Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 265 (2010) 73–96

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Combining Model Reductions

Ferdinanda Camporesi[1,4]

Dipartimento di Scienze dell'Informazione
Università di Bologna
Bologna, Italy
&
Laboratoire d'informatique de l'École normale supérieure
(INRIA/ÉNS/CNRS)
Paris, France

Jérôme Feret [1,5]

Laboratoire d'informatique de l'École normale supérieure
(INRIA/ENS/CNRS)
Paris, France

Heinz Koeppl[2,6] Tatjana Petrov[3,7]

School of Computer and Communication Sciences
EPFL
Lausanne, Switzerland

Abstract

Molecular biological models usually suffer from a large combinatorial explosion. Indeed, proteins form complexes and modify each others, which leads to the formation of a huge number of distinct chemical species (i.e. non-isomorphic connected components of proteins). Thus we cannot generate explicitly the quantitative semantics of these models, and even less compute their properties. Model reduction aims at reducing this complexity by providing another grain of observation. In this paper, we propose two unifying frameworks for combining model reductions: we propose a symmetric product operator for combining model reductions for stochastic semantics and we show how to abstract further existing reduced differential systems by the means of linear projections. We apply both frameworks so as to abstract further existing reduced quantitative semantics of the models that are written in Kappa, by taking into account symmetries among binding sites in proteins.

Keywords: rules-based modeling, model reduction, abstract interpretation, symmetries.

doi:10.1016/j.entcs.2010.08.006

1 Introduction

Signaling pathways describe the interactions between some proteins which are involved in communication between and within cells. These pathways usually suffer from a combinatorial blow-up in the number of chemical species (pairwise non-isomorphic connected components of proteins). Rules-based modeling [9,1] offers a convenient and compact solution for describing these pathways (and other molecular biological systems as well). The combinatorial complexity is avoided thanks to context-free rules, in which the set of all potential contexts of application for an interaction does not need to be written explicitly.

Yet, the combinatorial complexity raises again when one is interested in the quantitative semantics of rules-based models. Stochastic semantics (based on the use of CTMCs, or master equations) and differential semantics cannot be explicitly written, because the state space is a vector space the dimension of which is the number of reachable species. Model reduction [2,5,10,7,11] consists in reducing this dimension, by discovering a coarser grain of observation. Sound and automatic model reduction can be achieved by the means of formal methods. The framework in [10,7] for reducing differential semantics is based on the fact that rules cannot observe the correlation between specific parts of some chemical species. Thus these chemical species can easily be cut into fragments. In [11] backward bisimulations [4] are used in order to ensure that rules cannot enforce correlations between the state of some identified parts of chemical species.

In this paper, we propose two generic constructions to combine model reductions, one for stochastic semantics and one for differential semantics. In Sect. 2, we give a motivating example: we show that fragments-based model reductions can be abstracted further by taking into account the fact that some binding sites have exactly the same capabilities of interactions (we say that these sites are *symmetric*). In Sect. 3, we propose a generic framework for reducing stochastic semantics and combining these abstractions. This framework is based on the use of backward bisimulations [4] in order to prove statistical invariants, and use these invariants to reduce the state space. We propose a binary product for combining backward bisimulation-based model reductions, so as to build the least model reduction which is at least as much as abstract as each model reduction that is given as an argument of this operator. In Sect. 4, we use linear projections to abstract further model reductions for differential semantics. Interestingly, the algorithm that is used to generate the reduced model can be adapted to deal with symmetric sites on the fly. Our two frameworks are highly reusable, because they do not require much

[1] The contribution of Ferdinanda Camporesi and Jérôme Feret was partially supported by the AbstractCell ANR-Chair of Excellence.

[2] Heinz Koeppl acknowledges the support from the Swiss National Science Foundation, grant no. 200020-117975/1.

[3] Tatjana Petrov acknowledges the support from SystemsX.ch, the Swiss Initiative in Systems Biology.

[4] Email: campores@di.ens.fr

[5] Email: feret@di.ens.fr

[6] Email: heinz.koeppl@epfl.ch

[7] Email: tatjana.petrov@epfl.ch

soundness assumptions about the relation between the symmetries that are used to quotient further the coarse-grained variables and the reduced models (much of proofs are made once for all in the non reduced model).

Then we apply our framework on the models written in Kappa: in Sect. 5, we give the operational semantics of Kappa. In Sect. 6 we give the definition of symmetric sites in Kappa, we review the stochastic and differential semantics, and we use symmetric sites, so as to reduce further the dimension of the state space of these semantics.

2 Case study

Let us start out with a simple motivating example. We consider two kinds of agents P and X. Instances of P denote phosphate ions, whereas instances of X denote copies of a given protein. We assume that each protein X has two kinds of sites: m sites x_1,. . . ,x_m and n sites y_1,. . . ,y_n (m and n are two integer parameters of our model). Each site can recruit at most one phosphate ion P and, then, dissociate from it. The state of a protein X is denoted as a (ordered) tuple of symbols among $\{u, p\}$. The symbol p stands for a phosphorylated site, whereas the symbol u stands for an unphosphorylated site. For instance, with $m = 2$ and $n = 1$, a protein X having the sites x_1 and y_1 phosphorylated and the site x_2 unphosphorylated is denoted by $X[p, u, p]$.

We assume that for each integer i between 1 and m the phosphorylation of the site x_i does not depend on the phosphorylation state of the other sites. The sites $x_1, \ldots, x_m$ can all be phosphorylated at a same rate k. Nevertheless, we also assume that for each integer j between 1 and n, the phosphorylation of the site y_j depends both on the index j of the site y_j and on the number of sites among $x_1, \ldots, x_m$ which are currently phosphorylated in the protein X: the rate of activation of the site y_j in a protein X which have exactly i phosphorylated sites among $x_1, \ldots, x_m$ is denoted by $k_{j,i}$. Last, we assume that any phosphorylated site can be unphosphorylated at a same rate k_d.

We give in Fig. 1 the set of reactions for the model with parameters $m = 2$ and $n = 1$. In the general case, there are 2^{m+n} reachable configurations for the protein X. Thus, when $m + n$ gets big, we can no longer enumerate chemical species (nor reactions). We can sample stochastic semantics by using agents-based simulation algorithms [8]. But the integration of the differential semantics, or the computation of more complex properties about the distribution of traces (or states) is impossible due to the combinatorial complexity.

We notice that we can use symmetries among the sites $x_1, \ldots, x_m$ so as to reduce the dimension of both the stochastic and the differential semantics. Indeed what is important, is not which sites x_i are phosphorylated in a given protein, but how many are phosphorylated. We introduce an equivalence relation $\sim$ over proteins X: we say that two proteins are $\sim$-equivalent if, and only if, (i) the number of phosphorylated sites among the list $x_1, \ldots, x_m$ is the same for both proteins, and (ii) the phosphorylation state of the sites y_j is the same in both proteins, for any integer

$$P + X[u,u,u] \underset{k_d}{\overset{k}{\rightleftharpoons}} X[p,u,u]$$
$$P + X[u,u,p] \underset{k_d}{\overset{k}{\rightleftharpoons}} X[p,u,p]$$
$$P + X[u,p,p] \underset{k_d}{\overset{k}{\rightleftharpoons}} X[p,p,p]$$
$$P + X[u,p,u] \underset{k_d}{\overset{k}{\rightleftharpoons}} X[p,p,u]$$

(a) Phosphorylation and dephosphorylation of the first site.

$$P + X[u,u,u] \underset{k_d}{\overset{k}{\rightleftharpoons}} X[u,p,u]$$
$$P + X[u,u,p] \underset{k_d}{\overset{k}{\rightleftharpoons}} X[u,p,p]$$
$$P + X[p,u,p] \underset{k_d}{\overset{k}{\rightleftharpoons}} X[p,p,p]$$
$$P + X[p,u,u] \underset{k_d}{\overset{k}{\rightleftharpoons}} X[p,p,u]$$

(b) Phosphorylation and dephosphorylation of the second site.

$$P + X[u,u,u] \underset{k_d}{\overset{k_{1,0}}{\rightleftharpoons}} X[u,u,p]$$
$$P + X[u,p,u] \underset{k_d}{\overset{k_{1,1}}{\rightleftharpoons}} X[u,p,p]$$
$$P + X[p,p,u] \underset{k_d}{\overset{k_{1,1}}{\rightleftharpoons}} X[p,p,p]$$
$$P + X[p,u,u] \underset{k_d}{\overset{k_{1,2}}{\rightleftharpoons}} X[p,u,p]$$

(c) Phosphorylation and dephosphorylation of the third site.

Fig. 1. Chemical reactions for $m = 2$ and $n = 1$.

$$P + X[u,u,u] \underset{k_d}{\overset{2k}{\rightleftharpoons}} X[p,u,u]$$
$$P + X[u,u,p] \underset{k_d}{\overset{2k}{\rightleftharpoons}} X[p,u,p]$$

(a) Phosphorylation and dephosphorylation of the first site.

$$P + X[p,u,u] \underset{2k_d}{\overset{k}{\rightleftharpoons}} X[p,p,u]$$
$$P + X[p,u,p] \underset{2k_d}{\overset{k}{\rightleftharpoons}} X[p,p,p]$$

(b) Phosphorylation and dephosphorylation of the second site.

$$P + X[u,u,u] \underset{k_d}{\overset{k_{1,0}}{\rightleftharpoons}} X[u,u,p]$$
$$P + X[p,u,u] \underset{k_d}{\overset{k_{1,1}}{\rightleftharpoons}} X[p,u,p]$$
$$P + X[p,p,u] \underset{k_d}{\overset{k_{1,2}}{\rightleftharpoons}} X[p,p,p]$$

(c) Phosphorylation and dephosphorylation of the third site.

Fig. 2. Simplified chemical reactions for $m = 2$ and $n = 1$.

j between 1 and n. Thus the set of reachable configurations for the protein X, is quotiented into $(m + 1)2^n$ $\sim$-equivalence classes. A simplified set of reactions can be proposed, by choosing a representative among each $\sim$-equivalence class. Indeed, up to updating reaction rate constants, we may assume that the sites $x_1, \dots, x_m$ are always phosphorylated in increasing order, and dephosphorylated in decreasing order. In Fig. 2, we give the set of so obtained simplified reactions (for $m = 2$ and $n = 1$). We can notice that whenever the sites x_1 and x_2 are both unphosphorylated, only the site x_1 can be phosphorylated (with a rate twice as big as in the initial reaction) and that whenever the sites x_1 and x_2 are both phosphorylated, only the site x_2 can be dephosphorylated (at a rate twice bigger than in the initial reaction). Such a simplified set of reactions can be used to compute a reduced stochastic semantics and a reduced differential semantics.

In this paper, we propose two formal frameworks so as to combine model reductions for stochastic and differential semantics, and we apply these frameworks for combining existing model reductions [10,7,11] with a model reduction based on the detection of symmetric sites. Both reduced semantics can be derived automatically, without explicitly computing neither the unreduced semantics, nor any intermediate semantics. For instance, the fragments-based model reduction that is proposed in [10,7] abstract away the correlation between the phosphorylation states of the sites y_j (for any integer j between 1 and n), because, this correlation is tested in no reaction. This yields $n2^{m+1}$ fragments for the protein X. Combined with sites symmetries, we would get only $2n(m+1)$ classes of fragments. The fragments-based approach achieves no reduction in the case of the stochastic semantics [11]: thus we can only reduce to $(m + 1)2^n$ classes of species (or fragments) in this case.

3 Stochastic semantics

In this section, we review the generic framework that has been proposed in [11] for reducing the stochastic semantics of weighted labeled transition systems. This reduction technique is based on the use of backward bisimulations [4] so as to prove statistical invariants. We use these invariants to lump [3] some states of the transition system together, this is a weak lumping which is sound only if the statistical invariants are satisfied by the initial distribution of states. We extend this framework with a commutative operator to combine abstractions. We show that this operator is a pushout, and that abstraction composition distributes over it.

3.1 Weighted labeled transition systems

We first introduce the notion of weighted labeled transition system.

Definition 3.1 A *weighted labeled transition system* (WLTS) is a tuple $(\mathcal{Q}, \mathcal{L}, \rightarrow, w, \mathcal{I}, \pi_0)$ where: (i) $\mathcal{Q}$ is a set of states, (ii) $\mathcal{L}$ is a set of transition labels, (iii) $\rightarrow \subseteq \mathcal{Q} \times \mathcal{L} \times \mathcal{Q}$ is a relation, (iv) w is a mapping between $\mathcal{Q} \times \mathcal{L}$ and $\mathbb{R}^+$, (v) $\mathcal{I} \subseteq \mathcal{Q}$ is a finite subset of states, and (vi) $\pi_0 : \mathcal{I} \rightarrow [0,1]$ is a discrete probability distribution.

