Automatic reduction of stochastic rules-based models in a nutshell

Ferdinanda Camporesi[*,†], Jérôme Feret[†], Heinz Koeppl[**] and Tatjana Petrov[**]

[*]*Dipartimento di Scienze dell'Informazione*
Università di Bologna
Bologna, Italy
[†]*Laboratoire d'informatique de l'École normale supérieure*
(INRIA/ÉNS/CNRS)
Paris, France
[**]*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 other, which leads to the formation of a huge number of distinct chemical species. Thus we cannot generate explicitly the quantitative semantics of these models, and it is even harder to compute their properties.

In this extended abstract, we summarize a framework for reducing the combinatorial complexity of models of biochemical networks. We use rules-based languages to describe the interactions between proteins. Then we compile these models into continuous-time Markov chains. Finally, we use backward bisimulations in order to reduce the dimension of the state space of these Markov chains. More specifically, these backward bisimulations are defined thanks to an abstraction of the control flow of information within chemical species and thanks to an algorithm which detects which protein sites have the same capabilities of interaction.

Keywords: Rules-based modeling, continuous-time Markov chains, model reduction
PACS: 87.16.A-, 87.16.ad, 87.16.Xa

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 kinds of chemical species. Rules-based modeling [1, 2] 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 continuous-time Markov chains, or master equation) and differential semantics cannot be explicitly written, because the state space is of the form $\mathbb{K}^n$ (with $\mathbb{K}$ equal to $\mathbb{N}$ or $\mathbb{R}^+$), where n is the number of reachable species. Model reduction [3, 4, 5, 6] consists in reducing the dimension of the state space, by discovering a coarser grain of observation. In [7, 8], we propose a framework to formalize model reductions for stochastic semantics by means of backward bisimulations [9, 10]. The soundness of this approach is ensured formally, and is stated in the following way: the density distribution of sets of traces in a reduced model is equal to the sum of the density distribution of sets of traces in the initial (i.e. non reduced) model. Moreover, the reduced model is still Markovian, (providing some further assumptions on the initial distribution of the model). Thus, our model reductions can be seen as a means to achieve weak lumpability [11].

In this paper, we illustrate the framework by applying it to a case study.

RULES-BASED MODELING

We introduce an example of a model of interaction between proteins and we give its encoding in a rules-based language called Kappa. More information about the language Kappa can be found in [1].

We consider two kinds of proteins P and U. We assume that each protein P has exactly $n_x + n_y$ binding sites, $x_1, \ldots, x_{n_x}, y_1, \ldots, y_{n_y}$, and that each protein U has exactly one binding site x. We also assume that, at time $t = 0$, there are n_P proteins of kind P injectively labeled with integer indexes that range between 1 and n_P, and n_U proteins of kind U injectively labeled with integer indexes that range between between 1 and n_U, and that no binding site is bound yet.

Now we describe the interactions between proteins. We assume that each binding site of a protein P can bind the binding site x of a protein U. In Kappa, these interactions can be described by the following rules:

$$U(x)\,,\, P(x_i) \to U\left(x^1\right)\,,\, P\left(x_i^1\right)@k_1 \qquad U(x)\,,\, P(y_i) \to U\left(x^1\right)\,,\, P\left(y_i^1\right)@k_2.$$

A rule is a symbolic representation of a set of reactions. Moreover, each rule is fitted with a rate constant, which denotes the velocity of each encoded reaction. Integer superscripts encode a pairing relation between sites: two sites with the same superscript are considered to be bound together. So as to apply a rule, one need to build an *embedding* between the left hand side (lhs) of a rule and a chemical soup. This embedding identifies to which proteins the rule is applied. The state of the proteins has to match with the preconditions displayed in the lhs of the rule. Then, the binding state of the sites of the proteins is updated according to the right hand side (binding superscripts may be injectively renamed so as to avoid conflict). For instance we can apply the rule: $U(x)\,,\, P(x_1) \to U\left(x^1\right)\,,\, P\left(x_1^1\right)@k_1$ to the state: $\{U_1\left(x^1\right), U_2(x), U_3\left(x^2\right), P_1\left(x_1^1,x_2\right), P_2\left(x_1,x_2^2\right)\}$ via the embedding which maps the protein U of the lhs of the rule to the protein U with the identifier 2, and the protein P of the lhs of the rule to the protein P with the identifier 2; doing so, we obtain the state: $\{U_1\left(x^1\right), U_2\left(x^3\right), U_3\left(x^2\right), P_1\left(x_1^1,x_2\right), P_2\left(x_1^3,x_2^2\right)\}$.

