Rule-based modelling, symmetries, refinements

Vincent Danos[1,3,4], Jérôme Feret[2], Walter Fontana[3], Russell Harmer[3,4], and Jean Krivine[5]

[1] University of Edinburgh
[2] École Normale Supérieure
[3] Harvard Medical School
[4] CNRS, Université Paris Diderot
[5] École Polytechnique

Abstract. *Rule-based modelling is particularly effective for handling the most combinatorial aspects of cellular signalling. The evolution of signalling systems is described in terms of basic interactions between partial complexes. This gives access to readable and tractable descriptions of what otherwise could be nearly infinite dimensional dynamical systems. Such partial complexes can be extended, or refined, to provide more specific conditions for an interaction to happen. We set out in this paper to study this notion of refinement, and the related notion of symmetry.*
Specifically we present a method to refine rule sets in a way that preserves the implied stochastic semantics. The stochastic semantics of a rule set is dictated by the number of different ways in which a given rule can be applied to a system (obeying the mass action principle). The refinement formula we obtain explains how to refine rules and which choice of refined rates will cancel out symmetries lost or gained during refinement. It has a pleasing mathematical simplicity, and we expect it to be reusable with little modification across many variants of stochastic graph rewriting. A particular case of the above is the derivation of a maximal refinement which is equivalent to a (possibly infinite) Petri net.

1 Semi-liquid computing

To the eye of the computational scientist cellular signalling looks like an intriguing computational medium. Various types of agents (proteins) of simple design interact in what at first sight may seem to be a completely liquid universe of random encouters. But in fact not, a rich decentralized choreography of bindings (complex formation) and mutual modifications (post-translational modifications) can be observed. Transient devices (complexes) are built by agents to integrate, convey, and amplify signals and channel to the appropriate outputs (transcriptional regulation). The intricate pathways of the response to growth factors sketched in Fig. 1 is a well-studied example. This universe of semi-liquid computing is brought about by a surprisingly little number of elementary interactions. It sits somewhere in between the worlds of the random graphs of statistical physics which perhaps lack expressivity [1], and the solid colliding sphere models of chemical kinetics which perhaps lack programmability [2].

The generativity of those systems, that is to say the number of different non isomorphic combinations (aka complexes or species) that may come to exist along different realizations of the implied stochastic process, may well be enormous, but this does not say how complex the system really is. A lot fewer rules than there are reactions (interactions between complete complexes) may be good enough to describe it. For instance the sketch of Fig. 1 once properly formalized uses about 200 rules whereas it produces about 10^{40} combinations; and that is a more meaningful estimate of the inherent complexity.

Rule-based languages [3–10], and more generally process algebraic approaches to modelling [11–18], because they can express such generic interactions, can work around this apparent descriptive complexity and achieve compact descriptions. Beyond this descriptive compactness there is another benefit to rule-based modelling which is that one can trace the evolution of a system at the level of agents (or individuals) and explore the causal relationships between events occurring in a system [5], but we will not treat this aspect of the question here.

The difference between an assembly of agents with random uncorrelated encounters and a signalling system is that there is a causal structure channelling the interactions towards a particular response. Typically a binding will not happen before one or both of the bindees has been modified. Combining those micro-causal constraints into a coherent pathway is a programming art that we don't master or understand yet, but which signalling systems have honed for a considerable time. Rule-based modelling incorporates such causality constraints in the rules themselves by using partial complexes; the more detailed the complex the more conditions to meet for a particular event to happen.

The purpose of the present paper is to investigate the algebraic structure that underlie such conditions on rule application. We consider the problem of replacing a rule with a family of refined rules which will exhibit the same collective activity, and will therefore generate an identical stochastic behaviour. This theoretical investigation reveals an intimate connexion between this question and the notion of symmetry.

There are also more practical reasons why we think this investigation is of interest. First it may happen that in a modelling situation a rule has to be revised because people come to believe that its rate depend on more information about the context than the rule actually provide. This procedure, called a kinetic refinement, can be usefully decomposed as first introducing a neutral refinement -as defined in this paper- and only then changing the rates. In this application the neutral refinement serves as a baseline. Another interest is to consider a maximal expansion, that is to say replacing a rule with all its ground instances (in general an infinite set). One can use this to map a entire rule set to a set of multiset rewriting rules obtaining a (possibly infinite) Petri net. This transformation is probably unfeasible in general, but one can easily imagine that running truncated versions of a ground expansion using an ODE semantics could be useful for model calibration, or any exploration mechanism that is particularly demanding in terms of the number of simulations required.

