{k_1} . In the process *rna* the stochastic delay τ_{k_2} and τ_{k_3} represent translation and degradation respectively. Finally, the process *p* represents the protein product of the gene and specifies degradation using the stochastic delay τ_{k_4} .

$$\begin{aligned}
 gene &:= \tau_{k_1} \cdot (gene|rna) & prop(\tau_{k_1}) &= k_1 \\
 rna &:= \tau_{k_2} \cdot 0 + \tau_{k_3} \cdot (rna|p) & prop(\tau_{k_2}) &= k_2|rna| \\
 p &:= \tau_{k_4} \cdot 0 & prop(\tau_{k_3}) &= k_3|rna| \\
 & & prop(\tau_{k_4}) &= k_4|p|
 \end{aligned} \tag{10}$$

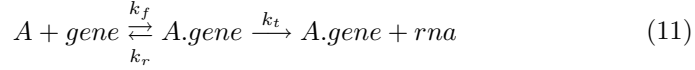
Constitutive gene expression is modelled in a similar way using π -calculus in [4]; although, in contrast to (10), transcription and translation are not modelled explicitly, instead the protein product is produced from the gene after a stochastic delay.

- Positive regulation:

Unlike *constitutive genes*, *facultative genes* are only expressed when needed according to some signals received by the cell from its surroundings. *Activators* are transcription factors which bind to the promoter of genes and activate their expression by recruiting polymerase.

Michaelis-Menten kinetics are often used in ODEs to model positive regulation [3], of a gene by an activator whose concentration is represented by a variable *A*. The three processes, binding, debindind and transcription, pre-

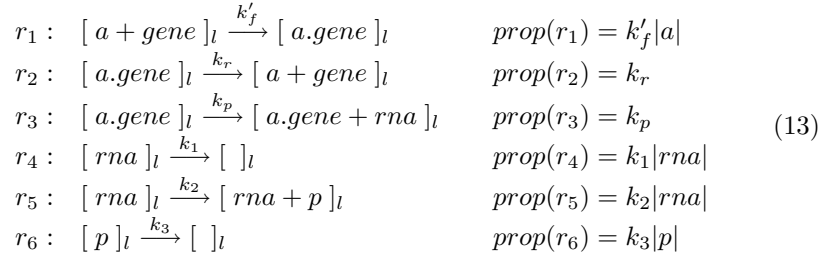
sented in (11), are simplified such that the rate of transcription is assumed to depend on the concentration of the activator A . This is represented on the so called *Michaelis constant* K_m , which reflects the affinity between the activator and promoter, and on the maximum turn-over rate, V_m , which in this case corresponds to the maximum transcription rate. Thus the rate of transcription is given by the term $V_m \frac{A}{K_m + A}$, where $K_m = \frac{k_r + k_t}{k_f}$.



The two equations in (12) model gene positive regulation. For the mRNA we have the corresponding Michaelis-Menten dynamics, $V_m \frac{A}{K_m + A}$, minus the exponential decay, $k_1 RNA$, specifying the mRNA degradation. The rate of translation is assumed to be proportional to the concentration of mRNA, $k_2 RNA$ and protein degradation follow the *exponential decay law*, $k_3 P$.

$$\begin{aligned} \frac{dRNA}{dt} &= V_m \frac{A}{K_m + A} - k_1 RNA \\ \frac{dP}{dt} &= k_2 RNA - k_3 P \end{aligned} \quad (12)$$

Note that this approach does not model explicitly the binding and debinding of the activator to the promoter of the gene and so it does not capture the boolean nature of the activation of genes. These reactions are specified explicitly in P systems using the two first rules in (13) which follow the same dynamics as complex formation and dissociation. The other rules represent transcription, RNA degradation, translation and protein degradation in the same way as in the constitutive expression of genes.



The time courses of positive gene regulation obtained using our simulator, available from [22], and the rules in (13) are depicted in Figure 5 (right). As the case of constitutive expression, our results are in agreement with those of ODEs presented in Figure 5 (left). Nevertheless, ODEs do not represent the level of noise; which is more noticeable here than in the previous case. This is due to the boolean nature of the activation of a gene by an activator which produces the so called burst in transcription. This noise is referred to as *telegraph noise* in physics, in analogy to the telegraph which is either silent or in a sending state as the operator taps.

