THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

Agile Modelling of Cellular Signalling (Invited Paper)

OPEN ACCESS

Agile Modelling of Cellular Signalling (Invited Paper)

Vincent Danos[2,1]

School of Informatics
University of Edinburgh
Edinburgh, UK

Abstract

We illustrate with a simple example how using a rule-based approach to the modelling of protein interaction networks allows for quickly putting together models (ease of expression), and quickly modifying them (ease of variation).

Keywords: Modification, covalent modification, cascade, Huang-Ferell model

1 Introduction

The purpose of this note is to demonstrate the flexibility of a structured and rule-based approach to modelling cellular signalling. Although the approach is especially useful when the biological system of interest is highly combinatorial, we will use an example that the intuition can grasp comfortably.

Specifically we shall:
- start with a basic motive of reversible covalent modification [8];
- obtain a rule-based version of the Huang-Ferell model of a cascade [10];
- and rewire our cascade to illustrate how easy it is to do so.

Simple as this cascade is, it already generates about six hundred different species, and would be tedious to specify by traditional reaction-based methods, and even more so to modify. In fact one of the major advantages of the approach we advocate is to avoid the combinatorial explosion of traditional approaches. A more complex example which illustrates this point in spectacular fashion can be found in Ref. [3].

There one also considers the causal analysis of models, a topic which the present note will not discuss.

Other approaches using process-algebraic languages [12,1,11,5,2] could do well in principle but for the fact that they require the enumeration of all possible internal states of a process. Those based on a binary model of communication need to encode more in the processes internal states, which makes the problem more acute. Rule-based languages such as Kappa [6] -which we use here, or the BioNetGen language [9] describe biological interactions in a more natural and intuitive way and as such are also at a clear practical advantage.
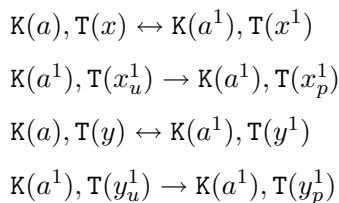
2 Reversible modifications

The description of a system in Kappa consists of a collection of agents and rules. An agent has a set of sites, which may have internal states, typically used to denote phosphorylation and other post-translational modifications, and may also bind to other sites, to represent the formation of complexes. Rules describe conditions under which agents associate, dissociate or get modified. Agent deletion or creation is also possible. This idealised repertoire allows to model a substantial part of the interactions involved in cellular signalling.

In our first example we need only three types of agents: a kinase K, a target T with two phosphorylatable sites x and y, and a phosphatase P.

At the rule level, a phosphorylation event can be decomposed in a triplet of elementary events:
- K binds its target T at site x or y,
- K may (but need not) phosphorylate the site to which it is bound,
- K dissociates from T.

This translates in the following four rules (two rules per site in T):

$$K(a), T(x) \leftrightarrow K(a^1), T(x^1)$$
$$K(a^1), T(x_u^1) \rightarrow K(a^1), T(x_p^1)$$
$$K(a), T(y) \leftrightarrow K(a^1), T(y^1)$$
$$K(a^1), T(y_u^1) \rightarrow K(a^1), T(y_p^1)$$

In the textual notation we are using here, internal states are shown as subscripts u (unphosphorylated) and p (phosphorylated) to the sites they are attached to; bonds are represented as shared superscripts across agents to indicate the two endpoints of a link. A double arrow indicates a reversible rule. The left hand side of a rule specifies a condition to trigger the rule, while the right hand side specifies changes to agents mentioned on the left. Thus the second reaction says that when the kinase is bound to its target at x, this site may get phosphorylated.

Note that the sites of an agent need not all be present in a rule, eg in the first rule above, T does not mention y. Likewise, if a site is mentioned at all, its internal state may be left unspecified, eg in the same first rule one does not say whether x in

T is phosphorylated or not. This is the 'don't care, don't write' convention: only the information which is conditioning the application of a rule needs to be represented.

