Formal Executable Descriptions of Biological Systems

Michele Curti
Pierpaolo Degano
University of Pisa
{curtim,degano}@di.unipi.it

Davide Prandi
Corrado Priami
Paola Quaglia
University of Trento
{prandi,priami,quaglia}@dit.unitn.it

Linda Brodo
University of Sassari
brodo@uniss.it

Abstract

The similarities between systems of living entities and systems of concurrent processes may support biological experiments in silico*. Process calculi offer a formal framework to describe biological systems, as well as to analyse their behaviour, both from a qualitative and a quantitative point of view. A couple of little examples help us in showing how this can be done. We mainly focus our attention on the qualitative and quantitative aspects of the considered biological systems, and briefly illustrate which kinds of analysis are possible. We use a known stochastic calculus for the first example. We then present some statistics collected by repeatedly running the specification, that turn out to agree with those obtained by experiments* in vivo*. Our second example motivates a richer calculus. Its stochastic extension requires a non trivial machinery to faithfully reflect the real dynamic behaviour of biological systems.*

1 Introduction

The recent, often astonishing developments in biology have produced a huge amount of data on the structure of living matter; consider for example the success of the human genome project. Less instead is known on the versatile biological functions that cells and their components display.

So in the last years we have seen a shift from *structure* to *functionality*, and the growth of the so-called *-omics* disciplines within biology. These study particular components of cells in terms of their structure *and* functions. The branch devoted to investigating how genes interact is referred to as *genomics*. Most genes, the basic bricks of life, essentially encode proteins and gene expression corresponds to the production of the encoded protein, in a manner not sufficiently clear yet. *Proteomics* studies the proteins encoded by any gene and the description of their functions. Indeed, the interaction between proteins is the basic machinery driving cell working. Further substances, namely metabolites are involved in the chemistry of cells, and studied within *metabolomics*. Equally important is *transcriptomics*, that investigates how some interactions can produce gene mutation or transcription. All the above disciplines witness the development of a new paradigm, that moves from the classical reductionist approach to a system level understanding of life. It is called *systems biology* [24].

Mapping the human genome would be impossible without computers, algorithms and syntax to model structures: it has been crucial representing DNA as a formal language over a four character alphabet and using search and matching algorithms over strings. Much in the same way, computer science appears to be essential for understanding the behaviour of living organisms: passing from structure to functions amounts to equipping syntax with semantics. We claim then that programming language techniques and tools can be used to model, analyse and simulate the dynamic behaviour of biological systems. More generally, we advocate a convergence between computer science and life science, calling for a paradigm shift also on the computer science side, or at least for a fruitful cross-fertilization between the two disciplines.

The pioneering work by Regev, Silverman and Shapiro [41] brought out the similarities between distributed, concurrent, mobile computer systems (when non ambiguous, just *systems*, for short) and biological systems, e.g. metabolic networks, gene regulatory networks, signalling pathways. Biological systems are made up of millions of biological components that are active simultaneously and that can interact to cooperate towards a common goal. Furthermore, the interactions between components are mainly binary and can occur only if the partners are correctly located (e.g. they are near enough, no membrane is dividing them, the affinity or propensity to interaction is sufficiently high). Finally, the actual interactions may change the future behaviour of the whole system even if they occur locally. All these features describe distributed, concurrent, mobile computer systems as well, except maybe for these artificial systems have a smaller number of compo-

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE

nents. There are various process calculi, e.g. [29, 23, 30, 5] that specify the form and the dynamic behaviour of systems, and that allow for analysing them. We claim that they will turn useful in the biological applicative domain.

Process calculi describe a system in terms of the actions its components, called *processes*, can perform. A few operators combine processes to yield a system, actually a process itself. Among them, parallel composition is peculiar: the processes in parallel can act independently and asynchronously so changing their state, but two of them can also synchronise and exchange a message, when one performs an output action and the other a matching input action. Each operator of the calculus comes equipped with rules governing its dynamics. These rules have a logical flavour and define inductively the *semantics* of the calculus. The semantics formally describe the possible behaviour of a system and is often represented through a directed graph, called a *transition system*. Indeed a transition is deduced within a logical theory and says how a system moves from one configuration to the next one. The paths along a transition system model then the *computations* of the system.

Based on the above shallow presentation, we can put forward the abstraction principle that we implement to model biological systems, assuming to work at a molecular level:

- molecules are abstracted as processes;
- the actual interactions between molecules are represented by communications;
- the affinity of interaction between molecules is modelled through the communication capabilities of the processes representing the interacting molecules;
- the changes on the behaviour of the biological system following an interaction are rendered by corresponding modifications on the overall process, induced by the communication preformed.

A major challenge in describing biological systems is that their temporal/spatial behaviour is heavily affected by quantities representing chemical and physical parameters, like concentration, temperature etc. One cannot therefore avoid integrating qualitative and quantitative aspects of process behaviour. The literature has many proposals, mainly on markovian and stochastic process algebras [19, 21, 22, 10, 2, 3, 34].

Once modelled a biological systems as a process, one can start analysing it by studying the behaviour of the specification. In other words, one performs a "virtual" experiment *in silico*. There are two different ways one can follow.

The first approach is the classical one exploited in the stochastic process algebras field. It consists of first deriving a continuous time Markov chain out of the transition system of the specification. Then one carries out stationary analysis relying on standard mathematical tools. This kind of approach helps in estimating the time and the probability of bio-chemical reactions in biological systems in their steady state.