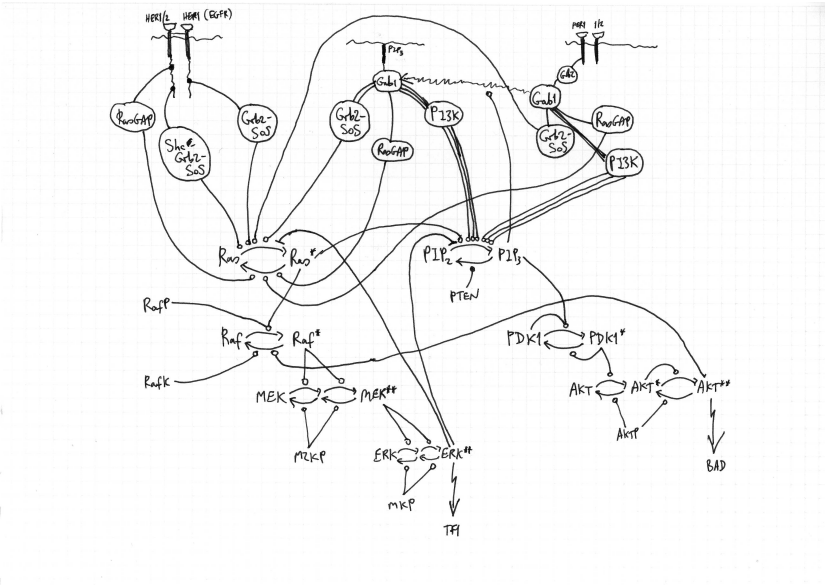


Fig. 1. *A informal sketch of the many interactions involved in the ERK/AKY pathway responding to an EGF stimulus. The corresponding rule-based model generates about* 10^{40} *different species.*

We obtain a refinement formula of general validity and assumes nothing about the arity of the rule to be refined, and actually assumes little about the rewriting framework itself. We also explain how to sharpen this result by considering more inclusive notions of refinements.

For the sake of the presentation, we have simplified somewhat the Kappa framework. The first refinement formula is treated in detail; extensions are only sketched. Note that there is no explanation given in this paper, further than the ones given above, about the relevance of Kappa for the actual practical modelling of signalling. The interested reader can consult Ref. [5, 6]. As a mathematical development, the paper is self-contained.

2 A simple refinement

We start with a simple refinement example. That is an occasion to get familiar with some of the notations. It is also an illustration of what the notion of refinement is trying to achieve.

Consider two agent types B, C each with only one site x, and define a family of systems $x(n_1, n_2)$ consisting of n_1 single Cs, ie $C(x)$ and n_2 dimers $C(x^1), B(x^1)$ (in textual notation binding is represented by sharing an exponent). Now consider the rule $r := C(), B() \to_1 C()$ and suppose it has rate 1. Note that r does not mention x at all (we say that both agents have an empty

signature in this case). This means that r applies regardless of the binding state of x in B and C. Both agents could be free, or bound, or even bound together. Whichever is the case, the effect of the rule will be the same, namely to delete a B and to bring $x(n_1, n_2)$ to a new state $x(n_1 + 1, n_2 - 1)$ (if $n_2 > 0$).

Now we would like to refine r into mutually exclusive sub-cases depending on the relationship in which C and B stand; specifically we want to use the following three refined rules:

$$
\begin{aligned}
r_1 &:= C(x^1), B(x^1) \to_1 C(x) \\
r_2 &:= C(x^1), B(x^1), C(x^2), B(x^2) \to_2 C(x), C(x^2), B(x^2) \\
r_3 &:= C(x^1), B(x^1), C(x) \to_1 C(x), C(x)
\end{aligned}
$$

Each of them is a particular case of r in the sense that their left hand sides embed (sometimes in more than one way) that of r (see below the notion of morphism). Intuitively, r_1 is the sub-case where B, C are bound together, r_2 is the sub-case where they are both bound but not together, and r_3 is the sub-case where B is bound but C is free. Given the particular family of states $x(n_1, n_2)$ we are dealing with, those seem to cover all possible cases, and to be indeed exclusive.

Define the activity of a rule as the number of possible ways to apply the rule multiplied by its rate. This determines its likelihood to apply next and only depends on the current state of the system. Now we have chosen for each refined rule a particular rate, and r_2 has been assigned a rate of 2. We claim this is the unique correct choice if one wants the stochastic behaviour of the system to be preserved by the refinement. This is what we would like to prove in general. Figure 2 shows a run of the refined system with $x(0, 100)$ as the initial state. The y axis traces the activity of all rules including the base one r. We see indeed that at any one time the refined activities add up to the original one (the top curve).

There are other things worth noticing, namely that: r_1 keeps a low probability that decreases linearly during the simulation since its activity is exactly the number of dimers n_2; r_2 dominates the early events, since near the initial state there is only dimers, an no free Cs yet; as time passes there will more Cs, and the other rule r_3 will come to dominate. Hence we see that the relative importance of the sub-cases changes over time.

The corrective factor applied to r_2 accounts for two opposite effects: on the one hand r embeds in many ways into r_2 which would tend to scale the rate of r_2 upwards, on the other hand r_2 is more symmetric than r and that would tend to scale the rate of r_2 downwards.

What we are interested in is to handle the general case, i.e., to explain what constitutes a good set of refined rules as r_1, r_2, and r_3 above, and given such a set, to explain how one can compute the new refined rates in a way that activity is preserved. We will return to the example once we have a general solution.