Let us now consider $(\mathcal{Q}, \mathcal{L}, \rightarrow, w, \mathcal{I}, \pi_0)$ a WLTS. A state $q \in \mathcal{I}$ is called an initial state. Moreover, the probability that the system is in the state $q \in \mathcal{I}$ at time $t = 0$ is equal to $\pi_0(q)$. An element $(q, \lambda, q') \in \rightarrow$ denotes a transition from state q to state q'; the symbol λ is the label of the transition. We denote by $q \xrightarrow{\lambda} q'$ the fact that the tuple (q, λ, q') belongs to $\rightarrow$. In the following we will assume that a label fully identifies a transition step. That is to say that given a label $\lambda \in \mathcal{L}$, and four states $q_1, q_2, q_1', q_2' \in \mathcal{Q}$ such that $q_1 \xrightarrow{\lambda} q_1'$ and $q_2 \xrightarrow{\lambda} q_2'$, then we have $q_1 = q_2$ and $q_1' = q_2'$. We denote by $\mathcal{L}(q) \subseteq \mathcal{L}$ the set of labels for which there exists $q' \in \mathcal{Q}$ such that $q \xrightarrow{\lambda} q'$. Moreover, we also assume that the system is finitely branching, that is to say that given a state q, the set $\mathcal{L}(q)$ is finite. The function w associates each transition $q \xrightarrow{\lambda} q'$ to its weight (or rate) $w(q, \lambda) \in \mathbb{R}^+ \backslash \{0\}$.

Now we define a continuous-time semantics for WLTS. This semantics is defined as a probability density distribution of the traces with k steps, for any natural number $k \in \mathbb{N}$. First we give the definition of finite traces as follows:

Definition 3.2 A *finite trace* is given by an initial state $q_0 \in \mathcal{I}$ and a finite sequence $(\lambda_i, t_i, q_i)_{1 \leqslant i \leqslant k} \in (\mathcal{L} \times \mathbb{R}^+ \times \mathcal{Q})^k$ of triples such that: for any integer i such that $1 \leqslant i \leqslant k$, we have $q_{i-1} \xrightarrow{\lambda_i} q_i$.

Such a trace is denoted as: $q_0 \xrightarrow{\lambda_1, t_1} q_1 \cdots q_{k-1} \xrightarrow{\lambda_k, t_k} q_k$. Whenever $i > 1$, the non negative real number t_i denotes the amount of time between the i-th transition of the system and the previous one, moreover t_1 denotes the amount of time between $t = 0$ and the first transition. Moreover, the number of transitions (here k) is called the size of the trace.

Now we define the probability density distribution of the traces of size k, for any natural number $k \in \mathbb{N}$. For that purpose, we introduce $\mathbb{IR}^+$ as the set of intervals of positive real numbers.

Definition 3.3 Given a natural number $k \in \mathbb{N}$, an initial state $q_0 \in \mathcal{I}$ and a sequence $(\lambda_i, I_i, q_i)_{1 \leqslant i \leqslant k} \in (\mathcal{L} \times \mathbb{IR}^+ \times \mathcal{Q})^k$ of tuples, the set of traces that is defined as follows:

$$q_0 \xrightarrow{\lambda_1, I_1} q_1 \ldots q_{k-1} \xrightarrow{\lambda_k, I_k} q_k := \left\{ q_0 \xrightarrow{\lambda_1, t_1} q_1 \ldots q_{k-1} \xrightarrow{\lambda_k, t_k} q_k \;\middle|\; t_i \in I_i \right\},$$

is called a *cylinder set of traces.*

We denote by $\mathcal{T}_{\mathbb{IR}^+}$ the set of cylinder sets τ of traces.

Now we define the probability of a cylinder set of traces.

Definition 3.4 Let k be a natural number in $\mathbb{N}$. The probability that a trace of size k lies in the following cylinder set of traces: $q_0 \xrightarrow{\lambda_1, I_1} q_1 \ldots q_{k-1} \xrightarrow{\lambda_k, I_k} q_k$, is given by the following expression:

$$\pi_0(q_0) \prod\nolimits_i \left(\frac{w(q_{i-1}, \lambda_i) \left(e^{-\mathrm{a}(q_{i-1}) \mathrm{inf}(I_i)} - e^{-\mathrm{a}(q_{i-1}) \mathrm{sup}(I_i)} \right)}{\mathrm{a}(q_{i-1})} \;\middle|\; 1 \leqslant i \leqslant k \right),$$

where for any state q, $\mathrm{a}(q)$ is the activity of the system at state q which is defined as: $\mathrm{a}(q) := \sum_\lambda (w(q, \lambda) \mid \lambda \in \mathcal{L}(q))$.

We notice that initial states are selected according to the distribution π_0. Moreover, whenever the system is in the state q, the next state is selected by computing the transition labeled with $\lambda \in \mathcal{L}(q)$ with probability $\frac{w(q,\lambda)}{\mathrm{a}(q)}$ and the waiting time until a next reaction happens is chosen according to an exponential probability distribution with the parameter that is equal to the activity $\mathrm{a}(q)$ of the system.

3.2 Abstraction

The description of a system can be less or more fine grained, which leads to the notion of abstraction between WLTSs:

Definition 3.5 An abstraction between two WLTSs $\mathcal{S} := (\mathcal{Q}, \mathcal{L}, \rightarrow, w, \mathcal{I}, \pi_0)$ and $\mathcal{S}^\sharp := (\mathcal{Q}^\sharp, \mathcal{L}^\sharp, \rightsquigarrow, w^\sharp, \mathcal{I}^\sharp, \pi_0^\sharp)$ is a tuple $(\mathcal{S}, \mathcal{S}^\sharp, \beta^\mathcal{L}, \beta^\mathcal{Q}, \gamma^\mathcal{Q})$ where $\beta^\mathcal{L} : \mathcal{L} \rightarrow \mathcal{L}^\sharp$, $\beta^\mathcal{Q} : \mathcal{Q} \rightarrow \mathcal{Q}^\sharp$, and $\gamma^\mathcal{Q} : \mathcal{Q} \rightarrow [0, 1]$ are three mappings such that the following six properties are satisfied: (i) Both mappings $\beta^\mathcal{L}$ and $\beta^\mathcal{Q}$ are onto. (ii) For any *abstract* state $q^\sharp \in \mathcal{Q}^\sharp$, the restriction of the mapping $\gamma^\mathcal{Q}$ to the *concrete* state $q \in \mathcal{Q}$ such that $\beta^\mathcal{Q}(q) = q^\sharp$ is a finite probability distribution. (iii) For any two concrete states $q_1, q_2 \in \mathcal{Q}$ such that $\beta^\mathcal{Q}(q_1) = \beta^\mathcal{Q}(q_2)$, we have $\mathrm{a}(q_1) = \mathrm{a}(q_2)$. (iv) For any concrete state $q_1 \in \mathcal{Q}$, we have $q_1 \in \mathcal{I}$ if, and only if, $\beta^\mathcal{Q}(q_1) \in \mathcal{I}^\sharp$. (v) We have $q^\sharp \overset{\lambda^\sharp}{\rightsquigarrow} q^{\sharp\prime}$, if and only if there exists a transition $q \xrightarrow{\lambda} q'$ such that $\beta^\mathcal{Q}(q) = q^\sharp$, $\beta^\mathcal{Q}(q') = q^{\sharp\prime}$, $\beta^\mathcal{L}(\lambda) = \lambda^\sharp$. (vi) For any concrete state $q \in \mathcal{I}$, we

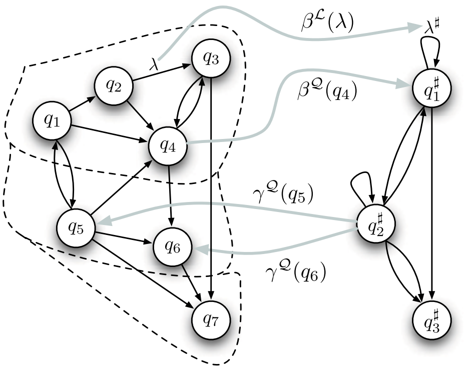
$\beta^{\mathcal{L}}(\lambda)$
$\lambda^\sharp$
λ
q_1
q_2
q_3
q_4
q_5
q_6
q_7
$\beta^{\mathcal{Q}}(q_4)$
$q_1^\sharp$
$\gamma^{\mathcal{Q}}(q_5)$
$q_2^\sharp$
$\gamma^{\mathcal{Q}}(q_6)$
$q_3^\sharp$

Fig. 3. An abstraction between two transition systems.

have: $\pi_0(q) = \gamma^{\mathcal{Q}}(q)\pi_0^\sharp(\beta^{\mathcal{Q}}(q))$. (vii) For any abstract transition $q^\sharp \overset{\lambda^\sharp}{\rightsquigarrow} q^{\sharp\prime}$ and any concrete state $q^\star$ such that $\beta^{\mathcal{Q}}(q^\star) = q^{\sharp\prime}$, we have:

$$\sum\nolimits_{q,\lambda}\left(\gamma^{\mathcal{Q}}(q)w(q,\lambda)\;\middle|\;\begin{array}{l} q \in \mathcal{Q}, \lambda \in \mathcal{L} \text{ s.t. } q \xrightarrow{\lambda} q^\star, \\ \beta^{\mathcal{Q}}(q) = q^\sharp, \beta^{\mathcal{L}}(\lambda) = \lambda^\sharp \end{array}\right) = \gamma^{\mathcal{Q}}(q^\star)w^\sharp(q^\sharp, \lambda^\sharp).$$

In this definition, the system $\mathcal{S}$ is called the concrete system, whereas the system $\mathcal{S}^\sharp$ is called the abstract system. Moreover, we often say that the tuple $\mathcal{A} := (\mathcal{S}, \mathcal{S}^\sharp, \beta^{\mathcal{L}}, \beta^{\mathcal{Q}}, \gamma^{\mathcal{Q}})$ is an abstraction between the concrete system $\mathcal{S}$ and the abstract system $\mathcal{S}^\sharp$ (eg see Fig. 3), and we write $\mathrm{pre}(\mathcal{A}) := \mathcal{S}$ and $\mathrm{post}(\mathcal{A}) := \mathcal{S}^\sharp$. The mappings $\beta^{\mathcal{Q}}$ and $\gamma^{\mathcal{Q}}$ relate concrete and abstract states, whereas the mapping $\beta^{\mathcal{L}}$ relates concrete and abstract labels. The first property ensures that any abstract transition label (resp. any abstract state) is actually the abstraction of a concrete transition label (resp. a concrete state). The second property entails that an abstract state $q^\sharp$ denotes a finite set of concrete states $\{q \mid \beta^{\mathcal{Q}}(q) = q^\sharp\}$, moreover the restriction of the mapping $\gamma^{\mathcal{Q}}$ to the states q' such that $\beta^{\mathcal{Q}}(q') = \beta^{\mathcal{Q}}(q)$ denotes the conditional probability that the system is in the state q knowing that it is in a state q' such that $\beta^{\mathcal{Q}}(q') = \beta^{\mathcal{Q}}(q)$. The fourth property ensures that when two concrete states have the same abstraction, then the first one is an initial state if, and only if the second one is initial. In such a case, their abstraction is an abstract initial state. The fifth property ensures that the set of abstract transitions is obtained by applying the abstraction function $\beta^{\mathcal{Q}}$ to initial and final states and the abstraction function $\beta^{\mathcal{L}}$ to the label of each concrete transition. The third and the fifth properties state that the abstraction is indeed a backward bisimulation [4].

Let us consider an abstraction $(\mathcal{S}, \mathcal{S}^\sharp, \beta^{\mathcal{L}}, \beta^{\mathcal{Q}}, \gamma^{\mathcal{Q}})$ between a concrete system $\mathcal{S} := (\mathcal{Q}, \mathcal{L}, \rightarrow, w, \mathcal{I}, \pi_0)$ and an abstract system $\mathcal{S}^\sharp := (\mathcal{Q}^\sharp, \mathcal{L}^\sharp, \rightsquigarrow, w^\sharp, \mathcal{I}^\sharp, \pi_0^\sharp)$. We want to state the correspondence between trace density distribution in these two systems. For that purpose, we introduce the function $\beta^{\mathcal{T}}$ which maps any cylinder set $q_0 \xrightarrow{\lambda_1, I_1} q_1 \cdots q_{k-1} \xrightarrow{\lambda_k, I_k} q_k$ of concrete traces to the cylinder set $\beta^{\mathcal{Q}}(q_0) \overset{\beta^{\mathcal{L}}(\lambda_1), I_1}{\rightsquigarrow} \beta^{\mathcal{Q}}(q_1) \cdots \beta^{\mathcal{Q}}(q_{k-1}) \overset{\beta^{\mathcal{L}}(\lambda_k), I_k}{\rightsquigarrow} \beta^{\mathcal{Q}}(q_k)$ of abstract traces. Thus, $\beta^{\mathcal{T}}$ applies $\beta^{\mathcal{Q}}$ to each state and $\beta^{\mathcal{L}}$ to each transition label, whereas time intervals are all preserved. Given a finite union X of cylinder sets of concrete traces, we denote by $P_c(q \mid X)$ the conditional probability that we are in a state q knowing that we have computed a trace in the set of cylinder sets X.

Theorem 3.6 ([11]) *For any cylinder set* $\tau^\sharp := q_0^\sharp \overset{\lambda_1^\sharp, I_1}{\rightsquigarrow} q_1^\sharp \cdots q_{k-1}^\sharp \overset{\lambda_k^\sharp, I_k}{\rightsquigarrow} q_k^\sharp$ *of abstract traces of any length k and any sequence of steps, the following holds.*

(i) **soundness.** *the probability $P(\tau^\sharp)$ of the cylinder set $\tau^\sharp$ of abstract traces is equal to the sum of the probabilities of the cylinder sets τ of traces such that $\beta^{\mathcal{T}}(\tau) = \tau^\sharp$.*
That is to say: $P(\tau^\sharp) = \sum_\tau (P(\tau) \mid \tau \in \mathcal{T}_{\mathbb{IR}^+} \textit{ s.t. } \beta^{\mathcal{T}}(\tau) = \tau^\sharp)$.