Our personal experience [1] is that the size of the Markov chains grows very fast, namely exponentially with the size of the biological system under analysis, due to the so-called state explosion problem, typical of process algebras. More powerful techniques are in order, to overcome the limitation on the biological systems practically analyzable.

The second way, namely transient analysis, resembles more closely the way biologists make their experiments. It requires to interpret process calculi as real programming languages and to endow them with a stochastic run-time support so to obtain *simulators* (often based on the stochastic π-calculus [37, 33]). Several computations are then run, representing each a single virtual experiment that simulates the behaviour of the biological system in hand. The computations are inspected to collect the relevant information about, e.g. the occurrences of selected communications or synchronizations, i.e. of reactions. The classical statistical analysis then applies. Promising results have been obtained with this approach, on some interesting biological systems [6, 38, 41, 26]. Most of the simulators used have been developed especially to be applied in the biological domain. A crucial point is that they exploit the Gillespie algorithm [16, 17], that allows to numerically simulate the time evolution of a chemically reacting system, taking into account the randomness in chemical systems. This algorithm is well founded because it is based on the same premise underlying the chemical master equation [18].

This paper is organized as follows. Section 2 reports on VICE, a virtual cell whose behaviour is in surprising accordance with that of (portions of) real prokariotes. With the help of a simple example, in Section 3 we illustrate the formal means needed to model complex formation, a crucial biological event. In both cases, we discuss how quantitative reasoning can be carried on the specifications and which measures can be obtained from them. Related work and some conclusions follow.

2 Specifying metabolic pathways

We anticipated in the Introduction that our knowledge on the genes within living organisms is dramatically increasing during the last years, while the dynamics of genes and proteins inside the cellular molecular machinery is still largely unknown [24]. Also, nowadays there are no experimental techniques able to track the dynamics of the complete metabolome of a cell. Our approach to this major problem of contemporary biology is representing all the known re-

[1] This happens with relatively small examples we worked out, mainly by hand, within our own stochastic version of BioAmbients.

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE

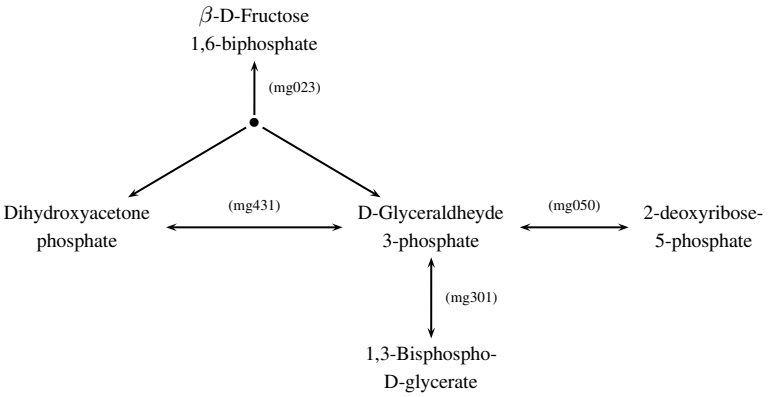


Figure 1. A fragment of the Glycolysis Pathway. The enzyme codes are from [15].

lationships between the elements in a metabolome *in silico*, so building up a sort of a *virtual cell* [28, 27].

Indeed two of us modelled a *whole* living organism, called VICE, which seems to behave *in silico* as a simplified "real" prokaryote [6]. Surprisingly, the experimental results obtained are in accordance with those obtained by experiments *in vivo*.

We worked together with a team of biologists, who carefully designed this hypotetic organism, and kept it as simple as possible. Indeed, even elementary biological entities, like bacteria, have a complexity extremely high, so making their simulation huge, or even computationally unfeasible. The starting point has been [31] that proposes a basic prokaryote genome, by eliminating duplicated genes and other redundancies from the smallest known bacterial genomes. Our co-workers further refined this proposal and obtained a very reduced prokaryote-like genome, which only contains 180 different genes.

Then the specification phase started, with an extended version of the π-calculus. We described the bio-chemical reactants involved in the most basic metabolic pathways of this hypotetic organism: in other words, we did not fully specify a pathway, but only how the species involved in it change their state by effect of a reaction. At a first stage, we focussed our attention merely on the behavioural aspects of the reactants involved, disregarding the quantitative aspects. Yet, running our simplified, theoretical, qualitative description the molecular machinery helped biologists to tune their choice of genes, showing that two were missing. We then used the enhanced aspects of our calculus to determine reaction rates, so to link each transition to a measurable biological parameter.

This phase eventually ended up with a hypotetical genome of VICE, rich enough for expressing a sufficiently large sub-set of the metabolic pathways that every living cell has. This genome got a formal specification, executable *in silico*. We could then start experimenting on a *whole* prokaryote, admittedly extremely simple and placed in an optimal environment. This makes the difference with other specifications that consider parts, rather than a whole "living" entity.

Even though VICE is equipped with the means for cell reproduction, we did not specify these activities. This is because our primary goal was to test whether VICE in interphase was able to "survive" in a "normal" environment, with enough water and essential nutrients, shaped so to dilute or remove all the potentially toxic catabolites, and to ban completely competition and other stressing factors.