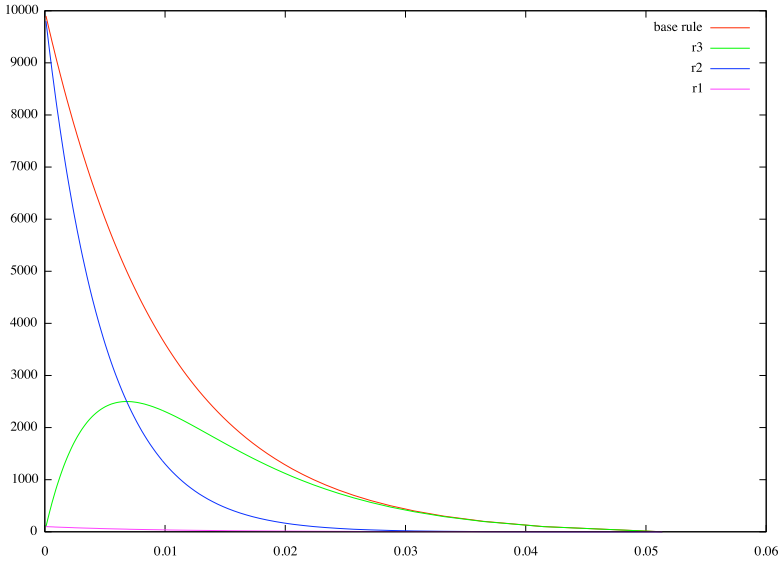


Fig. 2. *The activities of the refined rules r_1, r_2, and r_3 add up exactly to that of the initial rule r (top curve).*

3 Rule-based modelling

To give proper generality and portability to our study, we will frame the system into some simple categorical language where a system is seen as an object x and places in x where a rule may apply are identified using a notion of morphism f from the rule left hand side to x.

As said we shall also simplify the Kappa syntax in two respects. First we suppose agents have no internal states. Second, we also suppose no wildcards are used in left hand sides, e.g., expressions like $A(x^{\text{-}})$ meaning x is bound to some unspecified other site, are not considered. The former simplification is only a matter of readability, as internal values offer no difficulty. The latter simplification is more significant, and we will see later in our development that reintroducing wildcards allows to strengthen our main result. With those simplifications we can give a syntax-less presentation of Kappa that will greatly facilitate counts and the ensuing derivation of the refinement formula.

We suppose given two sets $\mathcal{A}$ and $\mathcal{S}$ of agent names and sites.

A matching over a set X is an irreflexive and symmetric binary relation μ on X such that no element is in relation with more than one element.

Definition 1 *An object is a quadruple $(V, \lambda, \sigma, \mu)$ where:*
– *V is a set of nodes,*
– *$\lambda \in \mathcal{A}^V$ assigns names to nodes,*
– *$\sigma \in \mathcal{P}(\mathcal{S})^V$ assigns sets of sites to nodes,*
– *μ is a matching over $\sum_{v \in V} \sigma(v)$.*

The matching represents bindings, and hence any given site can be bound at most once. A node however can be bound many times via different sites.

We define $(u, x) \in \mu$ as shorthand for $\exists(v, y) : (u, x, y, v) \in \mu$, and say u, x is *free* when $(u, x) \notin \mu$, *bound* when $(u, x) \in \mu$.

The simplest non empty object is a single node named A with no sites and therefore no binding. As in the preceding section we write $A()$ for this object. There we also introduced a textual notation to designate objects where bindings are indicated by exponents.

Note that we sometimes use the same family of symbols x, y, etc. for sites and objects. Hopefully this will not cause any confusion since those are entities of a very different nature.

We define a *signature* as a map $\Sigma : \mathcal{A} \to \mathcal{S}$; that can be used to constrain the set of sites per agent type. We write $x \leq \Sigma$ if for all $v \in V_x$, $\sigma_x(v) \subseteq \Sigma(\lambda_x(v))$; likewise we write $\Sigma \leq x$ if for all $v \in V_x$, $\Sigma(\lambda_x(v)) \subseteq \sigma_x(v)$, and $x : \Sigma$ when $x \leq \Sigma \leq x$.

When $x : \Sigma$ for some Σ, we say x is *homogeneous*, which means all agent of the same type in x use exactly the same set of sites.

Definition 2 *An arrow $(V, \lambda, \sigma, \mu) \to (V', \lambda', \sigma', \mu')$ is a map $f : V \to V'$ such that*
– 1) f preserve names: $\lambda' \circ f = \lambda$
– 2) f preserve sites: $\sigma' \circ f \supseteq \sigma$
– 3a) f preserve edges: $(u, x, y, v) \in \mu \Rightarrow (f(u), x, y, f(v)) \in \mu'$
– 3b) f reflects edges: $(f(u), x) \in \mu', (u, x) \in V \Rightarrow (u, x) \in \mu$
– 4) f is a monomorphism

This then is the category of *graphs with sites* we shall work with. We also call arrows morphisms sometimes; we write $[x, y]$ for the arrows from x to y; $iso[x, y]$ for the isomorphisms (meaning invertible arrows), and therefore $[x, x] = iso[x, x]$ denotes the set of *automorphisms* (or symmetries) of x; we say that y *embeds* x when $[x, y] \neq \varnothing$.

Define the *image* of $f \in [x, y]$ as $Im(f) := \{f(v), x; v \in V, x \in \sigma(v)\}$.

Note that $Im(f)$ is but a subset of $\sum_{v \in V} \sigma'(f(v))$, and only sites in $Im(f)$ are mentioned in the arrow-defining clauses above.