(ii) **state completeness.** *For any concrete state $q_k^\star \in \mathcal{Q}$ such that $\beta^{\mathcal{Q}}(q_k^\star) = q_k^\sharp$, the conditional probability that the system is in the state $q_k^\star$ knowing that we have computed a trace in a cylinder set $\tau \in \mathcal{T}_{\mathbb{IR}^+}$ such that $\beta^{\mathcal{T}}(\tau) = \tau^\sharp$ is, when it is defined, equal to $\gamma^{\mathcal{Q}}(q_k^\star)$.*
That is to say: $P_c(q_k^\star \mid \{\tau \in \mathcal{T}_{\mathbb{IR}^+} \textit{ s.t. } \beta^{\mathcal{T}}(\tau) = \tau^\sharp\}) = \gamma^{\mathcal{Q}}(q_k^\star)$.

The soundness of the abstraction is stated in a classic way: it ensures that we can perform the computation of the probability that an abstract trace lies in a given a cylinder set $\tau^\sharp$ of abstract traces (of size k) either in the abstract, or in the concrete by summing the probabilities that a concrete trace lies in the cylinder set $\tau \in \mathcal{T}_{\mathbb{IR}^+}$ of traces for any $\tau \in \mathcal{T}_{\mathbb{IR}^+}$ such that $\beta^{\mathcal{T}}(\tau) = \tau^\sharp$. Both ways will give the same result. The completeness states that, even if the computation was made in the abstract, we can recover the distribution over the concrete states that are abstracted by the final abstract state. Indeed, the distribution of the final states is given by the restriction of $\gamma^{\mathcal{Q}}$ to the states q such that $\beta^{\mathcal{Q}}(q) = q_k^\sharp$.

It follows from Def. 3.5 and Theorem 3.6.(ii) that two abstractions $\mathcal{A}_1 := (\mathcal{S}_1, \mathcal{S}_1^\sharp, \beta_1^{\mathcal{L}}, \beta_1^{\mathcal{Q}}, \gamma_1^{\mathcal{Q}})$ and $\mathcal{A}_2 := (\mathcal{S}_2, \mathcal{S}_2^\sharp, \beta_2^{\mathcal{L}}, \beta_2^{\mathcal{Q}}, \gamma_2^{\mathcal{Q}})$ such that $\mathcal{S}_1 = \mathcal{S}_2$, $\beta_1^{\mathcal{Q}} = \beta_2^{\mathcal{Q}}$, $\beta_1^{\mathcal{L}} = \beta_2^{\mathcal{L}}$, are equal providing that for any (abstract) state $q^\sharp \in \mathcal{Q}^\sharp$, there exists a reachable concrete state $q \in \mathcal{V} \to \mathbb{R}$ such that $\beta_1^{\mathcal{Q}}(q) = q^\sharp$. In the following, we assume without any loss of generality that any state in a WLTS is reachable.

3.3 Composition and factorization

In this section, we remind two binary operators over abstractions, namely, the composition and the factorization of abstractions, we will introduce a new one (called product) in Sect. 3.4.

Two abstractions between the systems $\mathcal{S}^\flat$ and $\mathcal{S}$, and between the systems $\mathcal{S}$ and $\mathcal{S}^\sharp$, can be composed in order to form an abstraction between the systems $\mathcal{S}^\flat$ and $\mathcal{S}^\sharp$. More precisely, let us consider six mappings $\beta_1^{\mathcal{L}}$, $\beta_1^{\mathcal{Q}}$, $\gamma_1^{\mathcal{Q}}$, $\beta_2^{\mathcal{L}}$, $\beta_2^{\mathcal{Q}}$, and $\gamma_2^{\mathcal{Q}}$ such that $\mathcal{A}_1 := (\mathcal{S}^\flat, \mathcal{S}, \beta_1^{\mathcal{L}}, \beta_1^{\mathcal{Q}}, \gamma_1^{\mathcal{Q}})$ and $\mathcal{A}_2 := (\mathcal{S}, \mathcal{S}^\sharp, \beta_2^{\mathcal{L}}, \beta_2^{\mathcal{Q}}, \gamma_2^{\mathcal{Q}})$ are two abstractions.

Proposition 3.7 (composition [11]) *The tuple* $\mathcal{A}_3 := (\mathcal{S}^\flat, \mathcal{S}^\sharp, \beta_3^{\mathcal{L}}, \beta_3^{\mathcal{Q}}, \gamma_3^{\mathcal{Q}})$ *where:* $\beta_3^{\mathcal{L}} := \beta_2^{\mathcal{L}} \circ \beta_1^{\mathcal{L}}$, $\beta_3^{\mathcal{Q}} := \beta_2^{\mathcal{Q}} \circ \beta_1^{\mathcal{Q}}$, *and* $\gamma_3^{\mathcal{Q}}(q^\flat) := \gamma_1^{\mathcal{Q}}(q^\flat)\gamma_2^{\mathcal{Q}}(\beta_1^{\mathcal{Q}}(q^\flat))$, *is also an abstraction.*

Moreover, the abstraction $\mathcal{A}_3$ is called the composition *of $\mathcal{A}_1$ and $\mathcal{A}_2$, which is denoted by $\mathcal{A}_3 = \mathcal{A}_2 \circ \mathcal{A}_1$.*

Definition 3.8 An abstraction $\mathcal{A} = (\mathcal{S}, \mathcal{S}^\sharp, \beta^{\mathcal{L}}, \beta^{\mathcal{Q}}, \gamma^{\mathcal{Q}})$ such that $\mathcal{S} = \mathcal{S}^\sharp$, and both functions $\beta^{\mathcal{L}}$ and $\beta^{\mathcal{Q}}$ are identity functions, is called *an identity abstraction*, and is denoted by $Id_{\mathcal{S}}$.

We notice that, for any abstraction $\mathcal{A}$, by definitions of abstraction composition and identity abstractions, we have $\mathcal{A} \circ Id_{\text{pre}(\mathcal{A})} = \mathcal{A}$ and $Id_{\text{post}(\mathcal{A})} \circ \mathcal{A} = \mathcal{A}$. We can use abstraction composition to define the notion of isomorphism.

Definition 3.9 An abstraction $\mathcal{A}_1$, for which there exists an abstraction $\mathcal{A}_2$ between post$(\mathcal{A}_1)$ and pre$(\mathcal{A}_1)$ such that both $\mathcal{A}_1 \circ \mathcal{A}_2 = Id_{\text{post}(\mathcal{A}_1)}$, and $\mathcal{A}_2 \circ \mathcal{A}_1 = Id_{\text{pre}(\mathcal{A}_1)}$ is called an isomorphism. In such a case, $\mathcal{A}_2$ is called the *inverse* of $\mathcal{A}_1$ and is denoted by $\mathcal{A}_1^{-1}$.

We now introduce a factorization operator as the converse of the composition operator. Provided some compatibility requirements, one can decompose an abstraction between the systems $\mathcal{S}^\flat$ and $\mathcal{S}^\sharp$, by providing an abstraction between the systems $\mathcal{S}^\flat$ and $\mathcal{S}$. More precisely, let us introduce three WLTSs $\mathcal{S}^\flat := (\mathcal{Q}^\flat, \mathcal{L}^\flat, \rightarrow_\flat, w^\flat, \mathcal{I}^\flat, \pi_0^\flat)$, $\mathcal{S} := (\mathcal{Q}, \mathcal{L}, \rightarrow, w, \mathcal{I}, \pi_0)$, and $\mathcal{S}^\sharp := (\mathcal{Q}^\sharp, \mathcal{L}^\sharp, \rightsquigarrow, w^\sharp, \mathcal{I}^\sharp, \pi_0^\sharp)$, and let us consider six mappings $\beta_1^{\mathcal{L}}$, $\beta_1^{\mathcal{Q}}$, $\gamma_1^{\mathcal{Q}}$, $\beta_2^{\mathcal{L}}$, $\beta_2^{\mathcal{Q}}$, and $\gamma_2^{\mathcal{Q}}$ such that $\mathcal{A}_1 := (\mathcal{S}^\flat, \mathcal{S}^\sharp, \beta_1^{\mathcal{L}}, \beta_1^{\mathcal{Q}}, \gamma_1^{\mathcal{Q}})$ and $\mathcal{A}_2 := (\mathcal{S}^\flat, \mathcal{S}, \beta_2^{\mathcal{L}}, \beta_2^{\mathcal{Q}}, \gamma_2^{\mathcal{Q}})$ are two abstractions such that (i) for any $q_1^\flat, q_2^\flat \in \mathcal{Q}^\flat$ such that $\beta_2^{\mathcal{Q}}(q_1^\flat) = \beta_2^{\mathcal{Q}}(q_2^\flat)$, we have: $\beta_1^{\mathcal{Q}}(q_1^\flat) = \beta_1^{\mathcal{Q}}(q_2^\flat)$; and (ii) for any $\lambda_1^\flat, \lambda_2^\flat \in \mathcal{Q}^\flat$ such that $\beta_2^{\mathcal{L}}(\lambda_1^\flat) = \beta_2^{\mathcal{L}}(\lambda_2^\flat)$, we have: $\beta_1^{\mathcal{L}}(\lambda_1^\flat) = \beta_1^{\mathcal{L}}(\lambda_2^\flat)$.

Proposition 3.10 (factorization [11]) *The tuple $\mathcal{A}_3 := (\mathcal{S}, \mathcal{S}^\sharp, \beta_3^{\mathcal{L}}, \beta_3^{\mathcal{Q}}, \gamma_3^{\mathcal{Q}})$ where: $\beta_3^{\mathcal{L}}(\lambda)$ is defined as $\beta_1^{\mathcal{L}}(\lambda^\flat)$ for a given $\lambda^\flat \in \mathcal{L}^\flat$ such that $\beta_2^{\mathcal{L}}(\lambda^\flat) = \lambda$; $\beta_3^{\mathcal{Q}}(q)$ is defined as $\beta_1^{\mathcal{Q}}(q^\flat)$ for a given $q^\flat \in \mathcal{Q}^\flat$ such that $\beta_2^{\mathcal{Q}}(q^\flat) = q$; and $\gamma_3^{\mathcal{Q}}(q)$ is defined by: $\gamma_3^{\mathcal{Q}}(q) = \sum_{q^\flat} \left(\gamma_1^{\mathcal{Q}}(q^\flat) \mid q^\flat \in \mathcal{Q}^\flat \text{ s.t. } \beta_2^{\mathcal{Q}}(q^\flat) = q\right)$, is also an abstraction.*

Moreover, the abstraction $\mathcal{A}_3$ is called the factorization *of $\mathcal{A}_1$ by $\mathcal{A}_2$, which is denoted by $\mathcal{A}_1 \backslash \mathcal{A}_2$.*

Intuitively, the abstraction $\mathcal{A}_1$ can be factorized by the abstraction $\mathcal{A}_2$ only if the abstraction $\mathcal{A}_1$ is coarser than $\mathcal{A}_2$, that is to say that each pair of states (or transition labels) which cannot be distinguished in the abstraction $\mathcal{A}_2$, cannot be distinguished in the abstraction $\mathcal{A}_1$ either.

As expected, abstraction composition is the inverse of abstraction factorization (and conversely), as stated by the following algebraic properties:

Proposition 3.11 (Algebraic identities [11]) *Let $\mathcal{A}_1, \mathcal{A}_2$ be two abstractions. The following properties are satisfied:*

(i) *if $\mathcal{A}_1 \backslash \mathcal{A}_2$ is well-defined, then $(\mathcal{A}_1 \backslash \mathcal{A}_2) \circ \mathcal{A}_2$ is well-defined and $(\mathcal{A}_1 \backslash \mathcal{A}_2) \circ \mathcal{A}_2 = \mathcal{A}_1$;*

(ii) *if $\mathcal{A}_2 \circ \mathcal{A}_1$ is well-defined, then $(\mathcal{A}_2 \circ \mathcal{A}_1) \backslash \mathcal{A}_1$ is well-defined and $(\mathcal{A}_2 \circ \mathcal{A}_1) \backslash \mathcal{A}_1 = \mathcal{A}_2$.*

Let us consider an abstraction $\mathcal{A}$. The factorization $\mathcal{A}\backslash\mathcal{A}$ is well-defined and is equal to $Id_{\mathrm{post}(\mathcal{A})}$ (by applying proposition 3.11.(ii) with $\mathcal{A}_1 := \mathcal{A}$ and $\mathcal{A}_2 := Id_{\mathrm{post}(\mathcal{A})}$). Moreover, the factorization $\mathcal{A}\backslash Id_{\mathrm{pre}(\mathcal{A})}$ is well-defined and is equal to $\mathcal{A}$ (by applying proposition 3.11.(ii) with $\mathcal{A}_1 := Id_{\mathrm{pre}(\mathcal{A})}$ and $\mathcal{A}_2 := \mathcal{A}$).

3.4 Product of two abstractions

Now we introduce a new operator to combine two abstractions. We call a *span* two abstractions $\mathcal{A}_1$ and $\mathcal{A}_2$ from the same WLTS $\mathcal{S}$. Such a span is denoted by $\overset{\smile}{\mathcal{A}_1\mathcal{S}\mathcal{A}_2}$, or also $\overset{\smile}{\mathcal{A}_1\mathcal{A}_2}$. Dually, we call a *cospan* two abstractions $\mathcal{A}_1$ and $\mathcal{A}_2$ going into the same WLTS $\mathcal{S}$, such a cospan in denoted by $\overset{\frown}{\mathcal{A}_1\mathcal{S}\mathcal{A}_2}$ or $\overset{\frown}{\mathcal{A}_1\mathcal{A}_2}$.