The protein U operates as an ubiquitin protein: whenever all the sites within the list $x_1, \ldots, x_{n_x}$ or all the sites within the list $y_1, \ldots, y_{n_y}$ of a protein P are bound, then this protein can be destroyed, which is denoted by the following rules:

$$U\left(x^1\right)\ldots U\left(x^{n_x}\right)\,,\, P\left(x_1^1,\ldots,x_{n_x}^{n_x}\right) \to \underbrace{U(x)\ldots U(x)}_{n_x}@k_3 \qquad U\left(x^1\right)\ldots U\left(x^{n_y}\right)\,,\, P\left(y_1^1,\ldots,y_{n_y}^{n_y}\right) \to \underbrace{U(x)\ldots U(x)}_{n_y}@k_4.$$

STOCHASTIC SEMANTICS

We describe a compilation procedure from rules-based models to weighted labeled transition systems (WLTS), a variant of continuous-time Markov chains with discrete transitions, where each transition is observed. We also define the continuous stochastic semantics of each WLTS as a density distribution of its computation traces.

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

In our example, we have a unique initial state, that we denote by q_0. Thus we have $\mathscr{I} = \{q_0\}$ and $\pi_0(q_0) = 1$. An element $(q, \lambda, q') \in \to$ denotes a transition from state q to state q'; the symbol λ is the label of the transition. Transition labels are used to identify computation steps, each label is made of the rule being used, the state of the system before applying the reaction, and the embedding between the rule lhs and the state of the system. We denote by $q \xrightarrow{\lambda} q'$ the fact that the tuple (q, λ, q') belongs to $\to$. We denote by $\mathscr{L}(q) \subseteq \mathscr{L}$ the set of labels for which there exists $q' \in \mathscr{Q}$ such that $q \xrightarrow{\lambda} q'$. The mapping w associate to each transition $q \xrightarrow{\lambda} q'$ the rate of the rules being used.

The stochastic semantics of a WLTS is defined as a probability density distribution of the finite traces. A *finite trace* is given by an initial state $q_0 \in \mathscr{I}$ and a finite sequence $(\lambda_i, t_i, q_i)_{1 \le i \le k} \in (\mathscr{L} \times \mathbb{R}^+ \times \mathscr{Q})^k$ of triples such that: for any integer i such that $1 \le i \le k$, we have $q_{i-1} \xrightarrow{\lambda_i} q_i$. Such a trace is denoted as: $q_0 \xrightarrow{\lambda_1, t_1} q_1 \cdots q_{k-1} \xrightarrow{\lambda_k, t_k} q_k$. The non negative real number t_i denotes the waiting time before the i-th transition step. Moreover, the number of transitions (here k) is called the size of the trace.

We introduce $\mathbb{IR}^+$ as the set of intervals of positive real numbers. Given a natural number $k \in \mathbb{N}$, an initial state $q_0 \in \mathscr{I}$ and a sequence $(\lambda_i, I_i, q_i)_{1 \le i \le k} \in (\mathscr{L} \times \mathbb{IR}^+ \times \mathscr{Q})^k$ of tuples, the set of traces that is defined as follows: $\{q_0 \xrightarrow{\lambda_1, t_1} q_1 \ldots q_{k-1} \xrightarrow{\lambda_k, t_k} q_k \mid t_i \in I_i\}$, is called a *cylinder set of traces*, and is denoted by $q_0 \xrightarrow{\lambda_1, I_1} q_1 \ldots q_{k-1} \xrightarrow{\lambda_k, I_k} q_k$. 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:

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

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

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 $\lambda \in \mathscr{L}(q)$ 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.

ABSTRACTIONS

The description of a system can be less or more fine grained, which leads to the notion of abstraction between WLTSs. We give three examples of abstractions formalized by the means of backward bisimulations [9, 10].

Indeed, the previous granularity of observation keeps too much information, so we propose to abstract away agent identifiers, which amounts to consider chemical soups as multi-sets of chemical species (where a chemical species is a connected component of proteins without identifiers).