One has obviously a forgetful functor to the category of graphs and graph morphisms, and that allows us to import the usual graph-theoretical vocabulary of connected components and paths, which we will freely use in the sequel. Note that from the point of view of graphs the reflectivity condition 3b) above does not really make sense, one really needs sites to express edge reflection. Moreover the rather stringent notion of arrow constrains a lot the homsets $[x, y]$:

Lemma 1 (rigidity) *Suppose x is connected, then any non-empty partial injection f from V_x to V_y extends to at most one morphism in $[x, y]$.*

Proof: If f is strictly partial, that is to say $V_x \setminus dom(f)$ is not empty, pick a v in there such that for some node $w \in dom(f)$, and some sites x, y, $(w, y, v, x) \in \mu_x$. This is always possible because x is connected. Then, either $(f(w), y, v', x) \in \mu_y$

for some $v' \in V_y$, and by 3a) must extend f as $f(v) = v'$, or there is no such extension. $\square$

Clearly being a monomorphism, ie being post-cancellable, is equivalent to being a one-one map. On the other hand there are far more epimorphisms than surjections:

Lemma 2 (epis) *A map $h \in [x, y]$ is an epimorphism iff every connected component of y intersects $f(x)$; that is to say for all connected component $c_y \subseteq y$, $h^{-1}(c_y) \neq \varnothing$.*

Proof: Suppose $f_1 h = f_2 h$ for $h \in [x, y]$, $f_i \in [y, z]$, and let $c_y \subseteq y$ be a connected component of y such that $h^{-1}(c_y) \neq \varnothing$. Pick u such that $h(u) \in c_y$, then $f_1(h(u)) = f_2(h(u))$ and by the preceding lemma $f_1/c_y = f_2/c_y$. $\square$

We write $[x, y]^e \subseteq [x, y]$ for the epis from x to y.

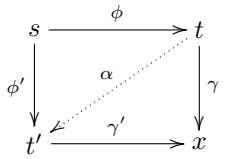
4 Object refinements

Now that we have our basics in place we turn to the first question of what constitutes a refinement of a (partial) object s. As we have seen in the example a refinement of s is intuitively a collection of objects t_i that embed s, and such that any embedding in an object of higher signature x (ie that has everywhere more sites) can be unambiguously attributed to one t_i. We first make this intuition into a real definition and then proceed to define the refinements of rules.

Definition 3 (factorisation) *One says an object t factors $f \in [s, x]$ if $f = \gamma\phi$ for some $\phi, \gamma \in [s, t]^e \times [t, x]$; ϕ, γ is called a factorisation of f via t.*

The first thing to notice is that one cannot ask for unique factorisations.

Indeed, suppose given a factorisation $\phi, \gamma \in [s, t]^e \times [t, x]$ of f via t and an *isomorphism* $\alpha \in [t, t']$. Define $\phi', \gamma' := \alpha\phi, \gamma\alpha^{-1} \in [s, t'] \times [t', x]$; this new pair verifies $\gamma\phi = \gamma'\phi'$ and so is also a factorisation of f via t'.



(1)

In this case we will say that ϕ, γ and ϕ', γ' are *conjugate* under α, and write $\phi, \gamma \simeq_{tt'} \phi', \gamma'$. We also write $[s, t] \times_{[t,t]} [t, x]$ for the quotient of $[s, t] \times [t, x]$ under $\simeq_{tt}$; that notation is justified by the following:

Lemma 3 (conjugates) *The equivalence relation $\simeq_{tt}$ has $|[s, t] \times [t, x]|/|[t, t]|$ classes.*

Proof: Suppose, using the notations of (1), that $\phi, \gamma \simeq_{tt'} \phi', \gamma'$, then this uniquely determines α since $\gamma\alpha^{-1} = \gamma\alpha'^{-1}$ implies $\alpha = \alpha'$ by γ being a monomorphism.

In particular the set of conjugates of ϕ, γ over the same t is in one-one correspondence with $[t,t]$. $\square$

Unicity of factorisation is then to be understood up to isomorphisms; furthermore, even if one selects one representative t_i per isomorphy class, unicity is up to automorphisms of each of the representative t_i.

Definition 4 (object refinement) *Given s, Σ such that $s \leq \Sigma$, a refinement of s under Σ, written $\Sigma(s)$, is a set of objects obtained by selecting one representative in each isomorphism class defined by $\{t \mid t : \Sigma, [s,t]^e \neq \varnothing\}$.*

Note that the actual choice of representatives does not matter, but we do have to choose one for our counting purposes.

Another noteworthy fact is that $\Sigma(s)$ in general will be infinite. However in practice one may get information about the reachables of the system which will allow to control the size of the expansion [3]; indeed it is not necessary to include ts which are not reachable, and we took advantage of this in the example of the first section.