Proposition 3.12 (pushout) *For any span $\overset{\smile}{\mathcal{A}_1\mathcal{A}_2}$, there exists a universal cospan $\overset{\frown}{\mathcal{A}_3\mathcal{A}_4}$ such that $\mathcal{A}_3 \circ \mathcal{A}_1 = \mathcal{A}_4 \circ \mathcal{A}_2$ and for any other cospan $\overset{\frown}{\mathcal{A}_5\mathcal{A}_6}$ such that $\mathcal{A}_5 \circ \mathcal{A}_1 = \mathcal{A}_6 \circ \mathcal{A}_2$, there exists a unique abstraction u such that $u \circ \mathcal{A}_3 = \mathcal{A}_5$ and $u \circ \mathcal{A}_4 = \mathcal{A}_6$ (eg see Fig 4).*

With these notations, the abstraction $\mathcal{A}_3 \circ \mathcal{A}_1$ is called a product between the abstractions $\mathcal{A}_1$ and $\mathcal{A}_2$ and the cospan $\overset{\frown}{\mathcal{A}_3\mathcal{A}_4}$ is called a pushout *of the span $\overset{\smile}{\mathcal{A}_1\mathcal{A}_2}$.*

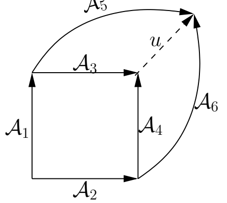

Fig. 4: *The pushout property.*

Intuitively, the product of two abstractions from the same WLTS, is the abstraction that is more abstract than these two abstractions, and which is the least abstract among all the abstractions which satisfy this property. Now let us sketch the proof of proposition 3.12.

Proof. Let $\overset{\smile}{\mathcal{A}_1\mathcal{S}\mathcal{A}_2}$ be a span. We denote $\mathcal{A}_1 := (\mathcal{S}, \mathcal{S}^{\sharp_1}, \beta_1^{\mathcal{L}}, \beta_1^{\mathcal{Q}}, \gamma_1^{\mathcal{Q}})$ and $\mathcal{A}_2 := (\mathcal{S}, \mathcal{S}^{\sharp_2}, \beta_2^{\mathcal{L}}, \beta_2^{\mathcal{Q}}, \gamma_2^{\mathcal{Q}})$. Moreover, we denote $\mathcal{S} := (\mathcal{Q}, \mathcal{L}, \rightarrow, w, \mathcal{I}, \pi_0)$, $\mathcal{S}^{\sharp_1} = (\mathcal{Q}^{\sharp_1}, \mathcal{L}^{\sharp_1}, \rightarrow_{\sharp_1}, w^{\sharp_1}, \mathcal{I}^{\sharp_1}, \pi_0^{\sharp_1})$, and $\mathcal{S}^{\sharp_2} := (\mathcal{Q}^{\sharp_2}, \mathcal{L}^{\sharp_2}, \rightarrow_{\sharp_2}, w^{\sharp_2}, \mathcal{I}^{\sharp_2}, \pi_0^{\sharp_2})$. We define the equivalence relation $\sim_{\mathcal{Q}}$ over $\mathcal{Q}$ as the smallest equivalence relation which satisfies $q \sim_{\mathcal{Q}} q'$ whenever $\beta_1^{\mathcal{Q}}(q) = \beta_1^{\mathcal{Q}}(q')$ or $\beta_2^{\mathcal{Q}}(q) = \beta_2^{\mathcal{Q}}(q')$. We also introduce the equivalence relation $\sim_{\mathcal{L}}$ over $\mathcal{L}$ as the smallest equivalence relation which satisfies $\lambda \sim_{\mathcal{L}} \lambda'$ whenever $\beta_1^{\mathcal{L}}(\lambda) = \beta_1^{\mathcal{L}}(\lambda')$ or $\beta_2^{\mathcal{L}}(\lambda) = \beta_2^{\mathcal{L}}(\lambda')$. We introduce the set $\mathcal{Q}^{\sharp_{1,2}}$ as the set of $\sim_{\mathcal{Q}}$-equivalence classes, and the set $\mathcal{L}^{\sharp_{1,2}}$ as the set of $\sim_{\mathcal{L}}$-equivalence classes. We denote by $\beta_{1,2}^{\mathcal{Q}}$ the function mapping any state $q \in \mathcal{Q}$ to its $\sim_{\mathcal{Q}}$-equivalence class, and by $\beta_{1,2}^{\mathcal{L}}$ the function mapping any transition label $\lambda \in \mathcal{L}$ to its $\sim_{\mathcal{L}}$-equivalence class.

Given a cylinder set τ of (concrete) traces of $\mathcal{S}$, we denote by $\beta_1^{\mathcal{T}}(\tau)$ (resp. $\beta_2^{\mathcal{T}}(\tau)$, resp. $\beta_{1,2}^{\mathcal{T}}(\tau)$), the cylinder set of traces that is obtained by replacing each state q with the abstract state $\beta_1^{\mathcal{Q}}(q)$ (resp. $\beta_2^{\mathcal{Q}}(q)$, resp. $\beta_{1,2}^{\mathcal{Q}}(q)$) and any transition label λ with the abstract transition label $\beta_1^{\mathcal{L}}(\lambda)$ (resp. $\beta_2^{\mathcal{L}}(\lambda)$, resp. $\beta_{1,2}^{\mathcal{L}}(\lambda)$). For any states $q, q' \in \mathcal{Q}$ such that $q \sim_{\mathcal{Q}} q'$, there exist an integer $n \in \mathbb{N}\backslash\{0\}$ and two sequences $q_1, \ldots, q_n \in \mathcal{Q}^n$ and $i_1, \ldots, i_{n-1} \in \{1,2\}^{n-1}$ such that: $q = q_1$, $q' = q_n$, and $\beta_{i_k}^{\mathcal{Q}}(q_k) = \beta_{i_k}^{\mathcal{Q}}(q_{k+1})$ for any integer k between 1 and $n-1$. Let us show that the quantity:

$$\prod\nolimits_k \left(\frac{\gamma_{i_k}^{\mathcal{Q}}(q_k)}{\gamma_{i_k}^{\mathcal{Q}}(q_{k+1})} \middle| 1 \leqslant k < n \right),$$

is independent from the choice of the two sequences $(i_k)_k$ and $(q_k)_k$.

Let k be an integer between 1 and $n-1$, and $\tau_\star$ be a cylinder set of concrete traces ending in state q (such a trace exists since we have assumed that all states were reachable). Let τ be a cylinder set of traces such that $\beta_{1,2}^{\mathcal{T}}(\tau) = \beta_{1,2}^{\mathcal{T}}(\tau_\star)$. We know by Theorem 3.6.(ii) that: $\gamma_{i_k}^{\mathcal{Q}}(q_k)P_c(q_{k+1} \mid \beta_{i_k}^{\mathcal{Q}}(\tau)) = \gamma_{i_k}^{\mathcal{Q}}(q_{k+1})P_c(q_k \mid \beta_{i_k}^{\mathcal{Q}}(\tau))$. By using $\beta_{i_k}^{\mathcal{Q}}$ to partition the set $\beta_{1,2}^{\mathcal{T}}(\tau_\star)$, it follows by summation that: $\gamma_{i_k}^{\mathcal{Q}}(q_k)P_c(q_{k+1} \mid \beta_{1,2}^{\mathcal{Q}}(\tau_\star)) = \gamma_{i_k}^{\mathcal{Q}}(q_{k+1})P_c(q_k \mid \beta_{1,2}^{\mathcal{Q}}(\tau_\star))$. Thus, after simplification, we get that:

$$\prod\nolimits_k \left(\frac{\gamma_{i_k}^{\mathcal{Q}}(q_k)}{\gamma_{i_k}^{\mathcal{Q}}(q_{k+1})} \middle| 1 \leqslant k < n \right) = \frac{P_c(q \mid \beta_{1,2}^{\mathcal{Q}}(\tau_\star))}{P_c(q' \mid \beta_{1,2}^{\mathcal{Q}}(\tau_\star))}.$$

Thus, this quantity does not depend on the choice of the two sequences $(q_k)_k$ and $(i_k)_k$. We call this quantity, the probability ratio between q and q' and we denote it by $\mathrm{pratio}(q, q')$.

Now we consider the mapping $\gamma_{1,2}^{\mathcal{Q}} : \mathcal{Q} \to [0, 1]$ such that: for any $q \in \mathcal{Q}$, $\sum_{q'}(\gamma_{1,2}^{\mathcal{Q}}(q') \mid q' \sim_{\mathcal{Q}} q) = 1$ and for any $q, q' \in \mathcal{Q}$, $\gamma_{1,2}^{\mathcal{Q}}(q) = \mathrm{pratio}(q, q')\gamma_{1,2}^{\mathcal{Q}}(q')$. By definition of $\gamma_{1,2}^{\mathcal{Q}}$, there exists a unique tuple $(\rightsquigarrow_{1,2}, w^{\sharp_{1,2}}, \mathcal{I}^{\sharp_{1,2}}, \pi_0^{\sharp_{1,2}})$ such that $\mathcal{A}_{1,2} := (\mathcal{S}, (\mathcal{Q}^{\sharp_{1,2}}, \mathcal{L}^{\sharp_{1,2}}, \rightsquigarrow_{1,2}, w^{\sharp_{1,2}}, \mathcal{I}^{\sharp_{1,2}}, \pi_0^{\sharp_{1,2}}), \beta_{1,2}^{\mathcal{L}}, \beta_{1,2}^{\mathcal{Q}}, \gamma_{1,2}^{\mathcal{Q}})$ is an abstraction (see Def. 3.5). By definition of $\beta_{1,2}^{\mathcal{Q}}$, for any concrete states $q, q' \in \mathcal{Q}$, we have $\beta_{1,2}^{\mathcal{Q}}(q) = \beta_{1,2}^{\mathcal{Q}}(q')$ whenever $\beta_1^{\mathcal{Q}}(q) = \beta_1^{\mathcal{Q}}(q')$. The same way, for any concrete transition labels $\lambda, \lambda' \in \mathcal{L}$, we have $\beta_{1,2}^{\mathcal{L}}(\lambda) = \beta_{1,2}^{\mathcal{L}}(\lambda')$ whenever $\beta_1^{\mathcal{L}}(\lambda) = \beta_1^{\mathcal{L}}(\lambda')$. By proposition 3.10, $\mathcal{A}_{1,2}$ can be factorized by $\mathcal{A}_1$ and we define $\mathcal{A}_3 := \mathcal{A}_{1,2} \backslash \mathcal{A}_1$, the same way, we can define $\mathcal{A}_4 := \mathcal{A}_{1,2} \backslash \mathcal{A}_2$.

Let us now consider two other abstractions $\mathcal{A}_5$ and $\mathcal{A}_6$ such that $\mathcal{A}_5 \circ \mathcal{A}_1 = \mathcal{A}_6 \circ \mathcal{A}_2$. We want to show that there exists a unique abstraction u such that $\mathcal{A}_5 = u \circ \mathcal{A}_3$ and $\mathcal{A}_6 = u \circ \mathcal{A}_4$. Let us first prove that $\mathcal{A}_5$ can be factorized by $\mathcal{A}_3$. We denote $\mathcal{A}_3 := (\mathcal{S}_1, \mathcal{S}_{1,2}, \beta_3^{\mathcal{L}}, \beta_3^{\mathcal{Q}}, \gamma_3^{\mathcal{Q}})$, $\mathcal{A}_5 := (\mathcal{S}_1, \mathcal{S}'_{1,2}, \beta_5^{\mathcal{L}}, \beta_5^{\mathcal{Q}}, \gamma_5^{\mathcal{Q}})$, and $\mathcal{A}_6 := (\mathcal{S}_2, \mathcal{S}'_{1,2}, \beta_6^{\mathcal{L}}, \beta_6^{\mathcal{Q}}, \gamma_6^{\mathcal{Q}})$. Let $q^{\sharp_1}, q^{\sharp_1\prime} \in \mathcal{Q}^{\sharp_1}$ such that $\beta_3^{\mathcal{Q}}(q^{\sharp_1}) = \beta_3^{\mathcal{Q}}(q^{\sharp_1\prime})$. Let us consider $q, q' \in \mathcal{Q}$ such that $\beta_1^{\mathcal{Q}}(q) = q^{\sharp_1}$ and $\beta_1^{\mathcal{Q}}(q') = q^{\sharp_1\prime}$. By definition of $\beta_{1,2}^{\mathcal{Q}}$, we have: $\beta_{1,2}^{\mathcal{Q}}(q) = \beta_{1,2}^{\mathcal{Q}}(q')$. It follows that there exist an integer $n > 0$ and two sequences $q_1, \ldots, q_n \in \mathcal{Q}^n$ and $i_1, \ldots, i_{n-1} \in \{1, 2\}^{n-1}$ such that: $q = q_1$, $q' = q_n$, and for any integer k between 1 and $n-1$, we have $\beta_{i_k}^{\mathcal{Q}}(q_k) = \beta_{i_k}^{\mathcal{Q}}(q_{k+1})$. Thus $\beta_5^{\mathcal{Q}}(\beta_1^{\mathcal{Q}}(q_k)) = \beta_{4+i_k}^{\mathcal{Q}}(\beta_{i_k}^{\mathcal{Q}}(q_k)) = \beta_{4+i_k}^{\mathcal{Q}}(\beta_{i_k}^{\mathcal{Q}}(q'_k)) = \beta_5^{\mathcal{Q}}(\beta_1^{\mathcal{Q}}(q'_k))$. It follows that $\beta_5^{\mathcal{Q}}(\beta_1^{\mathcal{Q}}(q)) = \beta_5^{\mathcal{Q}}(\beta_1^{\mathcal{Q}}(q'))$. Thus, $\beta_5^{\mathcal{Q}}(q^{\sharp_1}) = \beta_5^{\mathcal{Q}}(q^{\sharp_1\prime})$. The same way, for any $\lambda^{\sharp_1}, \lambda^{\sharp_1\prime} \in \mathcal{L}^{\sharp_1}$ such that $\beta_3^{\mathcal{L}}(\lambda^{\sharp_1}) = \beta_3^{\mathcal{L}}(\lambda^{\sharp_1\prime})$, we have: $\beta_5^{\mathcal{L}}(\lambda^{\sharp_1}) = \beta_5^{\mathcal{L}}(\lambda^{\sharp_1\prime})$. By proposition 3.10, it follows that $\mathcal{A}_5$ can be factorized by $\mathcal{A}_3$. The same way, $\mathcal{A}_6$ can be factorized by $\mathcal{A}_4$. By proposition 3.11.(ii), this factorization is unique, it follows that $\mathcal{A}_5 \backslash \mathcal{A}_3 = \mathcal{A}_6 \backslash \mathcal{A}_4$, so we can take $u := \mathcal{A}_5 \backslash \mathcal{A}_3$. By proposition 3.11.(i), we have