We now intuitively show how the bio-chemical aspects of a living cell can be specified with the π-calculus. Actually, we used an enhanced version of this calculus [11, 8], in particular to express stochastic information of process behaviour, along the lines of [32]. We assume the reader has some basic knowledge on process calculi, and we briefly sketch below the fundamentals; a more detailed survey can be found in [11], especially on the enhanced operational semantics approach.

Molecules are represented as *concurrent processes* $P_1, \ldots, P_n$ put in parallel, and written as $P_1 \mid \ldots \mid P_n$. Each process can perform input/output actions on given *channels*. The basic mechanism of *communication* allows processes to exchange information, flowing from the *sender* to the *receiver*, along a channel the two partners share. The input/output actions can be sequentially composed with the "." operator. Finally, a process $P_1 + P_2$ can evolve either as P_1 or as P_2, and the choice will be driven according to a probabilistic distribution. Typically, the input/output actions will be associated with a given rate, that contributes to

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE

the definition of the rate at which a whole system of concurrent processes, i.e. of molecues, changes its state under a reaction. We shall come on this issue later on.

We now consider a fragment of the Glycolysis pathway, which has been extensively studied in the literature; this example has been taken from [6], to which we refer the interested reader for more details. We depict this pathway in Figure 1, where metabolites have their IUPAC names, and the labels on the arrows stand for the enzymes catalysing the reaction; we use here the enzyme code given in [15].

The metabolite β-D-fructose-1-6BP splits giving Dihydroxyacetone phosphate and D-Glyceraldehyde 3-phosphate, in presence of mg023, namely the enzyme fructose-bisphosphate aldolase. In turn, Dihydroxyacetone phosphate can produce D-Glyceraldehyde 3-phosphate, when catalysed by the enzyme mg431, namely triose phosphate isomerase. Additionally, the metabolite D-Glyceraldehyde 3-phosphate can behave in two further, mutually exclusive manners, becoming either the metabolite 1,3-Bisphospho-D-glycerate via mg301 (glyceraldehyde 3-phosphate dehydrogenase), or 2-deoxyribose 5-phosphate via mg050 (2-deoxyribose-5-phosphate aldolase). All the reactions considered above are reversible, e.g. the *aldol condensation* produces the metabolite β-D-Fructose 1,6-bisphospathe from Dihydroxyacetone phosphate and D-Glyceraldehyde 3-phosphate, exploiting mg023.

We specify the behaviour of each reactant in isolation as a process, and interpret bio-chemical reactions by process communications. We thus leave to the basic feature of the calculus, i.e. to communication the task of coordinating the overall behaviour of a net of molecules, following the paradigm of "cells as computations" put forward by Regev and Shapiro [40]. Unlike the original presentation in [41], we here use channels to model catalysing enzymes. As we will see in a while, this choice helps us in collecting statistics on the trend of the considered pathways.

The specification of the metabolites D-Glyceraldehyde 3-phosphate and Dihydroxyacetone_phosphate follows (as usual, we omit the inactive process **0** in trailing position).

D-Glyceraldehyde_3-phosphate =
τ_{mg301}.1,3-Bisphospho-D-glycerate1 +
τ_{mg431}.Dihydroxyacetone_phosphate +
τ_{mg050}.2-Deoxyribose-5-phosphate +
mg023(x).β-D-fructose-1-6bP

Dihydroxyacetone_phosphate =
τ_{mg431}.D-Glyceraldehyde_3-phosphate + $\overline{\text{mg023}}\langle a\rangle$

As said above, the two reactants undergo an aldol condensation in presence of the enzyme mg023. In our specification, one process offers the output $\overline{\text{mg023}}\langle a\rangle$ on the channel associated with the enzyme fructose-bisphosphate aldolase, i.e. mg023; the other process is ready to input on the same channel, through the action mg023(x). When in parallel, the two constitute a process P, that perform a synchronization and give rise to the transition of Figure 2.

The reader has certainly noticed that the label of the arrow carries more information than usual. This is the point where the enhanced semantics of [11] plays a role. Indeed, the label records *where* the communication took place, i.e. the enzyme involved, which were the *partners* involved and which their *role*: the one at the left of the operator |, identified by tag $||_0$, sent a message, the one at the right received it, at position $||_1$. Clearly, in the general case we will have a sequence of tags $||_0, ||_1$ prefixing inputs and outputs, and this sequence is mechanically generated by the rules governing the enhanced semantics of the π-calculus. Below, we shall sketch how this information enables us to derive the rates of transitions.

In the specification above there is also a non standard usage of the internal move τ, that here carries the additional information about which enzyme, e.g. mg050, catalysed the reaction that permitted D-Glyceraldehyde_3-phosphate to become 2-Deoxyribose-5-phosphate in isolation.

The other metabolites depicted in Figure 1 are specified as follows.

β-D-Fructose-1-6_bP =
τ_{mg215}.β-D-Fructose_6P +
τ_{mg023}. (Dihydroxyacetone_phosphate |
D-Glyceraldehyde_3-phosphate)

1,3-Bisphospho-D-glycerate =
τ_{mg301}.D-Glyceraldehyde_3-phosphate +
τ_{mg300}.3-Phospho-D-glycerate

2-Phospho-D-glycerate =
τ_{mg430}.3-Phospho-D-glycerate +
τ_{mg407}.Phosphoenolpyruvate

Once the whole set of reactants has been specified, we can start our *virtual* experiment. The initial state will contain the selected quantity of reactants, put in parallel with each other and with the wanted number of nutrients to represent the living environment. Then the experiment begins, and consists of a *computation*, i.e. a sequence of transitions, starting from the initial state. A comment on the rates of the transitions is now in order.