Lemma 4 (injectivity) *Given Σ, s, x such that $s \leq \Sigma \leq x$ the composition map from the disjoint sum $\sum_{t \in \Sigma(s)} [s,t]^e \times_{[t,t]} [t,x]$ to $[s,x]$ is injective.*

Proof: Suppose given two factorisations $f = \gamma\phi = \gamma'\phi'$ via t and t' as in (1).

$$\begin{array}{ccccc} v \in s & & \xrightarrow{\quad\phi\quad} & & t \supseteq c \ni \phi(v) \\ \Big\downarrow{\scriptstyle \phi'} & & \overset{\alpha}{\cdots\cdots\swarrow} & & \Big\downarrow{\scriptstyle \gamma} \\ \phi'(v) \in c' \subseteq t' & & \xrightarrow{\quad\gamma'\quad} & & x \end{array} \tag{2}$$

Pick a connected component $c \subseteq t$, such that $\phi(v) \in c$ for some $v \in s$. Call $c' \subseteq t'$ the connected component of $\phi'(v)$ in t'. By construction $\gamma(c)$ and $\gamma'(c')$ intersect at $\gamma\phi(v) = f(v) = \gamma'\phi'(v)$. It is easy to see that they both are Σ-homogeneous. This means they must be equal.

Indeed suppose $w \in \gamma(c)$ is a node which is directly connected to $\gamma(c) \cap \gamma'(c')$, meaning w is such that $(u,x,y,w) \in \mu_x$, for some $u \in \gamma(c) \cap \gamma'(c')$ and $(u,x),(w,y) \in Im(\gamma)$. Because c' is Σ-homogeneous, $u, x \in Im(\gamma')$, ie x is also a site of the (unique) antecedent of u in c', which we can write $x \in \sigma_{t'}\gamma'^{-1}(u)$. By condition 3b) this site cannot be free, and by 3a) it must be bound to $\gamma'^{-1}(w), y$, so $w \in \gamma'(c')$. Since $\gamma(c)$ is connected, $\gamma(c) \cap \gamma'(c')$ must contain $\gamma(c)$, and by symmetry $\gamma'(c')$.

Hence $\gamma'(c') = \gamma'(c')$, therefore c and c' are isomorphic. In fact, since ϕ is an epi, we can repeat the above for any connected component in t, and therefore t embeds in t' (it is readily seen that the assignment of a c' to a c above is injective), and by symmetry they must be isomorphic under a certain isomorphism α. By definition of $\Sigma(s)$ we have picked exactly one representative in each isomorphism class, therefore $t = t'$, $\alpha \in [t,t]$, and the two factorisations are conjugate under α. $\square$

Theorem 1 *Given Σ, s, x such that $s \leq \Sigma$ and $x : \Sigma$:*

$$[s, x] \simeq \sum_{t \in \Sigma(s)} [s, t]^e \times_{[t,t]} [t, x]$$

Proof: From the preceding lemma we know the composition map is injective, so all there remains to prove is that it is surjective.

Consider $f \in [s, x]$, define $f(s) := \{u \mid \exists x : (u, x) \in Im(f)\} \subseteq V_x$, and write $[f(s)]$ for the connected closure of $f(s)$ in x. We claim there is a $t \in \Sigma(s)$ which is isomorphic to $[f(s)]$. Indeed every node in $[f(s)]$ has a signature in accordance with Σ because $x : \Sigma$, and $[f(s)]$ embeds s since $f(s)$ does (via f). □

Using Lemma 3 in addition we can use the above theorem to obtain:

Corollary 1 *Given Σ, s, x such that $s \leq \Sigma$ and $x : \Sigma$, one has:*

$$|[s, x]| = \sum_{t \in \Sigma(s)} |[s, t]^e|/|[t, t]| \cdot |[t, x]| \tag{3}$$

There are several things worth noticing about the theorem and its numerical form as a corollary.

First the $|[s, t]^e|/|[t, t]|$ is a *static* term that can be computed once and for all, and which we shall use to the rule rates. The positive contribution $[s, t]^e$ is rather intuitive since the more copies of s one finds in t the higher the contribution of t to the number of instances of s should be; the negative contribution is a lot less intuitive however.

Second one cannot relax the homogeneity condition on x and ask only $\Sigma \leq x$. That would break the easy part of the proof, namely that of surjectivity. Here is an example; set $s := A(x) < \Sigma := A \mapsto \{x, y\} < A(x, y^1, z), A(x, y, z^1) =: x$. Choose f to be the 'left' morphism mapping s's unique A to $A(x, y^1, z)$ in x; then $[f(s)] = x$ and no $t \in \Sigma(s)$ can factorise f because the (y, z) binding is not reproducible in t, because $z \notin \Sigma(A)$.

However one can modify the notion of object (and accordingly that of arrow) by introducing new partial objects such as $t = A(x, y^{\neg \Sigma})$, meaning y binds an otherwise unspecified non-Σ site (ie A, y is bound to some B, z such that $z \notin \Sigma(B)$). This t is homogeneous and factorises the f above. This variant allows to recover surjectivity and extend our decomposition theorem above. Similar wildcard expressions are already present in the actual syntax of Kappa, and it is amusing to see that those convenient notations have also a theoretical status.

This begs a last remark, namely that we are the ones choosing how to relate the base object s and its refinement. E.g., here, we are using epis to relate them. Below we will allude to a finer-grained correspondence based on using a pointed version of the ambient category that will allow us to go beyond the homogeneity requirement in another way. But before we do that we will return to the example of the first section.