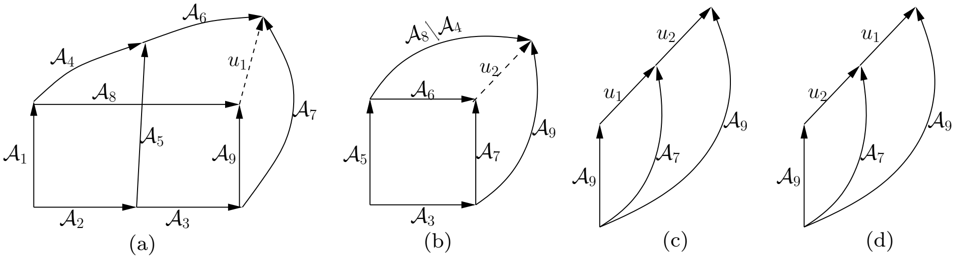

Fig. 5. Proof that composition distributes over pushout.

$\mathcal{A}_5 = u \circ \mathcal{A}_3$ and $\mathcal{A}_6 = u \circ \mathcal{A}_4$.

Moreover, by 3.11.(ii), there is at most one abstraction $u := \mathcal{A}_5\backslash\mathcal{A}_3$ such that $u \circ \mathcal{A}_3 = \mathcal{A}_6$, which ensures the uniqueness of the abstraction u of Fig. 4. □

Products and pushouts are defined up to isomorphisms. We call the ones that we have used in the proof *canonical*. We notice that this product operator is not the dual of the classical reduced product [6] of abstract interpretation.

3.5 Distributivity

Now we show that the pushout distributes over abstraction composition. We consider three abstractions $\mathcal{A}_1$,$\mathcal{A}_2$,$\mathcal{A}_3$ such that the composition $\mathcal{A}_3 \circ \mathcal{A}_2$ is well-defined and $\mathrm{pre}(\mathcal{A}_1) = \mathrm{pre}(\mathcal{A}_2)$. We introduce the five abstractions $\mathcal{A}_4$,$\mathcal{A}_5$,$\mathcal{A}_6$,$\mathcal{A}_7$,$\mathcal{A}_8$, and $\mathcal{A}_9$ such that: (i)$\overparen{\mathcal{A}_4\mathcal{A}_5}$ is a pushout of $\underparen{\mathcal{A}_1\mathcal{A}_2}$; (ii) $\overparen{\mathcal{A}_6\mathcal{A}_7}$ is a pushout of $\underparen{\mathcal{A}_5\mathcal{A}_3}$; (iii) $\overparen{\mathcal{A}_8\mathcal{A}_9}$ is a pushout of $\mathcal{A}_1\underparen{(\mathcal{A}_3 \circ \mathcal{A}_2)}$. The following proposition is satisfied:

Proposition 3.13 *There exists an isomorphism u, which satisfies: (i) $u \circ \mathcal{A}_8 = \mathcal{A}_6 \circ \mathcal{A}_4$, and (ii) $u^{-1} \circ \mathcal{A}_6 \circ \mathcal{A}_4 = \mathcal{A}_9 \circ \mathcal{A}_3 \circ \mathcal{A}_2$.*

Before proving proposition 3.13, we show the following lemma:

Lemma 3.14 *The abstraction $\mathcal{A}_8$ can be factorized by the abstraction $\mathcal{A}_4$.*

Proof. [Lemma 3.14] We assume, without any loss of generality, that the pushouts that we have chosen are all canonical. We denote by $\sim_{\mathcal{Q}}$ the equivalence relation over concrete states which is used to define the canonical pushout of $\underparen{\mathcal{A}_1\mathcal{A}_2}$ and by $\sim'_{\mathcal{Q}}$ the equivalence relation over the concrete states which is used to define the canonical pushout of $\mathcal{A}_1\underparen{(\mathcal{A}_3 \circ \mathcal{A}_2)}$. We denote by $\beta_1^{\mathcal{Q}}$, $\beta_4^{\mathcal{Q}}$ and $\beta_8^{\mathcal{Q}}$ the abstraction function over states in the abstractions $\mathcal{A}_1, \mathcal{A}_4$, and $\mathcal{A}_8$. Let $q^\sharp, q^{\sharp\prime}$ be two elements in the image of $\beta_1^{\mathcal{Q}}$ such that $\beta_4^{\mathcal{Q}}(q^\sharp) = \beta_4^{\mathcal{Q}}(q^{\sharp\prime})$. Let us consider q and q' such that $\beta_1^{\mathcal{Q}}(q) = q^\sharp$ and $\beta_1^{\mathcal{Q}}(q') = q^{\sharp\prime}$. We have $q \sim_{\mathcal{Q}} q'$. It follows that there exist $n > 0$ and two sequences $q_1, \ldots, q_n \in \mathcal{Q}^n$ and $i_1, \ldots, i_{n-1} \in \{1, 2\}^{n-1}$ such that: $q = q_1$, $q' = q_n$, and for any integer k between 1 and $n - 1$, we have $\beta_{i_k}^{\mathcal{Q}}(q_k) = \beta_{i_k}^{\mathcal{Q}}(q_{k+1})$. For any k between 1 and $n - 1$ such that $i_k = 2$, we also have $\beta_3^{\mathcal{Q}}(\beta_2^{\mathcal{Q}}(q_k)) = \beta_3^{\mathcal{Q}}(\beta_2^{\mathcal{Q}}(q_{k+1}))$. So we have $q \sim'_{\mathcal{Q}} q'$. So $\beta_8^{\mathcal{Q}}(q^\sharp) = \beta_8^{\mathcal{Q}}(\beta_1^{\mathcal{Q}}(q)) = \beta_8^{\mathcal{Q}}(\beta_1^{\mathcal{Q}}(q')) = \beta_8^{\mathcal{Q}}(q^{\sharp\prime})$. The same reasoning can be used to show that, for any pair $\lambda^\sharp, \lambda^{\sharp\prime}$ of element in the image of

$\beta_1^{\mathcal{L}}$ such that $\beta_4^{\mathcal{L}}(\lambda^\sharp) = \beta_4^{\mathcal{L}}(\lambda^{\sharp\prime})$, we have $\beta_8^{\mathcal{L}}(\lambda^\sharp) = \beta_8^{\mathcal{L}}(\lambda^{\sharp\prime})$, where $\beta_4^{\mathcal{L}}$ and $\beta_8^{\mathcal{L}}$ are the abstraction functions for transition labels in $\mathcal{A}_4$ and $\mathcal{A}_8$.

So, by proposition 3.10, the factorization $\mathcal{A}_8\backslash\mathcal{A}_4$ is well-defined. □

We can now give the proof of proposition 3.13.

Proof. [Proposition 3.13] This proof is depicted in Fig. 5. Since $\widehat{\mathcal{A}_8\mathcal{A}_9}$ is a pushout of $\mathcal{A}_1\widecheck{(\mathcal{A}_3 \circ \mathcal{A}_2)}$, there exists a (unique) abstraction u_1 such that: $u_1 \circ \mathcal{A}_8 = \mathcal{A}_6 \circ \mathcal{A}_4$ and $u_1 \circ \mathcal{A}_9 = \mathcal{A}_7$ (see Fig. 5(a)). Moreover, $\widehat{\mathcal{A}_6\mathcal{A}_7}$ is a pushout of $\widecheck{\mathcal{A}_5\mathcal{A}_3}$, so there exists a (unique) abstraction u_2 such that: $u_2 \circ \mathcal{A}_6 = \mathcal{A}_8\backslash\mathcal{A}_4$ and $u_2 \circ \mathcal{A}_7 = \mathcal{A}_9$ (see Fig. 5(b)). It follows that $u_2 \circ u_1 \circ \mathcal{A}_9 = u_2 \circ \mathcal{A}_7 = \mathcal{A}_9$ (see Fig. 5(c)). By proposition 3.11.(ii), $u_2 \circ u_1 = \mathcal{A}_9\backslash\mathcal{A}_9$, then $u_2 \circ u_1 = Id_{\mathrm{post}(\mathcal{A}_9)}$. Moreover, $u_1 \circ u_2 \circ \mathcal{A}_7 = u_2 \circ \mathcal{A}_9 = \mathcal{A}_7$ (see Fig. 5(d)). Thus, $u_1 \circ u_2 = \mathcal{A}_7\backslash\mathcal{A}_7$, then $u_1 \circ u_2 = Id_{\mathrm{post}(\mathcal{A}_7)}$.

It follows that both u_1 and u_2 are isomorphisms. □

In Sect. 6.3, we will apply the product between abstractions of WLTSs, so as to refine the three stochastic semantics (individual-based where each instance of protein is identified, population-based where proteins indexes are abstracted away, which comes down to consider that the state of the system is a multi-set of reachable species, and fragment-based where species are cut into independent fragments, so that the state is abstracted into a multi-set of fragments) which have been proposed in [11] by taking into account symmetries between the sites of proteins. We obtain the hierarchy of semantics which is depicted in Fig. 7 on page 22. Interestingly, the notion of symmetry needs only to be defined on the most concrete of these three semantics. Moreover, distributivity ensures that this hierarchy commutes (up to isomorphisms).

4 Differential semantics

In this section, we first recall the generic framework which has been proposed in [7] for defining differential semantics of rules-based models and to abstract this semantics. Then we show how we use linear projections so as abstract further an existing model reduction.

4.1 Concrete semantics

Let $\mathcal{V}$ be a finite set of variables. Maps from $\mathcal{V}$ to $\mathbb{R}$ form a normed vector space for a given norm $||\cdot||$. A mapping $\rho \in \mathcal{V} \mapsto \mathbb{R}$ such that for any variable $v \in \mathcal{V}$, we have $\rho(v) \geqslant 0$ is called a state, and we simply write $\rho \geqslant 0$. Moreover, given another finite set $\mathcal{V}^\sharp$ of variable and $||\cdot||^\sharp$ a norm on $\mathcal{V}^\sharp \to \mathbb{R}$, we say that a map ϕ between $\mathcal{V} \to \mathbb{R}$ and $\mathcal{V}^\sharp \to \mathbb{R}$ is expansive, whenever there exists a real number $\alpha > 0$ such that for any state $\rho \in \mathcal{V} \to \mathbb{R}$, we have $||\rho|| \leqslant \alpha||\phi(\rho)||^\sharp$. Since $\mathcal{V}$ and $\mathcal{V}^\sharp$ are finite, the set of expansive maps between $\mathcal{V} \to \mathbb{R}$ and $\mathcal{V}^\sharp \to \mathbb{R}$ is independent from the choice of the norms $||\cdot||$ and $||\cdot||^\sharp$.

Definition 4.1 An autonomous differential system over $\mathcal{V}$ is given by a map F from $\mathcal{V} \to \mathbb{R}$ to $\mathcal{V} \to \mathbb{R}$ which is continuously differentiable and an (initial) state ρ_0.

By the Cauchy-Lipschitz theorem, an autonomous differential system (F, ρ_0) defines a unique maximal differentiable function $f : [0, T) \to (\mathcal{V} \to \mathbb{R})$ such that, for any $f(0) = \rho_0$ and $f' = F \circ f$, with $T \leqslant +\infty$. This unique f is called *the solution* of (F, ρ_0).

4.2 Exact reduction of differential semantics

Definition 4.2 A reduction of an autonomous differential system (F, ρ_0) is a commuting square:

$$\begin{tikzcd}
{} \arrow[r, "F"] \arrow[d, "\phi"'] & {} \arrow[d, "\phi"] \\
{} \arrow[r, "F^\sharp"'] & {}
\end{tikzcd}$$

where, ϕ is a linear positive expansive mapping between $\mathcal{V} \to \mathbb{R}$ and $\mathcal{V}^\sharp \to \mathbb{R}$, for a given finite set $\mathcal{V}^\sharp$ of (abstract) variables; and $F^\sharp$ is a continuously differentiable map between $\mathcal{V}^\sharp \to \mathbb{R}$ and $\mathcal{V}^\sharp \to \mathbb{R}$.