Recall that all the transitions outgoing from a given process can be mechanically deduced using the logical rules specifying the enhanced semantics, i.e. by mechanically proving the existence of the transitions. The application of the rules, in particular those involving the parallel combinator, gives rise to the labels of the transitions. We exploit the information contained in these labels to derive the rates of

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE

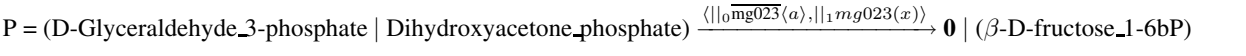


Figure 2. A syncronization modelling aldol condensation.

each transition. Roughly, a label contains either the channel on which a communication occurred, i.e. the enzyme catalysing the corresponding reaction, or a silent action, indexed by an enzyme, besides strings of tags $||_0, ||_1$. This is a natural choice, because our goal was to test whether VICE could "survive", and therefore we studied the overall flux of its pathways, through the so-called *control strength* of the enzymes involved in reactions. Intuitively, it represents the impact of enzyme activity: the greater the control strength, the more perturbated is the flux when the enzyme in inhibited [12, 13].

To do that, we first assigned "basic" reaction rates, called *basal rates* to "basic transitions", i.e. to firing pairs of matching output/input, or firing an internal action. These basal rates depend on the chemical and physical properties of the molecules involved in a specific reaction. To tune them, we used here the constants K_M of the "Michaelis-Menten" kinetics [14, 20]. To model the control strength, we had to take care also of the difference between *near-equilibrium* and *non-equilibrium* reactions. For the first, the rates of a reaction and of its reverse are close, while differ greatly for the second kind of reactions.

As a second step in assigning rates to the transitions of the whole system we took into account the concentration of the other reactants in the virtual solution and of the nutrients therein. A very rough estimate of this concentration is computed by looking at the strings of $||_0$ and $||_1$ present in a transition label. E.g. consider $||_1||_0\langle||_0\overline{\text{mg023}}\langle a\rangle, ||_1 mg023(x)\rangle$, that labels a transition corresponding to an aldol condensation (see Figure 2, where only the basic transition was displayed). From the label, we can deduce that the process P performed the transitions, and that it was in parallel with two unknown processes X (at position $||_0$) and Y (at position $||_1||_1$) in the following way: $X \mid (P \mid Y)$. Indeed, the strings over $||_0, ||_1$ represent the syntactic context in which the basic transition occurs. Following [32, 11], we compute the actual rate of a transition of the whole system by applying a suitable sequence of operations on its basal rate. We inductively traverse the syntactic context: the sequence of the tags and the enzyme involved in the transition determine the operations to perform. As a matter of fact, this is a very rough implementation of the Gillespie algorithm [16], that proposes an exact numerical calculation to stochastically simulate the time evolution of a chemical system.

Eventually, the transitions that leave a given state are labelled by their rate and by the enzyme catalysing the corresponding reaction. One of them is chosen, accordingly to the exponential distribution[2] defined so far, and applied to perform the next computation step, leading to a new state.

Now we can see our computations as virtual experiments, and start collecting statistical data, in our case on the usage of enzymes. First of all we checked our virtual cell was using all the pathways chosen. This showed that VICE endowes sufficient components in its genome. We also made sure that all these components were involved in some simulation runs, and so all of them are necessary.

We carried on these tests varying the concentration of metabolites, in particular sugars. We also experimented on different time intervals of the observation, simply represented by the number of transitions in computations.

Our main experiment compared some aspects of the behaviour of VICE with that of real prokaryotes acting *in vivo* under similar circumstances [20]. In particular, we investigated the glycolysis, on which the literature has a huge quantity of biological data. A first analysis showed that the distribution of the metabolites along this pathway of VICE significantly matches with those of real organisms reported in the literature. To our surprise, the diagrams showing the trend of VICE glycolysis almost overlap the real one; see Figure 3, taken from [6].

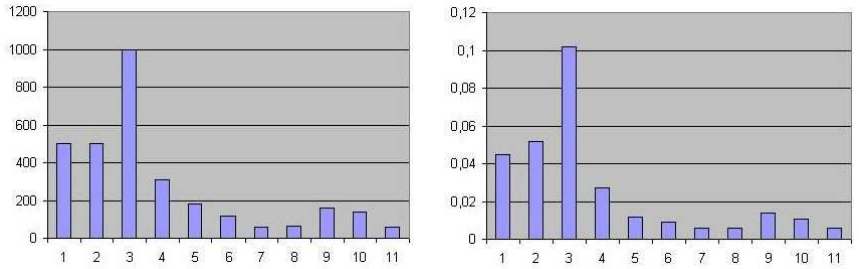
3 Membranes and enzime inhibition

In the previous section we showed how (an enhanced version of) the π-calculus can be used to formally describe metabolic pathways, and how quantitative reasoning can be carried out over the specifications so obtained. Running a simple example, we now present some recent developments in the application of process calculi to biological modelling.

The process calculi proposed for representing the dynamics of biological systems can roughly be classified

[2] An exponential distribution with rate r is a function $F(t) = 1-e^{-rt}$, where t is the time parameter. The value of $F(t)$ is smaller than 1 and $\lim_{t\to\infty} F(t) = 1$. The parameter r determines the slope of the curve F: the greater r, the faster $F(t)$ approaches 1. The probability of performing an action with parameter r within time x is $F(x) = 1 - e^{-rx}$, so r determines the time, Δt, needed to have a probability near to 1.