The action of the phosphatase P, which undoes the action of K, is described by a set of entirely similar rules:

$$\text{P}(a), \text{T}(x) \leftrightarrow \text{P}(a^1), \text{T}(x^1)$$
$$\text{P}(a^1), \text{T}(x_p^1) \rightarrow \text{P}(a^1), \text{T}(x_u^1)$$
$$\text{P}(a), \text{T}(y) \leftrightarrow \text{P}(a^1), \text{T}(y^1)$$
$$\text{P}(a^1), \text{T}(y_p^1) \rightarrow \text{P}(a^1), \text{T}(y_u^1)$$

The set of rules we obtain is a recurrent motif with many variations in cellular signal transduction [8].

Note that Kappa's fine-grained notation forces us to make our mechanistic choices explicit. For instance, we choose to view the phosphorylation of T by K as a 'distributive' mechanism, whereby the kinase must let go of its target in between two modifications, since it cannot remain bound to site x and phosphorylate site y. Other variants of multisite phosphorylation involve a 'processive' mechanism whereby the same kinase may modify several sites of its target without letting go of it. We also suppose that K associates to and dissociates from T irrespective of the internal state of the site it binds to. Again, one could choose differently.

3 A cascade

We shall now arrange this simple motif in a cascade where RAS activates RAF, which activates MEK, which in turn activates ERK; each intermediate has a dedicated phosphatase, in order of appearance PP2A1, PP2A2, and MKP3.

The hierarchical structure of the rule set reflects in its contact map (Fig. 1). This contact map is a graph with sites that is usful summary of a rule set: one has one node per agent type, any two sites that can be connected by a rule are connected, and any modifiable site is indicated as such.

Our rules for the cascade will split naturally in three groups, each very similar to the rule set considered in the previous section.

The first group deals with the activation of RAF by RAS (which presupposes that RAS is active):

$$\text{RAS}(\text{S1S2}), \text{RAF}(x_u) \rightarrow \text{RAS}(\text{S1S2}^1), \text{RAF}(x_u^1)$$
$$\text{RAS}(\text{S1S2}^1), \text{RAF}(x_u^1) \rightarrow \text{RAS}(\text{S1S2}^1), \text{RAF}(x_p^1)$$
$$\text{RAS}(\text{S1S2}^1), \text{RAF}(x^1) \rightarrow \text{RAS}(\text{S1S2}), \text{RAF}(x)$$
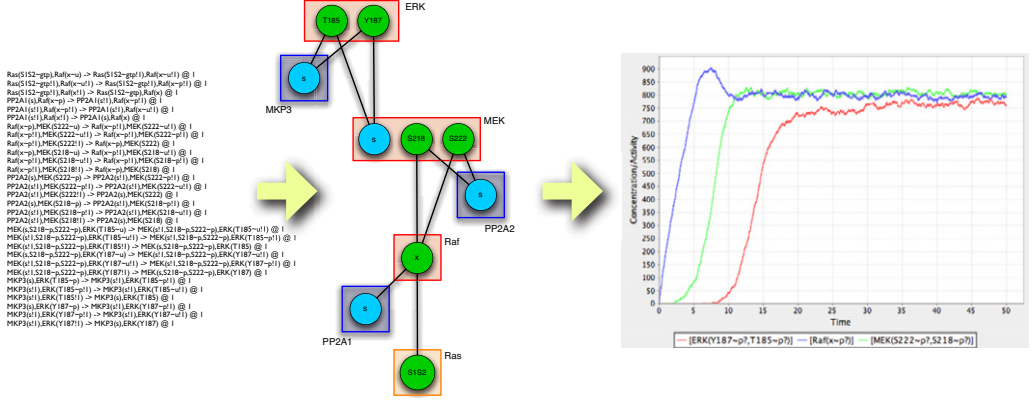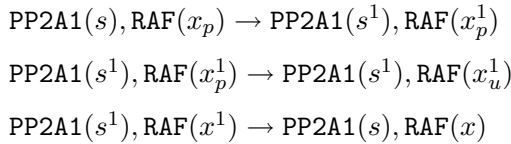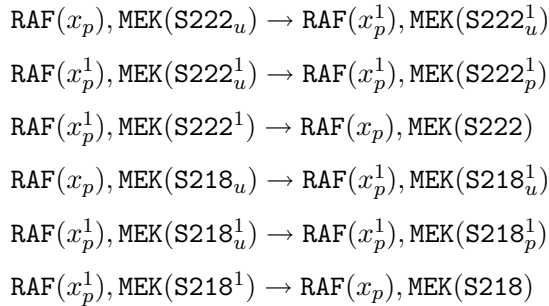
Fig. 1. *The contact map of the canonical* RAS *cascade, together with the time evolution of the level of the fully phosphorylated forms of* RAF, MEK, *and* ERK; *the initial state has 300 active* RAS, *a 1000 of each kinase* RAF, MEK, *and* ERK, *all unphosphorylated, and 200 of each phosphatase* PP2A1, PP2A2, *and* MKP3; *all rates are set to* 1.