Theorem 4.3 ([7]) *Let $f : [0, T) \to (\mathcal{V} \to \mathbb{R})$ be the solution of (F, ρ_0), and $f^\sharp : [0, T^\sharp) \to (\mathcal{V}^\sharp \to \mathbb{R})$ be the solution of $(F^\sharp, \phi(\rho_0))$, then $T = T^\sharp$ and $f^\sharp = \phi \circ f$. Moreover, if f is positive, then $f^\sharp$ is positive as well.*

4.3 Combining existing reductions with projections-based reductions

Let $\mathcal{V}$ and $\mathcal{V}^\sharp$ be two finite sets of concrete and abstract variables. We assume that we are given an abstraction function ϕ between $\mathcal{V} \to \mathbb{R}$ and $\mathcal{V}^\sharp \to \mathbb{R}$, and two maps F and $F^\sharp$ between $\mathcal{V} \to \mathbb{R}$ and $\mathcal{V} \to \mathbb{R}$; and between $\mathcal{V}^\sharp \to \mathbb{R}$ and $\mathcal{V}^\sharp \to \mathbb{R}$, such that the following diagram:

$$\begin{tikzcd}
{} \arrow[r, "F"] \arrow[d, "\phi"'] \arrow[dr, phantom, "\mathtt{I}"] & {} \arrow[d, "\phi"] \\
{} \arrow[r, "F^\sharp"'] & {}
\end{tikzcd}$$

is a model reduction.

Moreover, we assume that we are given a map P such that the following diagram:

$$\begin{tikzcd}
{} \arrow[r, "F"] \arrow[dd, "P"'] & {} \arrow[dr, "P"] & \\
& \mathtt{II} & {} \\
{} \arrow[r, "F"'] & {} \arrow[ur, "P"'] &
\end{tikzcd}$$

commutes.

We also consider two maps $P^\sharp$ and $P^\sharp_0$ such that: (i) $P^\sharp$ is a linear expansive mapping over $\mathcal{V}^\sharp \to \mathbb{R}$, (ii) $P^\sharp_0$ is a map over $\mathcal{V}^\sharp \to \mathbb{R}$, and (iii) the following diagrams:

$$\begin{tikzcd}
{} \arrow[r, "P"] \arrow[d, "\phi"'] \arrow[dr, phantom, "\mathtt{IV}"] & {} \arrow[d, "\phi"] \\
{} \arrow[r, "P^\sharp"'] & {}
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
{} \arrow[rr, "P^\sharp"] \arrow[dr, "P^\sharp"'] & \mathtt{III} & {} \\
& {} \arrow[ur, "P^\sharp_0"'] &
\end{tikzcd}$$

commute.

Theorem 4.4 *Under these assumptions, the following diagram:*

$$\begin{tikzcd}
{} & & \text{II} & & {} \\
\text{IV} & {} & {} & {} & \text{IV} \\
{} & & & & {} \\
{} & {} & {} & {} & {}
\arrow[from=1-1, to=1-5, "F"]
\arrow[from=1-1, to=2-2, "P"]
\arrow[from=1-5, to=2-4, "P"']
\arrow[from=2-2, to=2-3, "F"]
\arrow[from=2-3, to=2-4, "P"]
\arrow[from=1-1, to=3-1, "\phi"']
\arrow[from=3-1, to=4-1, "P^\sharp"']
\arrow[from=3-1, to=4-2, "P^\sharp"]
\arrow[from=4-1, to=4-2, "P^\sharp_0"']
\arrow[from=2-2, to=4-2, "\phi"']
\arrow[from=2-3, to=4-3, "\phi"']
\arrow[from=2-4, to=4-5, "\phi"']
\arrow[from=1-5, to=3-5, "\phi"]
\arrow[from=3-5, to=4-5, "P^\sharp"]
\arrow[from=4-2, to=4-3, "F^\sharp"']
\arrow[from=4-3, to=4-5, "P^\sharp"']
\arrow[from=2-2, to=4-3, phantom, "\text{I}"]
\arrow[from=2-3, to=4-4, phantom, "\text{IV}"]
\arrow[from=4-1, to=3-2, phantom, "\text{III}"]
\end{tikzcd}$$

is a model reduction of the initial autonomous differential system (F, ρ_0).

Interestingly, we notice that no commutative diagram was required to relate the functions $F^\sharp$ and $P^\sharp$. As a consequence, the technical proof (diagram II) needs to be done formally once for all in the concrete.

4.4 Application

In practice, the mappings P,$P^\sharp$, and $P^\sharp_0$ are linear projections which are defined thanks to the choice of a representative for each equivalence class of some given equivalence relations of concrete and abstract variables. Let $\sim_S$ be an equivalence relation over a set S. Let $\text{Rep}_{\sim_S}$ be a subset of S such that any $\sim_S$-equivalence class C contains exactly one element e in $\text{Rep}_{\sim_S}$. In such a case, we say that e is the representative of the equivalence class C. We introduce two linear projections P and P_0 over $S \to \mathbb{R}$ as follows: for any function ρ between S and $\mathbb{R}$ and any element $e \in S$, whenever e is a representative in $\text{Rep}_{\sim_S}$, we define $P(\rho)(e)$ as the sum of the expressions $\rho(e')$ for any element $e' \in S$ such as $e' \sim_S e$, and $P_0(\rho)(e) := \rho(e)$; otherwise, we set $P(\rho)(e) := 0$ and $P_0(\rho(e)) := 0$. The projections P and P_0 have the same set of fixpoints. So $P_0 \circ P = P$. Moreover, the projection P is expansive. We call the pair (P, P_0) the pair of linear projections that is induced by the triple $(S, \sim_S, \text{Rep}_{\sim_S})$.

Theorem 4.4 can be used both to design new model reductions and to abstract further existing model reductions. We illustrate this on our case study (eg. see Sect. 2). We consider the set S of partial configurations for the protein X (a partial configuration is defined by a subset of sites $x_1, \ldots, x_m$ and a subset of sites $y_1, \ldots, y_n$ each of them being phosphorylated, or not). Partial configurations are ordered by an inclusion relation $\lhd$: we write $a \lhd a'$ whenever (i) a documents a subset of the sites that are documented in a', and (ii) each site s documented in a has the same phosphorylation state in a as in a'. Then, we define the equivalence relation $\sim_S$ over S, as $a \sim_S a'$ if and only if, a and a' (i) document the same set of sites, (ii) have the same number of sites x_i phosphorylated, (iii) have the same set of sites y_j phosphorylated. We define the set of representatives $\text{Rep}_{\sim_S}$ as the set of partial configurations for which there exists an integer k between 1 and m such that all the sites $x_1, \ldots, x_k$ are either phosphorylated, or not documented, and all the sites $x_{k+1}, \ldots, x_m$ are either unphosphorylated, or not documented. We notice that for any partial configuration a such as $a \in \text{Rep}_{\sim_S}$, any partial configuration a' such that $a' \lhd a$ is in the set $\text{Rep}_{\sim_S}$ of representatives. A set of representatives which satisfies this property is called *homogeneous*.

We define the set $\mathcal{V}$ of variables as the set of (complete) configurations for the protein X (there are 2^{m+n} of them). The mapping F is obtained by applying mass action laws on the set of reactions that is given in Fig. 1. We denote by $\sim_{\mathcal{V}}$ the restriction of the relation $\sim_S$ to the elements in $\mathcal{V}$. We define the pair (P, P_0) as the pair of projections that is induced by the triple $(\mathcal{V}, \sim_{\mathcal{V}}, \mathrm{Rep}_{\sim_S} \cap \mathcal{V})$. The property $P \circ F = P \circ F \circ P$ comes from the fact that $\sim_{\mathcal{V}}$ induces a forward (straight) bisimulation over chemical mixtures as formalized in Sect. 6.4.

We can use theorem 4.4 to design a model reduction. We set $\mathcal{V}^\sharp := \mathcal{V}$, $\phi := id_{\mathcal{V}\to\mathbb{R}}$, $F^\sharp := F$, $P^\sharp := P$, and $P_0^\sharp := P_0$. Then, according to theorem 4.4, a model reduction is obtained by (i) applying the projection $P_0^\sharp$ on the input of F, which amounts to ignoring chemical species which are not the representative of their $\sim_{\mathcal{V}}$-equivalence class, and (ii) applying the projection $P^\sharp$ on the output, which amounts to collecting the contribution to the derivative of the concentrations of $\sim_{\mathcal{V}}$-equivalent chemical species.

More interestingly, we can use Theorem 4.4 so as to abstract further an existing model reduction. By applying the framework in [10,7], we can get a model reduction by abstracting away the correlation between the phosphorylation state of the sites in the list $y_1, \ldots, y_n$. Doing so, we define the set of abstract variables $\mathcal{V}^\sharp$ as the set of specific partial configurations for the protein X, called *fragments*. In a fragment, all the sites within the list $x_1, \ldots, x_m$ are documented, whereas exactly one site within the list $y_1, \ldots, y_n$ is documented. Thus there are $n2^{m+1}$ fragments.

The function ϕ summarizes the concentration of chemical species that contain a given fragment. For any concrete state $\rho \in \mathcal{V} \to \mathbb{R}$ and any fragment $v^\sharp \in \mathcal{V}^\sharp$, we set: $\phi(\rho)(v^\sharp) := \sum_v (\rho(v) \mid v^\sharp \lhd v)$. The relation $\sim_{\mathcal{V}^\sharp}$ over $\mathcal{V}^\sharp$ is defined as the restriction of the relation $\sim_S$ to the elements in $\mathcal{V}^\sharp$. We define the pair $(P^\sharp, P_0^\sharp)$ as the pair of projections that is induced by the triple $(\mathcal{V}^\sharp, \sim_{\mathcal{V}^\sharp}, \mathrm{Rep}_{\sim_S} \cap \mathcal{V}^\sharp)$. The commutative diagram (IV) comes from the definition of ϕ and the fact that the set $\mathrm{Rep}_{\sim_S}$ is homogeneous. As a consequence, a new model reduction can be derived. It remains only $2(m+1)n$ fragments. The mapping $P^\sharp \circ F^\sharp \circ P_0^\sharp$ can be computed by discarding the input which are not representative of their $\sim_{\mathcal{V}^\sharp}$-equivalence class, and by gathering the contribution of the fragments which belong to the same $\sim_{\mathcal{V}^\sharp}$-equivalence class.

We generalize this construction to any model written in Kappa in Sect. 6.4.

5 The Kappa language

We present a core of the Kappa language [9] in a process-like notation and we give its operational semantics. Particularly, we do not consider internal states, agent creation nor deletion, nor wild-card bond removals. These features are convenient when writing models. Yet they would raise no particular difficulties to our theoretical framework.

5.1 Syntax

We assume a finite set of agent names $\mathcal{A}$, representing different kinds of proteins; a finite set of sites $\mathcal{S}$, corresponding to protein domains; a signature map Σ from $\mathcal{A}$ to $\wp(\mathcal{S})$, listing the domains of a protein which can be bound; and a countable set of binding label $\mathbb{L}$.

Definition 5.1 An *agent* $A(\sigma)$ is defined by its type $A \in \mathcal{A}$ and its *interface* σ, where σ is a partial mapping between $\Sigma(A)$ and $\{\epsilon\} \uplus \mathbb{L}$. A site s such that $\sigma(s) = \epsilon$ is *free*, whereas a site s such that $\sigma(s) \in \mathbb{L}$ is *bound*. An interface is usually denoted as a list of sites and the binding state of each site is written in superscript. The detailed description of the syntax of an agent is given by the following grammar:

$$\begin{array}{rcll@{\qquad\qquad}rcll}
a & ::= & N(\sigma) & \text{(agent)} & s & ::= & n^{\lambda} & \text{(site)} \\
N & ::= & A \in \mathcal{A} & \text{(agent name)} & n & ::= & x \in \mathcal{S} & \text{(site name)} \\
\sigma & ::= & \epsilon \mid s{,}\sigma & \text{(interface)} & \lambda & ::= & \epsilon \mid i \in \mathbb{L} & \text{(binding state)}
\end{array}$$

We generally omit the superscript ϵ.

Definition 5.2 An *expression* E is a set of agents $A(\sigma)$. Thus the syntax of an expression is defined as follows:

$$E ::= \epsilon \mid a, E.$$

Definition 5.3 A *pattern* is an expression which satisfies the following three conditions: (i) no site name occurs more than once in a given interface; (ii) each site name s in the interface of the agent A occurs in $\Sigma(A)$; (iii) each binding label $i \in \mathbb{L}$ occurs exactly twice if it does at all. The set of binding labels which occur in a pattern E is usually denoted as $\mathrm{B}_{\mathbb{L}}(E)$. A *mixture* is a pattern that is fully specified, ie each agent A documents its full interface $\Sigma(A)$. A *chemical species* is a fully connected mixture (that is to say that a mixture $a_1, \ldots, a_n$ is a species, if for any subsequence $a_{i_1}, \ldots, a_{i_m}$ of agents of this mixture $a_{i_1}, \ldots, a_{i_m}$ is not a mixture).

Definition 5.4 A *rule* r is defined by two patterns E_ℓ and E_r, and a rate $k \in \mathbb{R}_0^+$, written $r := E_\ell \to E_r@k$ such that E_ℓ and E_r have the same number n of agents, and for any integer i between 1 and n, the i-th agent of E_ℓ and E_r have the same agent type and document the same set of sites (with potentially different binding states).