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE



(a) (b)

1	mg111	5	mg300	9	pyruvate dehydrogenase complex
2	mg215	6	mg430	10	mg299
3	mg023	7	mg407	11	mg357
4	mg031	8	mg216		

(c)

Figure 3. The trend of VICE glycolysis (a) towards the real one (b). Table (c) shows the correspondence between indexes on x-axis and enzymes. Their codes are in [15].

depending on their interpretation of a crucial biological event: *complex formation*. In the π-calculus, where names can be communicated in interactions, complex formation corresponds to the act of sending a private name, and the generated *complex* is just the parallel composition of the processes sharing the communicated name. Other calculi (e.g. BioAmbients [39], Brane Calculi [4], Beta-binders [35]) provide explicit means to model enclosing surfaces of entities, with a natural visual representation of complexes.

In Beta-binders processes are encapsulated into *boxes*, that mimic biological membranes, and have interaction sites, that resemble biological *motifs* where, e.g. biological molecules can bind together. This allows for a simple representation of biochemical reactions: in order for two proteins to interact, they have to undergo into physical contact through their motifs. In Beta-binders, the corresponding boxes establish a channel where the two proteins can exchange information. The *backbone*, i.e. the biological structure of complexes, is wrapped inside a box. It is specified as a π-calculus process, extended with a few additional primitives to manipulate interaction sites, i.e. for exposing a new site, or making a pre-existent site unavailable, or making a hidden site available again. Beta-binders was introduced building on the intuition that the communication capabilities of processes should be made more flexible than in the existing calculi. Indeed, a complex formation is specified in these calculi through a user-driven coordination of the synchronizations between the corresponding processes. This mechanism appears to be too rigid for either abstracting from unknown details or for predicting evolutionary behaviours. For this reason in Beta-binders sites are actually typed, and interactions between boxes depend on site *types*, unlike in the π-calculus, where names are used instead. In particular, interaction between boxes depends on the *affinity* of their motifs, which is denoted as a function α of the relevant site types. We do not detail function α, because it depends on the particular applicative scenario, but for example a similar notion is used in the field of drug discovery [25]. Also, the semantics rules that drive the behaviour of Beta-binders boxes comprise rules for either *joining* or *splitting* boxes.

Below, we comment on the representation of competitive inhibition [1] in a quantitative extension of Beta-binders. A competitive inhibitor is a molecule that occupies a catalytic site because of its similarity to the substrate. When occupying the site, the inhibitor prevents the normal substrate from binding and being catalysed. Operationally, competitive inhibitors bind reversibly to the active site. The following Michaelis-Menten equation states the generalised scheme for enzyme-catalysed production of a product P from the substrate S:

$$E + S \rightleftharpoons^{K_{ES}}_{K_{ES}^{-1}} ES \rightharpoonup^{K_P} E + P \qquad (1)$$

This equation [3] comprises four reactions. The first shows that the enzyme E can bind to S with constant rate K_{ES} and form the complex ES:

$$E + S \rightharpoonup^{K_{ES}} ES. \qquad (2)$$

[3]The symbol + in chemistry denotes the simultaneous presence of the molecules in the summation, rather than non-deterministic choice between teo processes, as done in the previous section.

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE

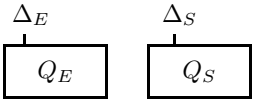
The reaction can then be reversed with constant rate K_{ES}^{-1}:

$$E + S \leftharpoondown_{K_{ES}^{-1}} ES. \tag{3}$$

Alternatively, the enzyme E can catalyse the reaction transforming the substrate S into the product P with constant rate K_P:

$$ES \rightharpoonup^{K_P} E + P. \tag{4}$$

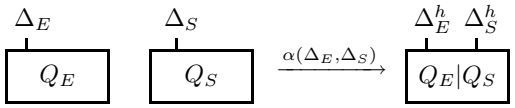
We specify each reactant as a distinct box that executes in parallel with the others. For example, the chemical scenario $E + S$ in (2) is graphically rendered as follows, where the leftmost box corresponds to E, and the other one to S: within the boxes the π-calculus processes Q_E and Q_S model the internal structure of E and S, namely their backbones.



The reaction (2) is modeled as the joining of the boxes for the enzyme E and the substrate S, with associated rate $\alpha(\Delta_E, \Delta_S) = K_{ES}$. The Beta-binders semantics of joining is such that the system in the above picture can perform a computation step and transform into a system composed by a single box.

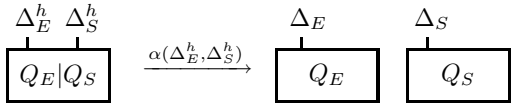
As in the previous section, computation steps, i.e. reactions, are denoted by $B \xrightarrow{r} B'$, and the real number r records the rate at which the transition occurs to be used in resolving race conditions. For computing the execution probability r for each possible molecular interaction, we use the Gillespie's algorithm [37, 33]. This probability depends on the basal rate r' and the number of the motifs which could have generated the same molecular collision. In our example, let the number of boxes for E be N_1, and those for S be N_2; then the overall rate r will be $r' \times N_1 \times N_2$. The rules of the operational semantics of Beta-binders collect the relevant information while deducing the transitions for the current configuration of the system. Differently than in the previous section, we give a very specific format to the labels of transitions and some non trivial computations may be needed to determine inductively the rates while, e.g. putting two complexes in parallel.