and the inactivation of RAF by PP2A1:

$$\mathrm{PP2A1}(s), \mathrm{RAF}(x_p) \rightarrow \mathrm{PP2A1}(s^1), \mathrm{RAF}(x_p^1)$$

$$\mathrm{PP2A1}(s^1), \mathrm{RAF}(x_p^1) \rightarrow \mathrm{PP2A1}(s^1), \mathrm{RAF}(x_u^1)$$

$$\mathrm{PP2A1}(s^1), \mathrm{RAF}(x^1) \rightarrow \mathrm{PP2A1}(s), \mathrm{RAF}(x)$$

The second group deals with the activation of MEK by activated RAF:

$$\mathrm{RAF}(x_p), \mathrm{MEK}(\mathrm{S222}_u) \rightarrow \mathrm{RAF}(x_p^1), \mathrm{MEK}(\mathrm{S222}_u^1)$$

$$\mathrm{RAF}(x_p^1), \mathrm{MEK}(\mathrm{S222}_u^1) \rightarrow \mathrm{RAF}(x_p^1), \mathrm{MEK}(\mathrm{S222}_p^1)$$

$$\mathrm{RAF}(x_p^1), \mathrm{MEK}(\mathrm{S222}^1) \rightarrow \mathrm{RAF}(x_p), \mathrm{MEK}(\mathrm{S222})$$

$$\mathrm{RAF}(x_p), \mathrm{MEK}(\mathrm{S218}_u) \rightarrow \mathrm{RAF}(x_p^1), \mathrm{MEK}(\mathrm{S218}_u^1)$$

$$\mathrm{RAF}(x_p^1), \mathrm{MEK}(\mathrm{S218}_u^1) \rightarrow \mathrm{RAF}(x_p^1), \mathrm{MEK}(\mathrm{S218}_p^1)$$

$$\mathrm{RAF}(x_p^1), \mathrm{MEK}(\mathrm{S218}^1) \rightarrow \mathrm{RAF}(x_p), \mathrm{MEK}(\mathrm{S218})$$

and the inactivation of MEK by PP2A2:

$$\mathrm{PP2A2}(s), \mathrm{MEK}(\mathrm{S222}_p) \rightarrow \mathrm{PP2A2}(s^1), \mathrm{MEK}(\mathrm{S222}_p^1)$$

$$\mathrm{PP2A2}(s^1), \mathrm{MEK}(\mathrm{S222}_p^1) \rightarrow \mathrm{PP2A2}(s^1), \mathrm{MEK}(\mathrm{S222}_u^1)$$

$$\mathrm{PP2A2}(s^1), \mathrm{MEK}(\mathrm{S222}^1) \rightarrow \mathrm{PP2A2}(s), \mathrm{MEK}(\mathrm{S222})$$

$$\mathrm{PP2A2}(s), \mathrm{MEK}(\mathrm{S218}_p) \rightarrow \mathrm{PP2A2}(s^1), \mathrm{MEK}(\mathrm{S218}_p^1)$$

$$\mathrm{PP2A2}(s^1), \mathrm{MEK}(\mathrm{S218}_p^1) \rightarrow \mathrm{PP2A2}(s^1), \mathrm{MEK}(\mathrm{S218}_u^1)$$

$$\mathrm{PP2A2}(s^1), \mathrm{MEK}(\mathrm{S218}^1) \rightarrow \mathrm{PP2A2}(s), \mathrm{MEK}(\mathrm{S218})$$
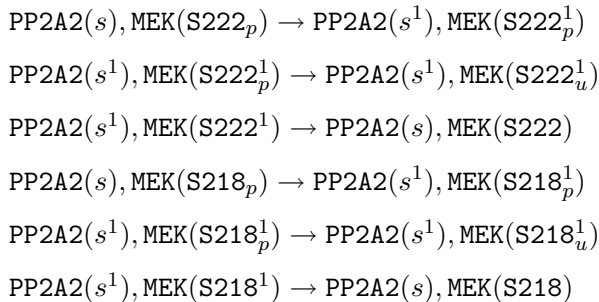
The third group handles the activation of ERK by activated MEK:

$$\text{MEK}(s, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{T185}_u) \rightarrow \text{MEK}(s^1, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{T185}_u^1)$$

$$\text{MEK}(s^1, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{T185}_u^1) \rightarrow \text{MEK}(s^1, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{T185}_p^1)$$

$$\text{MEK}(s^1, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{T185}^1) \rightarrow \text{MEK}(s, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{T185})$$

$$\text{MEK}(s, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{Y187}_u) \rightarrow \text{MEK}(s^1, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{Y187}_u^1)$$

$$\text{MEK}(s^1, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{Y187}_u^1) \rightarrow \text{MEK}(s^1, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{Y187}_p^1)$$

$$\text{MEK}(s^1, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{Y187}^1) \rightarrow \text{MEK}(s, \text{S218}_p, \text{S222}_p), \text{ERK}(\text{Y187})$$
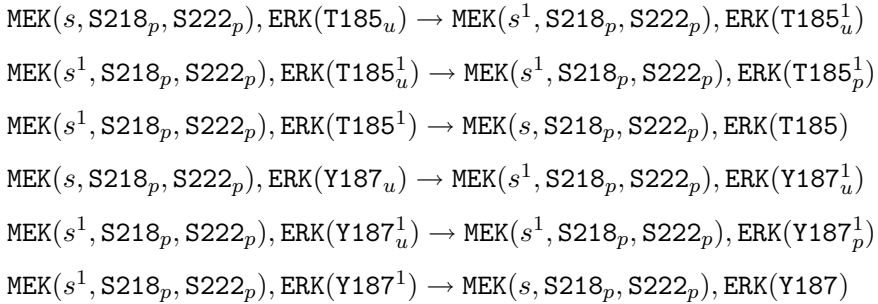
and the inactivation of ERK by MKP3:

$$\text{MKP3}(s), \text{ERK}(\text{T185}_p) \rightarrow \text{MKP3}(s^1), \text{ERK}(\text{T185}_p^1)$$

$$\text{MKP3}(s^1), \text{ERK}(\text{T185}_p^1) \rightarrow \text{MKP3}(s^1), \text{ERK}(\text{T185}_u^1)$$

$$\text{MKP3}(s^1), \text{ERK}(\text{T185}^1) \rightarrow \text{MKP3}(s), \text{ERK}(\text{T185})$$

$$\text{MKP3}(s), \text{ERK}(\text{Y187}_p) \rightarrow \text{MKP3}(s^1), \text{ERK}(\text{Y187}_p^1)$$

$$\text{MKP3}(s^1), \text{ERK}(\text{Y187}_p^1) \rightarrow \text{MKP3}(s^1), \text{ERK}(\text{Y187}_u^1)$$

$$\text{MKP3}(s^1), \text{ERK}(\text{Y187}^1) \rightarrow \text{MKP3}(s), \text{ERK}(\text{Y187})$$
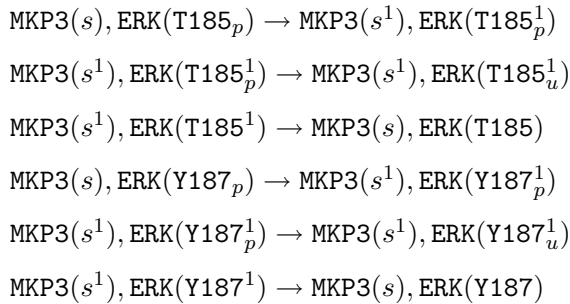
The MEK activation group above has six rules (two bindings, two modifications, and two unbindings) which specify the manner in which RAF activates MEK. Specifically, RAF can bind any of MEK's serine residues S222 and S218 in no particular order (see the contact map in Fig. 1) in the same fashion as our first example.