4.1 Example continued

We can now reconsider our initial example. Set $s := C(), B()$, for the left hand side of the base rule r, and t_i for that of the refined rule r_i:

$$\begin{aligned} t_1 &:= C(x^1), B(x^1) \\ t_2 &:= C(x^1), B(x^1), C(x^2), B(x^2) \\ t_3 &:= C(x^1), B(x^1), C(x) \end{aligned}$$

Set also $\Sigma := B, C \mapsto \{x\}$. Clearly $s < \Sigma$ and t_i, and $x(n_1, n_2)$ are Σ-homogeneous. Besides due to the particular form of $x(n_1, n_2)$, the t_is are the only elements in $\Sigma(s)$ that $x(n_1, n_2)$ embed. Using Lemma 3 we get:

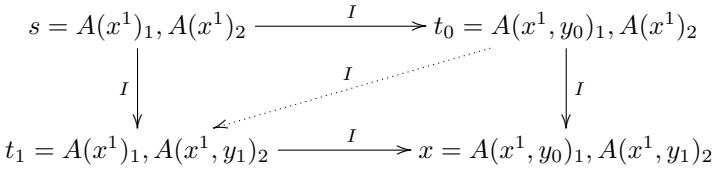
$$\begin{aligned} &|[s, x(n_1, n_2)]| = n_2(n_1 + n_2) \\ &|[s, t_1]^e \times_{[t_1, t_1]} [t_1, x(n_1, n_2)]| = |[s, t_1]^e[t_1, x(n_1, n_2)]|/|[t_1, t_1]| = 1.n_2/1 = n_2 \\ &|[s, t_2]^e \times_{[t_2, t_1]} [t_2, x(n_1, n_2)]| = |[s, t_2]^e[t_2, x(n_1, n_2)]|/|[t_2, t_2]| = 2.n_2(n_2 - 1)/2 \\ &|[s, t_3]^e \times_{[t_3, t_1]} [t_3, x(n_1, n_2)]| = |[s, t_3]^e[t_3, x(n_1, n_2)]|/|[t_2, t_2]| = 1.n_1 n_2/1 = n_1 n_2 \end{aligned}$$

and the corollary correctly predicts $n_1 n_2 + n_2(n_2 - 1) + n_2 = n_1(n_1 + n_2)$.

4.2 Pointed refinements

Let us look at an example which breaks injectivity. This is the kind of complications the theorem is staying cautiously away from by asking the ts to be homogeneous.

The set of nodes $V_s = \{1, 2\}$ is represented as subscripts to agents below; the subscripts to the y sites, y_0 and y_1, denote bindings to agents with only one site and different names (to save space):



If one refers to the situation of (1), the unique possible candidate conjugating α, i.e., the unique diagonal that makes both triangle commute, fails to be a morphism. That means that t_0, t_1 provide really distinct extensions of $f(s)$ in x and form an ambiguous decomposition of s. Indeed applying (wrongly since the t_is are not homogeneous) the refinement formula (3) betrays this redundancy problem since $|[s, x]| = 2$ while $|[s, t_i]|/|[t_i, t_i]||[t_i, x]| = 2$.

To deal with a case such as this one, one needs to break the symmetry. To do this a possibility is to work out the static part of the refinement formula in a *pointed* subcategory where objects have in addition to their usual structure a distinguished node (and so are supposed to be non-empty), and arrows are asked to preserve it. Then one can replace homogeneity by a weaker requirement, namely that across all expansions of s no two agents with the same coordinates

with respect to a distinguished node differ in their signature. In the example above that would force to decide whether the additional binding is to sit on the distinguished node or not, and *then* both extensions would become truely distinct and unambiguous. Obviously a little more work is needed to say with complete confidence that this will work, but it seems it will.

5 Rule refinements

Now that we know how to refine objects, we will proceed to the case of rules.

5.1 Action, rules, events

An atomic action on s is one of the following:
- an edge addition $+(u, x, y, v)$
- an edge deletion $-(u, x, y, v)$
- an agent addition $+(A, \sigma)$ with A a name, σ a set of free sites
- an agent deletion $-(u)$ with $u \in V_s$, $v \in V_s$, $x \in \sigma_s(u)$, and $y \in \sigma_s(v)$.

An action on s is a finite sequence of atomic actions on s. An atomic action is well defined on s:
- if $\alpha = +(u, x, y, v)$, when both (u, x) and (v, y) are free in s,
- if $\alpha = -(u, x, y, v)$, when $(u, x, y, v) \in \mu_s$.

This notion extends readily to non-atomic actions; we consider only well-defined actions hereafter.

Definition 5 *A rule is a triple $r = s, \alpha, \tau$ where:*
- s in an object,
- α is an action on s,
- and τ a rate which can be any positive real number.

We write $\alpha \cdot s$ for the effect of the action s on s.

Given $f \in [s, x]$ and α there is an obvious definition of the transport of α along f, written $f(\alpha)$, and it is easy to verify that $f(\alpha)$ is itself a well-defined action on x if α is a well-defined action on s (condition 3b) is crucial though).

Definition 6 *A set R of rules defines a labelled transition relation:*

$$x \longrightarrow_f^{s,\alpha,\tau} f(\alpha) \cdot x \tag{4}$$

where $s, \alpha, \tau \in R$, and $f \in [s, x]$.