Graphically we represent the transition modelling the reaction (2) as follows:

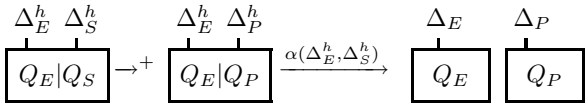


where the rightmost box corresponds to the complex ES in the Michaelis-Menten equation. Therein, the two backbones Q_E and Q_S are put in touch (composed in parallel), and the motifs Δ_E and Δ_S are hidden (written Δ_E^h and Δ_S^h, respectively). The value $\alpha(\Delta_E,\Delta_S)$ labelling the transition $\xrightarrow{\alpha(\Delta_E,\Delta_S)}$ is the specific basal rate of the enzyme-substrate reaction.

To model the reversed reaction as given in (3), the box that represents the complex ES is split with dissociation rate $\alpha(\Delta_E^h, \Delta_S^h) = K_{ES}^{-1}$.



Finally, consider the reaction in (4). This equation approximates the real biological phenomenon, that consists of a sequence of internal modifications. These change S into P so allowing the complex to split. Although we did not specify the backbones Q_E and Q_S in full detail, it was mentioned that they are essentially π-calculus processes possibly made up of several parallel sub-processes and able to interact the one with the other. In this respect, the sequence of biological internal modifications that induce the transformation of S into P, are rendered as a sequence of interactions among the parallel sub-components of $Q_E \mid Q_S$. Summarizing, the specification of the reaction (4) is:



where $\rightarrow^+$ stays for one or more computation steps. Note that the steps of the backbone are not labelled. This is because the Michaelis-Menten equation only provides the overall reaction constant rate K_P, abstracting from the phiscal phenomena that transform S in P. Indeed, the complex ES goes through a structural rearrangement (e.g. oxidation) becoming EP and then the complex is broken producing the product P and freeing the enzyme E. Beta-binders carefully models structure rearrangement through one or more interactions between the internal processes Q_E and Q_S [36]. Unfortunately, classic Michaelis-Menten schema does not provide us with information about the rates of the reactions that modify the internal structure of complexes like ES. Rather than looking inside the specific reaction and deriving the needed rates, we followed the classical approach to stochastic simulation and we abstracted away from internal modifications.

We now extend the above specification, taking advantage of the compositional nature of process calculi. Consider the following extension of the Michaelis-Menten equation (1):

$$IE + S \leftrightharpoons^{K_{EI}}_{K_{EI}^{-1}} I + E + S \leftrightharpoons^{K_{ES}}_{K_{ES}^{-1}} I + ES \rightharpoonup^{K_P} I + E + P \tag{5}$$

This last equation uses an inhibitor I that can bind to the enzyme E so preventing the interaction between enzyme and substrate:

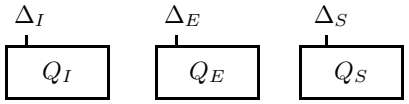
$$IE + S \leftharpoondown_{K_{EI}} I + E + S. \tag{6}$$

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE

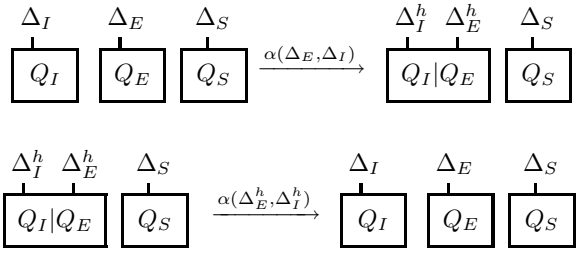
This reaction can be reversed with constant rate K_{EI}^{-1}:

$$IE + S \rightharpoonup^{K_{EI}^{-1}} I + E + S \tag{7}$$

The Beta-binders corresponding to (5) is obtained by just adding the enzyme-substrate model with yet another box playing as the inhibitor I.



Equation (6) is rendered by a joining with rate $\alpha(\Delta_E, \Delta_I) = K_{EI}$, while the corresponding of equation (7) is driven by a splitting rule with rate $\alpha(\Delta_E^h, \Delta_I^h) = K_{EI}^{-1}$:



Of course, the Beta-binders specification of (5) requires no modifications in the representation of the enzyme-substrate model for (1). Indeed, the inhibitor plays no role in the reactions already specified and corresponding to equation (1).

The competitive inhibition specified above is quite interesting from the quantitative point of view. Suppose that $K_{EI} \simeq K_{ES}$ and that $K_{EI}^{-1} << K_{ES}$. Then the formation of complexes EI and ES have similar probabilities, while the dissociation speed K_{EI}^{-1} of EI is very low. So the total reaction speed slows down. In this case the concentration of molecules involved in the reaction plays a crucial role in understanding the global phenomenon.

In order to investigate this kind of situations, a simulator is currently under development at Trento University. The simulator resolves race conditions, as the one between the enzyme-substrate reaction and the enzyme-inhibitor reaction, using Gillespie First Reaction methods [16]. Roughly speaking the simulator takes the current configuration B of the Beta-binders specification and iterates the following three steps. The simulator

1. computes the set $\{B \xrightarrow{r_1} B_1, \ldots, B \xrightarrow{r_n} B_n\}$ of all the transitions outgoing from B;
2. chooses the next configuration B_i following the stochastic simulation algorithm of [16]; actually, the transition that is allowed to happen is the one with the highest execution probability r_i;
3. updates B with B_i; note that molecular concentrations are modified so that new stochastic rates have to be determined again in the next computation step, and will be different from the current ones.