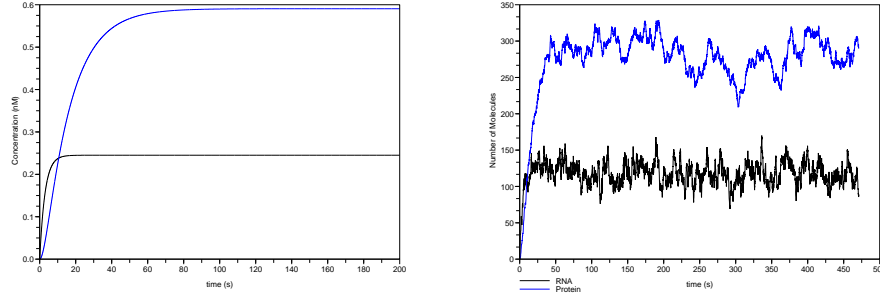


Figure 5: Time course of positive gene expression (left). Dynamics of positive gene regulation using P systems (right).

The π -calculus specification also represents binding and debinding in the same way as complex formation and dissociation using the processes a and $gene$. Transcription, translation and degradation are modelled using similar processes and stochastic delays as in the case of constitutive expression.

$$\begin{aligned}
 a &:= bind_{k'_f}?.0 & prop(bind_{k'_f}) &= k'_f|a| \\
 gene &:= bind_{k'_f}!.a.gene & prop(\tau_{k_r}) &= k_r \\
 a.gene &:= \tau_{k_r}.(a|gene) + \tau_{k_p}.(a.gene|rna) & prop(\tau_{k_p}) &= k_p \\
 rna &:= \tau_{k_1}.0 + \tau_{k_2}.(rna|p) & prop(\tau_{k_1}) &= k_1|rna| \\
 p &:= \tau_{k_3}.0 & prop(\tau_{k_2}) &= k_2|rna| \\
 & & prop(\tau_{k_3}) &= k_3|p|
 \end{aligned} \tag{14}$$

In [4] positive gene expression is modelled using π -calculus; although, unlike (10), binding and debinding of the activator to the gene promoter, transcription and translation are not modelled explicitly, instead a communication channel, through which the processes representing the activator and the gene communicate, is used to produce the protein product directly from the gene.

- Negative regulation:

As opposed to positive regulation in some conditions cells do not need the protein product encoded by a gene; in this case this gene is turned off or repressed by transcription factors called *repressors*. *Repressors* bind to the promoter site of genes blocking it so that polymerase cannot bind to it and thus preventing genes from being transcribed.

In this case Michaelis-Menten dynamics are also used usually to model gene repression or negative regulation [3]. For the mRNA, a basal transcription rate, *leakiness*, is assumed to take place at a rate k_1 similar to constitutive expression. Here the term representing Michaelis-Menten dynamics has negative sign to capture the negative regulation an increase of repressors produces on the

transcription of the gene. Finally, mRNA degradation follows the *exponential decay law* characterised by the constant k_2 .

The dynamics of the protein product of the gene are similar to the case of constitutive expression and positive regulation.

$$\begin{aligned}\frac{dRNA}{dt} &= k_1 - k_1 \frac{R}{K_m + R} - k_2 RNA \\ \frac{dP}{dt} &= k_3 RNA - k_4 P\end{aligned}\tag{15}$$

Similar to positive regulation the P system and π -calculus specifications represent explicitly the binding and debinding of the repressor to the gene. Transcription in the absence of repressor, translation and degradation are modelled as in the previous cases.

$$\begin{aligned}r_1 : [gene]_l &\xrightarrow{k_1} [gene + rna]_l & prop(r_1) &= k_1 \\ r_2 : [r + gene]_l &\xrightarrow{k'_f} [r.gene]_l & prop(r_2) &= k'_f |r| \\ r_3 : [r.gene]_l &\xrightarrow{k_r} [r + gene]_l & prop(r_3) &= k_r \\ r_4 : [rna]_l &\xrightarrow{k_2} []_l & prop(r_4) &= k_2 |rna| \\ r_5 : [rna]_l &\xrightarrow{k_3} [rna + p]_l & prop(r_5) &= k_3 |rna| \\ r_6 : [p]_l &\xrightarrow{k_4} []_l & prop(r_6) &= k_4 |p|\end{aligned}\tag{16}$$

$$\begin{aligned}r &:= bind_{k'_f} ?.0 & prop(bind_{k'_f}) &= k'_f |r| \\ gene &:= bind_{k'_f} !.r.gene + \tau_{k_1}.(gene|rna) & prop(\tau_{k_1}) &= k_1 \\ rna &:= \tau_{k_2}.0 + \tau_{k_3}.(rna|p) & prop(\tau_{k_2}) &= k_2 |rna| \\ p &:= \tau_{k_4}.0 & prop(\tau_{k_3}) &= k_3 |rna| \\ & & prop(\tau_{k_4}) &= k_4 |p|\end{aligned}\tag{17}$$

3 Conclusions

In this paper some fundamental chemical interactions occurring at the cellular level are described by using two modelling approaches, P systems and π -calculus. The interactions analysed in the paper cover intra-cellular level. The capabilities of the two modelling approaches are contrasted with respect to classical continuous models based on ordinary differential equations.