The labelled transition system just defined can be enriched quantitatively in a way that generalizes the notion of stochastic Petri nets [19] (Petri nets correspond to the case of a uniformly empty signature $\Sigma = \varnothing$).

To do this we need to define the activity of a rule.

Definition 7 *Given an object x and a rule $r = s, \alpha, \tau$, the activity of r at x is $a(x, r) := \tau|[s, x]|$, and the global activity of a set of rules R at x is $a(x) := \sum_{r \in R} a(x, r)$.*

Supposing $a(x) > 0$, the probability at x that the next event is $f \in [s, x]$ is $p(x, f) := \tau/a(x)$, and the subsequent time advance is a random variable $\delta t(x)$ such that $p(\delta t(x) > t) := e^{-a(x)t}$. For our present purpose, all we need to remember is that the quantitative structure of the transition system is entirely determined by the activities of its rules. In fact this means our result will hold for a larger class of stochastic system for what it is worth.

5.2 The main result

Given a rule $r = s, \alpha, \tau$ and $\theta \in [s, t]$, we define $\theta(r) := \theta(s), \theta(\alpha), \tau$.

We say r, r' are isomorphic rules, written $r \simeq r'$, if there is an isomorphism $\theta \in [s, s']$ such that $r' = \theta(r)$. If that is the case then r and $\theta(r)$ have isomorphic transitions:

$$x \longrightarrow^{r}_{f \in [s,x]} f(\alpha) \cdot x \Leftrightarrow x \longrightarrow^{\theta(r)}_{f\theta^{-1} \in [\theta(s),x]} f\theta^{-1}(\theta(\alpha)) \cdot x$$

and in particular the same activity $a(r, x) = a(\theta(r), x)$.

Definition 8 (rule refinement) *Given s, Σ such that $s \leq \Sigma$ and $r = s, \alpha, \tau$, the refinement of r under Σ is the following family of rules:*

$$\Sigma(s, \alpha, \tau) := (t, \phi(\alpha), \tau; t \in \Sigma(s), \phi \in [s, t]^e/[t, t]) \tag{5}$$

where the notation $\phi \in [s, t]^e/[t, t]$ means that for each t, one selects one $\phi \in [s, t]^e$ per symmetry class on t (the equivalence relation $\exists \theta \in [t, t] : \phi = \theta\phi'$).

It is easily seen that the particular selection made is irrelevant, but one has to choose one to define refinement as a syntactic transformation.

Note also that the above family can have isomorphic or even identical rules, it is important to have them all, i.e., *stricto sensu* the expansion is a multiset of rules not a set. However one can always pack n isomorphic copies together by choosing a representative and multiplying its rate by n so we carry on with our slight abuse of terminology.

Given R a rule set, r a rule in R, we write $R[r \backslash \Sigma(r)]$ for the rule set obtained by replacing r with $\Sigma(r)$.

We write $r = s, \alpha, \tau \leq \Sigma$ if $s \leq \Sigma$, and $R \leq \Sigma$ if for all $r \in R$, $r \leq \Sigma$.

Theorem 2 *Given R, Σ, such that $R \leq \Sigma$, one has $R[r \backslash \Sigma(r)] \leq \Sigma$, and R and $R[r \backslash \Sigma(r)] \leq \Sigma$ determine the same stochastic transition system over Σ-homogeneous objects.*

Proof: By Th. 1 events $f \in [s, x]$ associated to rule $r = s, \alpha, \tau$ are in one-one correspondence with factorisations $f = \gamma\phi$ via some t, and therefore determine a unique matching refined event γ. This refined event has the same effect as f since:

$$x \longrightarrow^{t,\phi,\tau}_{\gamma \in [t=\phi(s),x]} \gamma\phi(\alpha) \cdot x = f(\alpha) \cdot x$$

so r and its refinements are equally likely and have the same effect on the underlying state x; hence their stochastic transition systems are the same. □

Note that the activity of t, ϕ, τ in the refined system is $\tau|[t,x]|$ so the cumulated activity of the refined rules is:

$$\sum_{t\in\Sigma(s)} \sum_{\phi\in[s,t]^e/[t,t]} \tau|[t,x]| = \sum_{t\in\Sigma(s)} \tau|[s,t]^e|/|[t,t]||[t,x]| = a(r,x)$$

by Coro. 1, so we can directly derive the fact that the refined rules have the same activity, but we also need to prove they have the same effect.

5.3 Example concluded

We can now conclude our initial example.

There we had $s := C(), B()$, and:

$$\begin{aligned} t_1 &:= C(x^1), B(x^1) \\ t_2 &:= C(x^1), B(x^1), C(x^2), B(x^2) \\ t_3 &:= C(x^1), B(x^1), C(x) \end{aligned}$$

Since $|[s,t_2]^e| = 2$ (recall that epis must have images in all connected components), the refinement of r via t_2 will contribute two rules in $\Sigma(r)$. In this particular case the action of the rule to be refined is $\alpha(r) = -B$, and both epimorphisms $\phi \in [s,t_2]$ lead to the same transported action $\phi(-B)$ up to isomorphism. One can then pack them into one rule r_2, as we did intuitively when we considered the example, and as a consequence the rates must be added. This explains why r_2 has a rate of 2.