We have some initial simulations on toy examples, as the one we have presented, that follow the predictions of the chemical master equation. This make us confident that Beta-binders models are consistent with real biological systems under analysis.

Note that the implementation strictly mimicks the semantics of the calculus represented by the transition relation $\xrightarrow{r}$, as the operational semantics specifies the machine implementation, thus making it easy to develop prototypes. This is a pleasent property while computer scientist are about implementing requirements from the biologists, who often add new or different functionalities to their models.

4 Concluding remarks

The recent progress of molecular biology has made it possible the detailed description of the components that constitute living systems, notably genes and proteins, as isolated entities. Biological molecules do not live alone. Rather, they participate in very complex networks that are involved in the maintenance as well as the differentiation of cellular systems (consider, e.g. regulatory networks for gene expression). The new challenge is then scaling up from molecular biology to *systems biology*, and understanding how these individual components integrate to take part into complex systems, and how they function, evolve, and interact together.

We claim that computer science will play a central role in these fashinating investigations. Molecular biologists already use information technology to process, analyse, compare and share scientific knowledge. It is "easy" for a scientist to study a chosen sequence, archiving, editing it, comparing it with other sequences, and analyzing its various properties. The use of widely distributed databases is now common in the scientific community. The next step is to extend this approach for studying metabolic networks, signalling pathways, regulatory circuits or even a complete, entire integrated cell. The goal is to combine experimental data with advanced formal theories from computer science to design formal languages for the specification of interacting molecular entities.

The added value for systems biology in joining the process algebra approach is given by the abstraction mechanisms that computer scientists have been developed for concurrent systems over the last 30 years [40]. A living entity can therefore be described at different levels of detail, in an incremental fashion, by refining a specification till the very basic biochemical description.

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE

Besides abstraction, the main feature to be exploited in systems biology is *compositionality*. It allows to fix the building blocks of systems and to enlarge models by composition without changing the description of the subsystems already available. As exemplified in the paper, the underlying idea is to see biomolecular systems as a set of elementary components from which complex entities are constructed. New features can then be freely added, as shown at the end of the previous section. Several calculi are based on these considerations. Among them, the biochemical stochastic π-calculus[41], BioAmbients [39], CCS-R [9] and the κ-calculus [7], to cite only a few.

Another crucial extension concerns a manageable representation and treatment of quantitative information. This is of paramount importance, because not only understanding complex biological systems is simply impossible without measuring the parameters that affect their behaviour, but also simulating simple living entities *in silico* gives very little information when quantitative measures are not sufficiently accurate. Indeed, a formal quantitative foundation of biology requires deep extensions of the available process algebras to take care of such aspects like real time, temperature, pressure, threshold, etc. A first, little step towards these more structured extensions is offered by a few stochastic process algebras, among which the biochemical stochastic π-calculus, or variants of BioAmbients and Beta-binders, that essentially implement a stochastic simulation algorithm, based on the Gillespie's one. These extensions allow for transient analysis, but also the more standard stationary analysis based on markovian processes requires more powerful tools than the available ones to be efficiently carried on.

As a concluding remark, we note that there is an emergent line of research, aiming at defining calculi directly inspired by biology. BioAmbients [39], Brane calculi [4] and Beta-binders themselves follow this guideline, and appear to be better suited to modeling, analysing and simulating living systems.

Whether the bio-mimetic approach can further inspire and enhance our comprehension of how computer artificial systems can be modeled, designed and implemented is a long term, visionary challenge.

Acknowledgements. We are deeply indebted with Davide Chiarugi and Roberto Marangoni. Without their patience and enthusiasm, VICE would never start eating virtual sugars! We acknowledge partial support from the PRIN project *Sybilla*.

References

[1] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Molecular biology of the cell (IV ed.)*. Garland science, 2002.

[2] M. Bernardo, L. Donatiello, and R. Gorrieri. A formal approach to the integration of performance aspects in the modelling and analysis of concurrent systems. *Information and Computation*, 144:83–154, 1998.

[3] P. Buchholz. On a Markovian process algebra. Technical Report 500, Informatik IV, University of Dortmund, 1994.

[4] L. Cardelli. Brane Calculi. In *CMSB '04*, volume 3082 of *LNBI*, pages 257–280. Springer, 2004.

[5] L. Cardelli and A. Gordon. Mobile ambients. In *Proceedings of FoSSaCS'98*, LNCS 1378, pages 140–155. Springer-Verlag, 1998.

[6] D. Chiargugi, M. Curti, P. Degano, and R. Marangoni. VICE: A VIrtual CEll. In V. Danos and V. Schachter, editors, *Procs. of the* 2^{st} *Int. W/S on Computational Methods in Systems Biology*, volume 3082 of *LNBI*. Springer, 2005.

[7] M. Chiaverini and V. Danos. A core modeling language for the working molecular biologist. In C. Priami, editor, *CMSB '03*, volume 2602 of *LNCS*. Springer, 2003.