Moreover, a pair of rules of the form $(E_\ell \to E_r@k, E_r \to E_\ell@k_d)$ is usually denoted by $E_\ell \leftrightarrow E_r@k, k_d$.

Example 5.5 Now we can refactor the case study of Sect. 2 in Kappa. The signature of the model is given by $\mathcal{A} = \{A, P\}$, $\mathcal{S} = \{s, x_1, x_2, y_1\}$, $\Sigma(A) = \{x_1, x_2, y_1\}$, and $\Sigma(P) = \{s\}$. Reactions are modelled by the following rules:

$$\begin{array}{rcll}
P(s)\ ,X(x_1) & \leftrightarrow & P(s^1)\ ,X(x_1{}^1) & @k,k_d\\
P(s)\ ,X(x_2) & \leftrightarrow & P(s^1)\ ,X(x_2{}^1) & @k,k_d\\
P(s)\ ,X(x_1,x_2,y_1) & \leftrightarrow & P(s^1)\ ,X(x_1,x_2,y_1{}^1) & @k_{1,0},k_d\\
P(s^1)\ ,P(s)\ ,X(x_1{}^1,x_2,y_1) & \leftrightarrow & P(s^1)\ ,P(s^2)\ ,X(x_1{}^1,x_2,y_1{}^2) & @k_{1,1},k_d\\
P(s^1)\ ,P(s)\ ,X(x_1,x_2{}^1,y_1) & \leftrightarrow & P(s^1)\ ,P(s^2)\ ,X(x_1,x_2{}^1,y_1{}^2) & @k_{1,1},k_d\\
P(s^1)\ ,P(s^2)\ ,P(s)\ ,X(x_1{}^1,x_2{}^2,y_1) & \leftrightarrow & P(s^1)\ ,P(s^2)\ ,P(s^3)\ ,X(x_1{}^1,x_2{}^2,y_1{}^3) & @k_{1,2},k_d
\end{array}$$

5.2 Operational semantics

We assume that we are given a set of rules $\mathcal{R}$. Now we define a transition system over the set of mixtures (up-to α equivalence). This transition system will be used when defining both the stochastic and the differential semantics of the set of rules $\mathcal{R}$. Moreover, each transition will be labeled by a proof that the transition is enabled and by a kinetic rate.

The application of a rule $r := E_\ell \to E_r@k$ to a mixture E is performed in several steps. First, we apply α-equivalence in order not only to equate the binding labels in E and E_ℓ, but also to avoid conflicts between former binding labels in E and the labels of the bonds that are created by the rule application. Then, we introduce an *embedding* so as to map agents of the pattern E_ℓ to agents of the mixture E. The result of the application of the rules is obtained by overwriting the binding states of the sites of the mixture by the states of the corresponding sites in the rhs E_r of the rule r.

Now we give a formal definition of these primitives.

Definition 5.6 The α-equivalence, $\sim_\alpha$, is defined as the smallest binary equivalence relation between expressions such that $E_{[i/j]} \sim_\alpha E$ for any pair $(i,j) \in \mathbb{L}^2$ of binding labels, such that $i \notin \mathrm{B}_{\mathbb{L}}(E)$.

Definition 5.7 Let $A_\ell(\sigma_\ell), B(\sigma)$ be two agents, we say that the agent $B(\sigma)$ *matches* the agent $A_\ell(\sigma_\ell)$, if and only if, $A_\ell = B$ and for any $s \in \Sigma(A_\ell)$ such that $\sigma_\ell(s)$ is defined, then (i) $\sigma(s)$ is also defined and (ii) $\sigma(s) = \sigma_\ell(s)$.

Definition 5.8 Let E_ℓ, E be two patterns. We denote by n_ℓ (resp. n) the number of agents in E_ℓ (resp. in E). An *embedding* between the patterns E_ℓ and E is denoted by an into mapping ϕ between the set $\{i \in \mathbb{N} \mid 1 \leqslant i \leqslant n_\ell\}$ and the set $\{j \in \mathbb{N} \mid 1 \leqslant j \leqslant n\}$ such that, for any i, the agent $A(\sigma)$ at position $\phi(i)$ in E matches the agent $A_\ell(\sigma_\ell)$ at position i in E_ℓ. In such a case, we write $E_\ell \lhd_\phi E$.

Moreover, a *straight embedding*, is an embedding ϕ between two patterns E_ℓ and E such that: whenever two agents belong to distinct connected components in E_ℓ, then their image (by ϕ) is also mapped to agents which belong to distinct connected components. In such a case, we write $E_\ell |\!\!\lhd_\phi E$.

Definition 5.9 Let $A(\sigma_r)$ and $A(\sigma)$ be two agents of the same type. We define the replacement of $A(\sigma)$ with $A(\sigma_r)$ as the agent $A(\sigma')$ where σ' is defined for any site s such that $\sigma(s)$ or $\sigma_r(s)$ is defined. In such a case, $\sigma'(s) = \sigma_r(s)$ whenever

$\sigma_r(s)$ is defined; and $\sigma'(s) = \sigma(s)$ otherwise. The replacement of $A(\sigma)$ with $A(\sigma_r)$ is denoted by $A(\sigma)[A(\sigma_r)]$.

Definition 5.10 Let r be a rule $E_\ell \to E_r@k$, $E = a_1, \ldots, a_m$ be a pattern, and ϕ be an embedding such that $E_\ell \lhd_\phi E$. We write $E_r = b_1, \ldots, b_n$. Then the result of the application of r on E along ϕ, which is denoted by $E[E_r]_\phi$ is defined as the mixture $a'_1, \ldots, a'_m$, where for any integer i such that $1 \leqslant i \leqslant m$, a'_m is defined as $a_i[b_j]$ whenever there exists an integer j such that $\phi(j) = i$, and as a_i otherwise.

Now we can define the transition system associated to a set of rules.

Definition 5.11 The states of the system are $\sim_\alpha$-classes of mixtures, and the set of transitions is defined by:

$$[E]_{\sim_\alpha} \xrightarrow[\gamma]{(r,[E]_{\sim_\alpha},\phi)} [E[E_r]_\phi]_{\sim_\alpha}$$

for any mixture E, any rule $r = E_\ell \to E_r@k$ such that r is a rule in $\mathcal{R}$ and $\mathrm{B}_{\mathbb{L}}(E) \cap (\mathrm{B}_{\mathbb{L}}(E_r) \backslash \mathrm{B}_{\mathbb{L}}(E_\ell)) = \varnothing$, any embedding ϕ between E_ℓ and E, and where γ is defined as the quotient between the rate k and the number of automorphisms (i.e. embeddings between E_ℓ and E_ℓ), $\mathit{Card}(\mathrm{Auto}(E_\ell))$, in E_ℓ.

Moreover, a transition $q \xrightarrow[\gamma]{\lambda} q'$, which is defined thanks to a straight embedding, is also denoted by $q \overset{\lambda}{\underset{\gamma}{\longmapsto}} q'$.

We notice, that agents types and agents sites (but not their binding states) are preserved by transition application. Moreover, an agent can be tracked along a sequence of transitions thanks to its position.

Example 5.12 Let us show how to apply the rule $E_\ell \to E_r@k_{1,1}$ to the state $[E]_{\sim_\alpha}$ where: the lhs E_ℓ is defined as $P\big(s^1\big)$, $P(s)$, $X\big(x_1{}^1, x_2, y_1\big)$; the rhs E_r is defined as $P\big(s^1\big)$, $P\big(s^2\big)$, $X\big(x_1{}^1, x_2, y_1{}^2\big)$, and the mixture E is defined as $P\big(s^3\big)$, $P\big(s^2\big)$, $P(s)$, $X\big(x_1{}^3, x_2, y_1\big)$, $X\big(x_1, x_2{}^2, y_1\big)$.

We can apply the reaction r along the embedding $\phi := [1 \mapsto 1, 2 \mapsto 3, 3 \mapsto 4]$. First we apply $\sim_\alpha$: we replace the binding label 3 with the binding label 1, so that binding labels match, and the binding label 2 with the binding label 4, so as to avoid capture. Then we notice that $E_\ell \lhd_\phi E_{[3/1,4/2]}$. Moreover $(E_{[3/1,4/2]})[E_r]_\phi$ is equal to the mixture E' that is defined as: $P\big(s^1\big)$, $P\big(s^4\big)$, $P\big(s^2\big)$, $X\big(x_1{}^1, x_2, y_1{}^2\big)$, $X\big(x_1, x_2{}^4, y_1\big)$.

Thus we have: $[E]_{\sim_\alpha} \overset{(r,[E]_{\sim_\alpha},\phi)}{\underset{k_{1,1}}{\longmapsto}} [E']_{\sim_\alpha}$ (since ϕ is a straight embedding).

6 Symmetric sites in Kappa

6.1 Action of a transposition

In this section, we formalize the actions of transpositions of two sites on patterns, rules, and transition labels. Then we define when two sites are symmetric in a given set of rules.

Definition 6.1 A *transposition of sites* is a triple $(i, x, y) \in \mathbb{N} \times \mathcal{S} \times \mathcal{S}$.

Let $t := (i, x, y)$ be a transposition of sites. Intuitively, the transposition t denotes that we want to permute the binding state of the sites x and y in the i-th agent of a pattern, or a rule. Let $E := N_1(\sigma_1)\ ,\ \ldots\ ,N_k(\sigma_k)$ be a pattern and $r_1 := E_\ell \rightarrow E_r@k$ be a rule.

Definition 6.2 We say that the transposition (i, x, y) operates on the pattern E, if and only if, $i \leqslant k$ and x, y belongs to $\Sigma(N_i)$. In such a case, we introduce the pattern $E' := N_1(\sigma'_1)\ ,\ \ldots\ ,N_k(\sigma'_k)$, where for any j between 1 and k, $\sigma'_j := \sigma_j$ whenever $j \neq i$, and σ'_i is defined as the interface which has the same domain definition as σ_i, and which satisfies: $\sigma'_i(x) := \sigma_i(y)$, $\sigma'_i(y) := \sigma_i(x)$, and $\sigma'_i(z) := \sigma_i(z)$ for any site $z \in \mathrm{dom}(\sigma_i)\backslash\{x, y\}$.

The pattern E' is called the *action of the transposition* (i, x, y) *on the mixture* E, and is denoted by $\mathrm{subs}(i, \frac{x}{y}, E)$.

Definition 6.3 Whenever the transposition t operates on E_ℓ, the rule $r'_1 := \mathrm{subs}(i, \frac{x}{y}, E_\ell) \rightarrow \mathrm{subs}(i, \frac{x}{y}, E_r)@\frac{k\,\mathit{Card}(\mathrm{Auto}(\mathrm{subs}(i,\frac{x}{y},E_\ell)))}{\mathit{Card}(\mathrm{Auto}(E_\ell))}$ is well-defined, if and only if the rule r_1 is well-defined. In such a case, the rule r'_1 is called the *action of the transposition* (i, x, y) *on the rule* r_1, and is denoted by $\mathrm{subs}_{\mathcal{R}}(i, \frac{x}{y}, r_1)$.

Let $\lambda := (r_2, [E]_{\sim_\alpha}, \phi)$ be a transition label.

Definition 6.4 We assume that the transposition t operates on E. Let r'_2 be the rule that is defined as r_2 whenever there is no integer j such as $\phi(j) = i$; or $\mathrm{subs}_{\mathcal{R}}(\phi^{-1}(i), \frac{x}{y}, r_2)$ otherwise. Let E' be the mixture $\mathrm{subs}(i, \frac{x}{y}, E)$. Then ϕ is also an embedding between the lhs of r'_2 and E', we call the triple $(r'_2, [E']_{\sim_\alpha}, \phi)$ the *action of the transposition* (i, x, y) *on the transition label* λ, and we denote it by $\mathrm{subs}_{\mathcal{L}}(i, \frac{x}{y}, \lambda)$.

Proposition 6.5 *We have:*

$$[E]_{\sim_\alpha} \xrightarrow[\gamma]{\lambda} [E']_{\sim_\alpha} \Longleftrightarrow [\mathrm{subs}(i, \tfrac{x}{y}, E)]_{\sim_\alpha} \xrightarrow[\gamma]{\mathrm{subs}_{\mathcal{L}}(i,\frac{x}{y},\lambda)} [\mathrm{subs}(i, \tfrac{x}{y}, E')]_{\sim_\alpha}.$$

6.2 Definition of symmetric sites

We consider an agent type $A \in \mathcal{A}$, and two sites $x, y \in \Sigma(A)$ in its interface.

Definition 6.6 The sites x and y are *symmetric* in A for the set of rules $\mathcal{R}$, if for any rule $r := N_1(\sigma_1)\ ,\ \ldots\ ,N_l(\sigma_l) \rightarrow E_r@k$ in $\mathcal{R}$ and any integer i between 1 and l such that (i) $N_i = A$ and (ii) the transposition (i, x, y) operates on r, the rule $\mathrm{subs}_{\mathcal{R}}(i, \frac{x}{y}, r)$ belongs to the set of rules $\mathcal{R}$ as well.

So as to avoid testing, for all triples (A, x, y) such that $A \in \mathcal{A}$ and x,$y \in \Sigma(A)$, whether the condition in Def. 6.6 is satisfied, or not we preselect the potential triples thanks to a weaker property. Thus, we use the *contact map* which is the summary of all bonds which may appear in the rhs of a rule in $\mathcal{R}$. The contact map is a graph, the set of nodes are the pairs (A, x) with $A \in \mathcal{A}$ and $x \in \Sigma(A)$, and the set of edges are the pairs $((A, x), (A', x'))$ of nodes for which there exists a bond between the site x of an instance of an agent A, and the site x' of an instance of an agent A' in the rhs of a rule in $\mathcal{R}$. As an example, we depict in Fig. 6 the contact map of our case study of Sect. 2.