6 Conclusion

We have presented in this article the beginning of a theory of refinements for rule-based modelling. Specifically we have defined what constitutes a notion of a good set of refined rules, and how given such a set, one can compute the new refined rates in a way that the overall activity of the system is preserved and the underlying stochastic semantics therefore unchanged. We have also suggested two potential improvements that would extend the type of refinement one can consider. Those remain to be verified and proved to work though.

Another point worth commenting in this conclusion is that the formulas obtained in our two main results, Th. 1 and 2, are couched in rather general terms and is likely to be of a larger relevance than the particular case of graph-rewriting we were contemplating here. In particular the epi-mono factorisation system which we rely on implicitly for the concrete case we have treated would point to a more abstract approach. That in itself is valuable since such combinatorial results as we have presented here can become nearly intractable if looked at in a too concrete way. This in fact is one of the reasons why we framed our results in a categorical language which has revealed the pervasiveness of symmetries (the other reason is that the syntax is simpler to deal with).

It would be particularly interesting to recast the theory, to the extent it is possible, in the axiomatic framework of adhesive categories [20], or quasi-toposes, with a view on understanding the formula as a traditional partition formula (which it is at least intuitively).

Finally a longer term goal that this preliminary work might help to reach is that of finding exact model reduction techniques. But that would probably need to lift a key assumption made here, namely that refinements are made of mutually exclusive sub-cases, and we don't know how to do that yet.

References

1. B. Söderberg. General formalism for inhomogeneous random graphs. *Physical Review E*, 66(6):66121, 2002.
2. Daniel T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem*, 81:2340–2361, 1977.
3. Vincent Danos, Jérôme Feret, Walter Fontana, and Jean Krivine. Abstract interpretation of cellular signalling networks. In F. Logozzo et al., editor, *VMCAI'08*, volume 4905 of *LNCS*, pages pp. 83–97, 2008. Springer, Jan 2008.
4. Vincent Danos, Jérôme Feret, Walter Fontana, and Jean Krivine. Scalable simulation of cellular signaling networks. In Z. Shao, editor, *In Proceedings of APLAS 2007*, volume 4807, pages 139–157, 2007.
5. Vincent Danos, Jérôme Feret, Walter Fontana, Russell Harmer, and Jean Krivine. Rule-based modelling of cellular signalling. In Luis Caires and Vasco Vasconcelos, editors, *Proceedings of the 18^{th} International Conference on Concurrency Theory (CONCUR'07)*, Lecture Notes in Computer Science, Sep 2007.
6. Vincent Danos. Agile modelling of cellular signalling. In *Proceedings of ICCMSE'07*, 2007.
7. Michael L. Blinov, James R. Faeder, Byron Goldstein, and William S. Hlavacek. A network model of early events in epidermal growth factor receptor signaling that accounts for combinatorial complexity. *BioSystems*, 83:136–151, January 2006.
8. W.S. Hlavacek, J.R. Faeder, M.L. Blinov, R.G. Posner, M. Hucka, and W. Fontana. Rules for Modeling Signal-Transduction Systems. *Science's STKE*, 2006(344), 2006.
9. ML Blinov, J. Yang, JR Faeder, and WS Hlavacek. Graph theory for rule-based modeling of biochemical networks. *Proc. BioCONCUR 2005*, 2005.
10. James R. Faeder, Michael L. Blinov, Byron Goldstein, and William S. Hlavacek. Combinatorial complexity and dynamical restriction of network flows in signal transduction. *Systems Biology*, 2(1):5–15, March 2005.
11. A. Regev, W. Silverman, and E. Shapiro. Representation and simulation of biochemical processes using the π-calculus process algebra. In R. B. Altman, A. K. Dunker, L. Hunter, and T. E. Klein, editors, *Pacific Symposium on Biocomputing*, volume 6, pages 459–470, Singapore, 2001. World Scientific Press.
12. Corrado Priami, Aviv Regev, Ehud Shapiro, and William Silverman. Application of a stochastic name-passing calculus to representation and simulation of molecular processes. *Information Processing Letters*, 2001.
13. A. Regev and E. Shapiro. Cells as computation. *Nature*, 419, September 2002.
14. C. Priami and P. Quaglia. Beta binders for biological interactions. *Proceedings of CMSB*, 3082:20–33, 2004.

15. Vincent Danos and Jean Krivine. Formal molecular biology done in CCS. In *Proceedings of BIO-CONCUR'03, Marseille, France*, volume 180 of *Electronic Notes in Theoretical Computer Science*, pages 31–49. Elsevier, 2003.
16. A. Regev, E.M. Panina, W. Silverman, L. Cardelli, and E. Shapiro. BioAmbients: an abstraction for biological compartments. *Theoretical Computer Science*, 325(1):141–167, 2004.
17. Luca Cardelli. Brane calculi. In *Proceedings of BIO-CONCUR'03, Marseille, France*, volume 180 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 2003.
18. M. Calder, S. Gilmore, and J. Hillston. Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA. *Transactions on Computational Systems Biology*, 4230:1–23, 2006.
19. Daniel T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. Comp. Phys.*, 22:403–434, 1976.
20. S. Lack and P. Sobocinski. Adhesive and quasiadhesive categories. *Theoretical Informatics and Applications*, 39(3):511–546, 2005.