[8] M. Curti, P. Degano, C. Priami, and C. Baldari. Modelling biochemical pathways through enhanced π-calculus. *Theoretical Computer Science*, 325(1):111–140, 2004.

[9] V. Danos and J. Krivine. Formal molecular biology done in CCS-R. In *BioConcur '03, Workshop on Concurrent Models in Molecular Biology*, 2003.

[10] P. D'Argenio, J.-P. Katoen, and E. Brinksma. An algebraic approach to the specification of stochastic systems. In *Proceedings of PROCOMET'98, IFIP Series*, pages 126–147. Chapman-Hall, 1998.

[11] P. Degano and C. Priami. Enhanced operational semantics. *ACM Computing Surveys*, 28(2):352–354, 1996.

[12] T. Devlin and *al.* *Textbook of Biochemistry, fifth edition*. Wiley and Sons, Inc., 2002.

[13] D. Fell. *Understanding the control of metabolism*. Portland Press, London, United Kingdom, 1997.

[14] A. Fersht. *Structure and Mechanism in Protein Science: A Guide to Enzyme Catalysis and Protein Folding*. Freeman, 1999.

[15] C. Fraser and *al.* The minimal gene complement of mycoplasma genitalium. *Science*, 270(1):397–403, 1995.

[16] D. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. of Computational Physics*, 22, 1976.

[17] D. Gillespie. Exact stochastic simulation of coupled chemical reactions. *Journal of Physical Chemistry*, 81(25):2340–2361, 1977.

[18] D. Gillespie. A rigorous derivation of the chemical master equation. *Physica A*, 188:404, 1992.

[19] N. Götz, U. Herzog, and M. Rettelbach. TIPP- a language for timed processes and performance evaluation. Technical Report 4/92, IMMD VII, University of Erlangen-Nurnberg, 1992.

[20] G. Hammes and P. Shimmel. *The Enzymes, vol. 2*. P.D. Boyer (New York Academic Press), 1970.

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE

[21] P. Harrison and S. B. Stochastic process algebra for discrete event simulation. In *Quantitative Methods in Parallel Systems*, pages 18–37, 1995.

[22] J. Hillston. *A Compositional Approach to Performance Modelling*. Cambridge University Press, 1996.

[23] C. Hoare. *Communicating Sequential Processes*. Prentice-Hall, 1985.

[24] H. Kitano. *Foundations of System Biology*. MIT Press, 2002.

[25] I. Kuntz, J. Blaney, J. Oatley, R. Langridge, and T. Ferrin. A geometric approach to macromolecule-ligand interactions. *J. of Molecular Biology*, 161, 1982.

[26] P. Lecca, C. Priami, P. Quaglia, B. Rossi, C. Laudanna, and G. Costantin. Language modeling and simulation of autoreactive lymphocytes recruitment in inflamed brain vessels. 2003. To appear in SIMULATION: Transactions of The Society for Modeling and Simulation.

[27] L. Loew and J. Schaff. The virtual cell: a software environment for computational cell biology. *Trends Biotechnology*, 19(10):401–406, Oct. 2001.

[28] T. Masaru and *al.* E–CELL: software environment for whole–cell simulation. *Bioinformatics*, 15:72–84, 1998.

[29] R. Milner. *Communication and Concurrency*. Prentice-Hall, London, 1989.

[30] R. Milner, J. Parrow, and D. Walker. A calculus of mobile processes (i and ii). *Information and Computation*, 100(1):1–72, 1992.

[31] A. Mushegian and E. Koonin. A minimal gene set for cellular life derived by comparison of complete bacterial genomes. *Proceedings of National Academy of Science USA*, 93:10268–10273, 1996.

[32] C. Nottegar, C. Priami, and P. Degano. Performance evaluation of mobile processes via abstract machines. *IEEE Transactions on Software Engineering*, 27, 10:867–889, 2001.

[33] A. Phillips and L. Cardelli. A correct abstract machine for the stochastic pi-calculus. In *BioConcur '04*, ENTCS. Elsevier.

[34] C. Priami. Stochastic π-calculus. *The Computer Journal*, 38(6):578–589, 1995.

[35] C. Priami and P. Quaglia. Beta Binders for Biological Interactions. In *CMSB '04*, volume 3082 of *LNBI*, pages 21–34. Springer, 2004.

[36] C. Priami and P. Quaglia. Operational patterns in beta-binders. In *Transactions on Computational Systems Biology*, volume 3380 of *LNCS*. Springer, 2005.

[37] C. Priami, A. Regev, W. Silverman, and E. Shapiro. Application of a stochastic passing-name calculus to representation and simulation of molecular processes. *Information Processing Letters*, 80:25–31, 2001.

[38] A. Regev. Representation and simulation of molecular pathways in the stochastic π-calculus. In *Proceedings of the 2nd workshop on Computation of Biochemical Pathways and Genetic Networks*, 2001.

[39] A. Regev, E. Panina, W. Silverman, L. Cardelli, and E. Shapiro. BioAmbients: An Abstraction for Biological Compartments. *Theoretical Computer Science*, 325(1), 2004.

[40] A. Regev and E. Shapiro. Cells as computations. *Nature*, 419:343, 2002.

[41] A. Regev, W. Silverman, and E. Shapiro. Representation and simulation of biochemical processes using the π-calculus process algebra. In *Pacific Symposium of Biocomputing (PSB2001)*, pages 459–470, 2001.

Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05)
0-7695-2427-3/05 $20.00 © 2005 IEEE