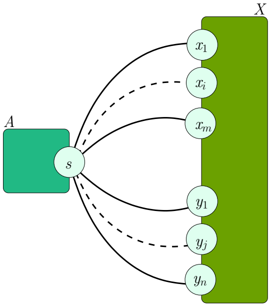

Fig. 6: A contact map.

Proposition 6.7 *If the sites x and y are symmetric in A for the set of rules $\mathcal{R}$, then the nodes (A, x) and (A, y) have the same set of partners in the contact map of the set of rules $\mathcal{R}$.*

6.3 Application to the reduction of stochastic semantics

We review the individual stochastic semantics of Kappa and show that symmetries among sites can be used to design an abstraction of it.

We consider an initial mixture $E_0 := N_1(\sigma_1), \ldots, N_k(\sigma_k)$. We associate the pair $(\mathcal{R}, E_0)$ with a WLTS $(\mathcal{Q}, \mathcal{L}, \rightarrow, w, \mathcal{I}, \pi_0)$. The state space $\mathcal{Q}$ of the system is the set of reachable mixtures (up to $\sim_\alpha$-equivalence); the set of transition labels $\mathcal{L}$ is the set of triples $(r, [E]_{\sim_\alpha}, \phi)$ where r is a rule in $\mathcal{R}$, E a mixture, and ϕ and embedding between the lhs of r and E; the transition relation $\rightarrow$ is defined as $q \xrightarrow{\lambda} q'$ whenever there exists a transition $q \xrightarrow[\gamma]{\lambda} q'$ in the operational semantics (eg see Section 5.2), moreover the weight of this transition is defined as $w(q, \lambda) := \gamma$; the set of initial states $\mathcal{I}$ is defined as the singleton $\{[E_0]_{\sim_\alpha}\}$, and its initial probability is defined as $\pi_0([E_0]_{\sim_\alpha}) := 1$.

We consider a set T of triples (A, x, y) such that the sites x and y in A are symmetric for the set of rules $\mathcal{R}$ and $E_0 \sim_\alpha \mathrm{subs}(i, \frac{x}{y}, E_0)$ for any integer i between 1 and k such that $N_i = A$. We introduce the equivalence relation $\sim_\mathcal{Q}$ over $\mathcal{Q}$ as the smallest equivalence relation such that $[E]_{\sim_\alpha} \sim_\mathcal{Q} [E']_{\sim_\alpha}$ whenever there exist a triple $(A, x, y) \in T$ and an integer i such that the i-th agent in E is of type A and $E' \sim_\alpha \mathrm{subs}(i, \frac{x}{y}, E)$. The same way, we define the equivalence relation $\sim_\mathcal{L}$ over $\mathcal{L}$ as the smallest equivalence relation such that $\lambda \sim_\mathcal{L} \lambda'$ whenever, if we denote $\lambda := (r, [E]_{\sim_\alpha}, \phi)$, there exist a triple $(A, x, y) \in T$ and an integer i, such that the i-th agent in E is of type A and $\lambda' = \mathrm{subs}_\mathcal{L}(i, \frac{x}{y}, \lambda)$.

By Proposition 6.5, the pair of equivalence relations $(\sim_\mathcal{Q}, \sim_\mathcal{L})$ defines a backward bisimulation [4]. That is to say that: (i) for any pair q_1, q_2 of $\sim_\mathcal{Q}$-equivalent states in $\mathcal{Q}$, we have $\mathrm{a}(q_1) = \mathrm{a}(q_2)$ and (ii) for any $\sim_\mathcal{Q}$-equivalence class $C_\mathcal{Q}$ of states, any $\sim_\mathcal{L}$-equivalence class $C_\mathcal{L}$ of transition labels, any pair (q'_1, q'_2) of states such that $q'_1 \sim_\mathcal{Q} q'_2$, we have: $\mathrm{bw}(C_\mathcal{Q}, C_\mathcal{L}, q'_1) = \mathrm{bw}(C_\mathcal{Q}, C_\mathcal{L}, q'_2)$, where for any state $q' \in \mathcal{Q}$, the flux $\mathrm{bw}(C_\mathcal{Q}, C_\mathcal{L}, q')$ from the class of states $C_\mathcal{Q}$ to the state q' through the transition steps with a label in the class $C_\mathcal{L}$ is defined as the sum of the expression

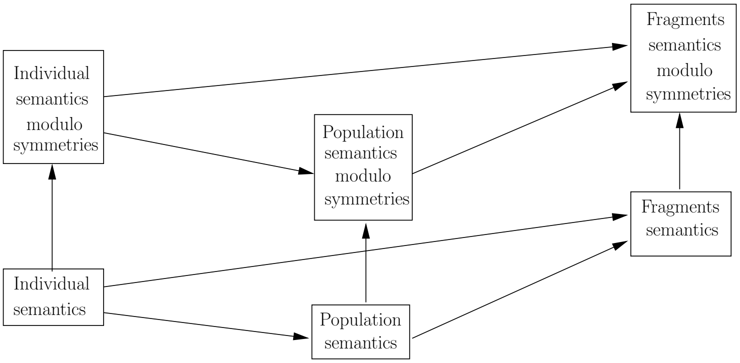
Individual semantics modulo symmetries
Population semantics modulo symmetries
Fragments semantics modulo symmetries
Fragments semantics
Individual semantics
Population semantics

Fig. 7. Hierarchy of stochastic semantics.

$w(q,\lambda)$, for any state $q \in C_{\mathcal{Q}}$ and any transition label $\lambda \in C_{\mathcal{L}}$, such that $q \xrightarrow{\lambda} q'$. Besides, by assumption on the initial mixture, for any pair $q_1, q_2 \in \mathcal{Q}$ such that $q_1 \in \mathcal{I}$ and $q_1 \sim_{\mathcal{Q}} q_2$, we have: $q_2 = q_1$ (and thus $q_2 \in \mathcal{I}$ and $\pi_0(q_1) = \pi_0(q_2)$). Thus, the pair $(\sim_{\mathcal{Q}}, \sim_{\mathcal{L}})$ induces an abstraction between the initial WLTS and a reduced WLTS where states are quotiented by $\sim_{\mathcal{Q}}$ and transition labels are quotiented by $\sim_{\mathcal{L}}$.

This abstraction can be combined with other existing abstractions such as the population semantics or the fragments semantics (both defined in [11]). Since the product of abstractions distributes over the composition of abstractions, we get the hierarchy of semantics which is depicted in Fig. 7.

6.4 Application to the reduction of differential semantics

We remind the differential semantics of Kappa and show that symmetries among sites can be used to design an abstraction of it. We can combine this abstraction with fragments-based model reductions [10,7].

First, we introduce a finite set of chemical species $\mathcal{V}$ closed under the rules in $\mathcal{R}$, which contains all species present in the system's initial state, and has at most one representative per species isomorphism class (i.e. whenever v and v' belongs to $\mathcal{V}$, and $v \lhd v'$, then $v = v'$).

A state is a mapping ρ between $\mathcal{V}$ and $\mathbb{R}^+$ (which denotes the concentration of each species). So as to define the function F which specifies the behavior of the system, we consider the set of chemical reactions which are generated by the set of rules $\mathcal{R}$. Given a rule $r := E_\ell \rightarrow E_r@k$ in $\mathcal{R}$, we may assume without any loss of generality that E_ℓ is written as $C_1, \ldots, C_k$ where each C_i is a connected pattern. A reaction is obtained, by choosing for any integer i between 1 and k, a reachable species R_i and an embedding ϕ_i between C_i and R_i. This way, we form a composite embedding $\phi := \sum_i \phi_i$ between E_ℓ and $R_1, \ldots, R_k$. Moreover, ϕ is a straight embedding. The result of the application of the rule r on $R_1, \ldots, R_k$ is, up to reordering of agents and α-conversion, a tuple of reachable species in $\mathcal{V}$ that we denote by $P_1, \ldots, P_l$. Then the function F is obtained by summing the contribution of each reaction, as follows:

$$F(\rho)(R_j) \overset{-}{=} \gamma \prod_i (\rho(R_i) \mid 1 \leqslant i \leqslant k), \qquad F(\rho)(P_{j'}) \overset{+}{=} \gamma \prod_i (\rho(R_i) \mid 1 \leqslant i \leqslant k).$$

where γ is the quotient between k and the number of automorphisms in E_ℓ, and j ranges between 1 and k, and j' between 1 and l.

A fragments-based model reduction is obtained by carefully choosing a set $\mathcal{V}^\sharp$ of partial species (pair-wisely non isomorphic) called fragments. The abstraction function ϕ is defined as in our case study: $\phi(\rho)(v^\sharp)$ is defined as a linear (the number of occurrence matters) combination of the concentration $\rho(v)$ of the species which contain this fragment.

Now we explain how to abstract such a model reduction further, by dealing with symmetries among sites. We consider a set T of triples (A, x, y) such that the sites x and y in A are symmetric for the set of rules $\mathcal{R}$ and such that for any agent of type A in a fragment the site x is documented, if and only if, the site y is documented. We introduce the equivalence relation $\sim$ over $\sim_\alpha$-equivalence classes of patterns as the smallest equivalence relation such that $[E]_{\sim_\alpha} \sim [E']_{\sim_\alpha}$ whenever there exist a triple (A, x, y) in T, and an integer i, such that the i-th agent in E is of type A, and $E' \sim_\alpha \mathrm{subs}(i, \frac{x}{y}, E)$. We consider $\mathrm{Rep}_\sim$ a set of homogeneous representatives for the $\sim$-equivalence classes (i.e. such as any connected part of representative is also a representative[8]). Let (P, P_0) be the pair of projections induced by the triple $(\mathcal{V}, \sim_\mathcal{V}, \mathrm{Rep}_\sim \cap \mathcal{V})$ and $(P^\sharp, P_0^\sharp)$ be the pair of projections that is induced by the triple $(\mathcal{V}^\sharp, \sim_\mathcal{V}^\sharp, \mathrm{Rep}_\sim \cap \mathcal{V}^\sharp)$. Proposition 6.5 ensures that $\sim$ defines a forward (straight) bisimulation. That is to say that for any $\sim$-equivalence class of states $C_{q'}$, and any pair q_1, q_2 of $\sim$-equivalent states, we have: $\mathrm{fw}(q_1, C_{q'}) = \mathrm{fw}(q_2, C_{q'})$ where for any state $q \in \mathcal{Q}$, the flux $\mathrm{fw}(q, C_{q'})$ is defined as the sum of the real numbers γ such that there exist a transition label λ and a state $q' \in C_{q'}$ which satisfy: $q \xmapsto[\gamma]{\lambda} q'$. It follows that: $P \circ F = P \circ F \circ P$. The other assumptions of Theorem 4.4 are all satisfied, which allows to reduce the system further by considering fragments up to symmetries among sites.

7 Conclusion

In this paper, we introduced two generic constructions to abstract further model reductions for stochastic and differential semantics. They can be used with hierarchies of model reductions easily, since most of the requirements is expressed in the non-reduced (concrete) model. We applied these frameworks to abstract existing fragments-based model reduction [10,7,11] so as to benefit from the fact that some sites have exactly the same capabilities of interaction.

[8] We omit technical details, but some algorithms can compute such sets of representatives.

References

[1] Blinov, M. L., J. R. Faeder and W. S. Hlavacek, *BioNetGen: software for rule-based modeling of signal transduction based on the interactions of molecular domains*, Bioinformatics **20** (2004).

[2] Borisov, N. M., N. I. Markevich, B. N. Kholodenko and E. D. Gilles, *Signaling through receptors and scaffolds: Independent interactions reduce combinatorial complexity*, Biophysical Journal **89** (2005).

[3] Buchholz, P., *Exact and ordinary lumpability in finite Markov chains*, Journal of Applied Probability **31** (1994).

[4] Buchholz, P., *Bisimulation relations for weighted automata*, Theor. Comp. Science **393** (2008).

[5] Conzelmann, H., J. Saez-Rodriguez, T. Sauter, B. N. Kholodenko and E. D. Gilles, *A domain-oriented approach to the reduction of combinatorial complexity in signal transduction networks*, BMC Bioinformatics **7** (2006).

[6] Cousot, P. and R. Cousot, *Systematic design of program analysis frameworks*, in: *POPL 1979*.

[7] Danos, V., J. Feret, W. Fontana, R. Harmer and J. Krivine, *Abstracting the differential semantics of rule-based models: exact and automated model reduction*, in: *LICS 2010*, to appear.

[8] Danos, V., J. Feret, W. Fontana and J. Krivine, *Scalable simulation of cellular signaling networks*, in: Z. Shao, editor, *APLAS 2007*, LNCS **4807**.

[9] Danos, V. and C. Laneve, *Core formal molecular biology*, Theor. Computer Science **325** (2003).

[10] Feret, J., V. Danos, J. Krivine, R. Harmer and W. Fontana, *Internal coarse-graining of molecular systems*, Proc. of the National Academy of Sciences **106** (2009).

[11] Feret, J., H. Koeppl and T. Petrov, *Stochastic fragments: A framework for the exact reduction of the stochastic semantics of rule-based models*, Int. Jour. of Software and Informatics To appear.