3.1 Simulation

Given a rate for each rule, a rule set determines a continuous time Markov chain. A state X of the Markov chain is a set of agents with their sites in various internal and binding states. The likelihood that a given rule applies is proportional to its total number of instances in X multiplied by the rule's rate, a quantity known as the rule activity, and the associated time advance is given by an exponential distribution with average frequency the total activity of X (ie the sum of the rules's activities). In the particular case where there is no bindings or internal states, ie in the case of flat chemical reactions (aka multiset rewriting, or Petri nets), the above semantics is known as 'Gillespie's method' [7], and is widely used to simulate the kinetics of coupled elementary chemical reactions. In the case of our particular model, one obtains simulations which show the canonical cascade behaviour where each of the successive tiers rise in turn with a slight delay, and little of the incoming signal can activate (ie doubly phosphorylate) ERK almost entirely at steady state (Fig. 1).

4 Same cascade rewired

Now suppose some data suggests that in fact `RAF` binds to `MEK` at the same site s to which `MEK` also binds. This is expressed in the rewiring of `RAF` to `MEK` in the variant contact map given Fig. 2. To modify the original model and obtain a model of the new situation, all one has to do is replace the six rules describing the interactions of `RAF` and `MEK` (shown above) with the four rules below.
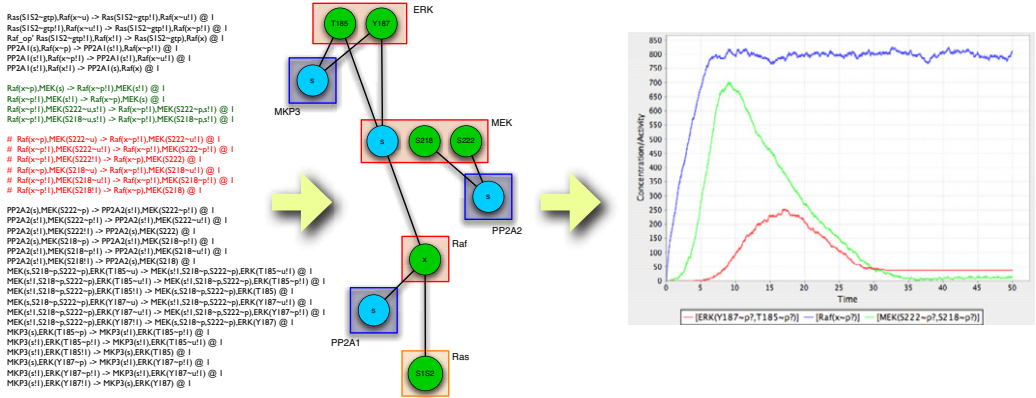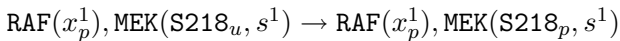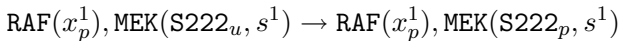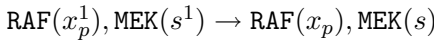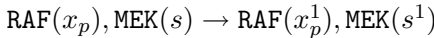

Fig. 2. *A variation on the theme:* `RAF` *competes now with* `ERK` *to bind at site* s *as expressed by the four new rules replacing the six older ones. As a result* `ERK` *is no longer activated.*

$$\text{RAF}(x_p), \text{MEK}(s) \rightarrow \text{RAF}(x_p^1), \text{MEK}(s^1)$$

$$\text{RAF}(x_p^1), \text{MEK}(s^1) \rightarrow \text{RAF}(x_p), \text{MEK}(s)$$

$$\text{RAF}(x_p^1), \text{MEK}(\text{S222}_u, s^1) \rightarrow \text{RAF}(x_p^1), \text{MEK}(\text{S222}_p, s^1)$$

$$\text{RAF}(x_p^1), \text{MEK}(\text{S218}_u, s^1) \rightarrow \text{RAF}(x_p^1), \text{MEK}(\text{S218}_p, s^1)$$

As a result of the modification, the cascade no longer activates `ERK` in a sustained way (Fig. 2). The reason for that behaviour might be that the competition between `RAF` and `ERK` on `MEK`'s site s is enough for the phosphatases to take over.