References

- [1] Besozzi, D., Ciobanu, G. (2005) A P System Description of the Sodium-Potassium Pump. In G.Mauri, Gh.Păun, M.J.Perez-Jimenez, G.Rozenberg, A.Salomaa (Eds.): *Membrane Computing WMC5*, Lecture Notes in Computer Science 3365, Springer, 211-223.

- [2] Bianco, L., Fontana, F., Manca, V. (2006) P Systems with Reaction Maps, *International Journal of Foundations of Computer Science*, **17** (1) 27–48.
- [3] Bolouri, H., Davidson, E.H. (2002) Modelling transcriptional regulatory networks, *BioEssays*, (24) 1118–1129.
- [4] Blossey, R., Cardelli, L., Phillips, A. (2006) A Compositional Approach to the Stochastic Dynamics of Gene Networks, *Transactions on Computational Systems Biology IV, LNBI 3939* 99-122.
- [5] Bower, J., Bolouri, H. (2001) Computational Modelling of Genetic and Biochemical Networks.
- [6] Cheruku, S., Păun, A., Romero-Campero, F.J., Pérez-Jiménez, M.J., Ibarra, O.H. Simulating FAS Induced Apoptosis by Using P Systems, to appear.
- [7] Ciobanu, G. (2004) Software verification of the biomolecular systems. In *Modelling in Molecular Biology*, Natural Computing Series, Springer, 40-59.
- [8] Fontana, F., Bianco, L., Manca, V. (2005) P Systems and the Modelling of Biochemical Oscillations, *LNCS*, **3850** 199 – 208.
- [9] Gillespie, D.T. (1977) Exact Stochastic Simulation of Coupled Chemical Reactions, *The Journal of Physical Chemistry*, **81** (25) 2340–2361.
- [10] Kuttler, C. (2006) Simulating bacterial transcription and translation in a stochastic pi calculus. *Transactions on Computational Systems Biology VI, LNBI 4220* 113–149.
- [11] Lodish, H., Berk, A., Zipursky, S.L., Matsudaira, P., Baltimore, D., Darnell, J. (2003) *Molecular Cell Biology*. Fifth Edition, Freeman.
- [12] Milner, R. (1999) *Communication and Mobile Systems: The π -calculus*. Cambridge University Press.
- [13] Păun, Gh. (2000) Computing with Membranes, *Journal of Computer and System Sciences*, **61**(1) 108–143.
- [14] Pescini, D., Besozzi, D., Mauri, G., Zandron, C. (2006) Dynamical probabilistic P systems, *International Journal of Foundations of Computer Science*, **17** (1) 183–195.
- [15] Pérez-Jiménez, M.J., Romero-Campero, F.J. (2006) P Systems, a New Computational Modelling Tool for Systems Biology, *Transactions on Computational Systems Biology VI, LNBI 4220* 176-197.
- [16] Pérez-Jiménez, M.J., Romero-Campero, F.J. Modelling Gene Expression Control Using P Systems: The Lac Operon, A Case Study, to appear.

- [17] Priami, C., Regev, A., Shapiro, E., Silverman, W. (2001) Applications of stochastic process algebra to bioinformatics of molecular processes, *Information Processing Letters* **80** 25–31.
- [18] Regev, A., Shapiro, E. (2002) Cellular abstractions: Cells as computation, *Nature* **419** 343.
- [19] Regev, A., Shapiro, E. (2004) The π -calculus as an abstraction for biomolecular systems, In Gabriel Ciobanu and Grzegorz Rozenberg, editors, *Modelling in Molecular Biology*. Springer Berlin.
- [20] Sarpeshkar, R., Delbruck, T., Mead, C.A. (1993) White noise in MOS transistors and resistors, *IEEE Circuits Devices Mag.*, 2329.
- [21] Terrazas, G., Krasnogor, N., Gheorghe, M., Bernardini, F., Diggle, S., Camara, M. (2005) An Environment Aware P-System Model of Quorum Sensing. *CIE 2005, LNCS*, **3526** 473–485.
- [22] P System Simulator:
<http://www.dcs.shef.ac.uk/~marian/PSimulatorWeb/PSystemMF.htm>
- [23] P System Web Site: <http://psystems.disco.unimib.it/>