[In fact the real reason is subtler and has to do with the two specific rules we have chosen to describe how `MEK` detaches from `ERK`. Both require `MEK` to be doubly phosphorylated. So if `PP2A2` dephosphorylates any of the two `MEK` sites while `MEK` is attached to an `ERK`, `MEK` and `ERK` will no longer be able to detach! That is true in both the original model and the variant. However in the original model, it is always possible for `PP2A2` to detach, `RAF` to take its place, and then rephosphorylate its substrate `MEK`, so that `MEK` can eventually detach. Not so in the variant where `RAF` cannot access a `MEK` attached to an `ERK`, since `MEK` uses the same site s to bind downstream `ERK` and upstream `RAS`. This shows how subtle the behaviour induced by a rule set can be.]

5   Conclusion

So how does a rule-based approach compares in terms of flexibility? As said earlier the number of generated species in the model we have considered is about six hundred (techniques to count the number of reachable complexes have been developed in Ref. [4]). A differential model, were it to take all combinations into account would be already quite large, because such models only know of reactions between structure-less entities. So a variation such as the one we performed would probably be very tedious, and in fact rapidly impossible for larger examples. If, on the other hand, the model was not considering all possibilities, then it could be even harder to modify, since whether what was significant before the variation still is so after is not known, and one would have to guess again. Process-algebra based notations record the behaviour of each agent in the agent itself (the reason why they are called process-centric) and that would also make the kind of variation delineated above less of an easy operation -and the tedium of having to emumerate internal states is still a problem.

Note also that one has only considered a simple example here, and that the gain in flexibility one obtains using rules is more pronounced with larger models [3]. It is also likely to be more useful for modelling situations where one knows less and there is greater uncertainty about which specific mechanisms are in place, because these will command most certainly many variations on the structure of the model to be explored.

References

[1] Cardelli, L., *Brane calculi*, in: *Proceedings of BIO-CONCUR'03, Marseille, France*, Electronic Notes in Theoretical Computer Science **180** (2003).

[2] Ciocchetta, F., S. Gilmore, M. L. Guerriero and J. Hillston, *Integrated simulation and model-checking for the analysis of biochemical systems integrated simulation and model-checking for the analysis of biochemical systems* (2008), to appear.

[3] Danos, V., J. Feret, W. Fontana, R. Harmer and J. Krivine, *Rule-based modelling of cellular signalling*, in: L. Caires and V. Vasconcelos, editors, *Proceedings of the 18th International Conference on Concurrency Theory (CONCUR'07)*, LNCS **4703**, 2007, pp. 17–41.

[4] Danos, V., J. Feret, W. Fontana and J. Krivine, *Abstract interpretation of cellular signalling networks*, in: F. L. et al., editor, *VMCAI'08*, LNCS **4905** (2008), pp. pp. 83–97.

[5] Danos, V. and J. Krivine, *Formal molecular biology done in CCS*, in: *Proceedings of BIO-CONCUR'03, Marseille, France*, Electronic Notes in Theoretical Computer Science **180** (2003), pp. 31–49.

[6] Danos, V. and C. Laneve, *Formal molecular biology*, Theoretical Computer Science **325** (2004), pp. 69–110.

[7] Gillespie, D. T., *Exact stochastic simulation of coupled chemical reactions*, J. Phys. Chem **81** (1977), pp. 2340–2361.

[8] Goldbeter, A. and D. Koshland, *An amplified sensitivity arising from covalent modification in biological systems*, Proceedings of the National Academy of Sciences **78** (1981), pp. 6840–6844.

[9] Hlavacek, W., J. Faeder, M. Blinov, R. Posner, M. Hucka and W. Fontana, *Rules for Modeling Signal-Transduction Systems*, Science's STKE **2006** (2006).

[10] Huang, C. and J. Ferrell, *Ultrasensitivity in the mitogen-activated protein kinase cascade*, Proceedings of the National Academy of Sciences **93** (1996), pp. 10078–10083.

[11] Priami, C. and P. Quaglia, *Beta binders for biological interactions*, Proceedings of CMSB'04 **3082** (2004), pp. 20–33.

[12] Regev, A., W. Silverman and E. Shapiro, *Representation and simulation of biochemical processes using the pi-calculus process algebra*, in: *Pacific Symposium on Biocomputing*, 2001, pp. 459–470.