In order to formalize this abstraction, we introduce two onto mapping $\beta^{\mathscr{Q}}$ and $\beta^{\mathscr{L}}$. The first one $\beta^{\mathscr{Q}}$ maps any state $q \in \mathscr{Q}$ (with protein identifiers) to the multi-set of chemical species that it contains. We introduce the equivalence relation $\sim_{\mathscr{Q}}$ which relates any pair of states in $\mathscr{Q}$ which have the same multi-set of chemical species. The mapping $\beta^{\mathscr{Q}}$ can be seen as the composition of the function $[\cdot]_{\sim_{\mathscr{Q}}}$ which maps any state to its $\sim_{\mathscr{Q}}$-equivalence class and an 1-to-1 function mapping each $\sim_{\mathscr{Q}}$-equivalence class to the multi-set of chemical species. The mapping $\beta^{\mathscr{L}}$ is obtained by mapping each transition label λ to its equivalence class $[\lambda]_{\sim_{\mathscr{L}}}$ where the equivalence relation $\sim_{\mathscr{L}}$ relates the transition labels of the transitions which operate the same way over chemical species (see [7] for a more formal definition). Equivalently, we could have renamed the equivalence classes of $\sim_{\mathscr{L}}$ thanks to a 1-to-1 mapping.

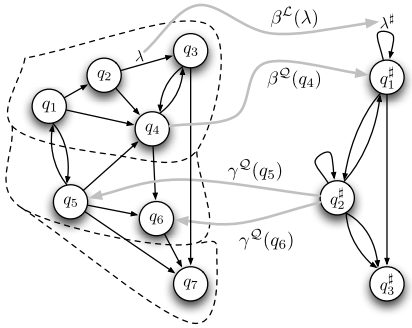

FIGURE 1. An abstraction between two transition systems.

The equivalence relations $\sim_{\mathscr{Q}}$ and $\sim_{\mathscr{L}}$ define a backward bisimulation [9, 10]. Indeed, if we define $\gamma^{\mathscr{Q}}$ as the function which maps each concrete state $q \in \mathscr{Q}$ to the inverse of the number of elements in the $\sim_{\mathscr{Q}}$-equivalence class of q, we get that: (i) for any $q_1, q_2 \in \mathscr{Q}$ such that $q_1 \sim_{\mathscr{Q}} q_2$, we have $\mathrm{a}(q_1) = \mathrm{a}(q_2)$; and (ii) for any $q_\star, q_1', q_2' \in \mathscr{Q}$ and $\lambda_\star \in \mathscr{L}$, $\mathrm{bwd}(q_\star, \lambda_\star, q_1') = \mathrm{bwd}(q_\star, \lambda_\star, q_2')$ where for any state $q' \in \mathscr{Q}$, $\mathrm{bwd}(q_\star, \lambda_\star, q')$ denotes the backward flow of q' from $q_\star$ via $\lambda_\star$, which is defined as $\sum_{q,\lambda}(\gamma^{\mathscr{Q}}(q)w(q,\lambda) \mid (q,\lambda) \in \mathscr{Q} \times \mathscr{L}$ s.t. $q \sim_{\mathscr{Q}} q_\star, \lambda \sim_{\mathscr{L}} \lambda_\star, q \xrightarrow{\lambda} q')$.

The three mappings $(\beta^{\mathscr{Q}}, \beta^{\mathscr{L}}, \gamma^{\mathscr{Q}})$ define an abstraction of the WLTS $\mathscr{S}$ (e.g., see Fig. 1): we define the tuple $\mathscr{S}^\sharp := (\mathscr{Q}^\sharp, \mathscr{L}^\sharp, \rightsquigarrow, w^\sharp, \mathscr{I}^\sharp, \pi_0^\sharp)$ as follows. The sets $\mathscr{Q}^\sharp$ and $\mathscr{L}^\sharp$ are the codomains of $\beta^{\mathscr{Q}}$ and $\beta^{\mathscr{L}}$. The transition relation $\rightsquigarrow$, the set of initial states $\mathscr{I}^\sharp$, and the initial distribution $\pi_0^\sharp$ are obtained by taking the quotient of $\rightarrow$, $\mathscr{I}$ and π_0 by the relations $\sim_{\mathscr{Q}}$ and $\sim_{\mathscr{L}}$. Besides, for any $q \xrightarrow{\lambda} q'$, the function $w^\sharp$ maps the pair $(\beta^{\mathscr{Q}}(q), \beta^{\mathscr{L}}(\lambda))$ to the backward flow $\mathrm{bwd}(q, \lambda, q')$ of the state q' from the state q via λ.

The tuple $\mathscr{S}^\sharp$ is indeed a WLTS, which is more abstract than $\mathscr{S}$, as expressed by these strong relationships: given a cylinder set τ of traces of the WLTS $\mathscr{S}$, we denote by $\beta^{\mathscr{T}}(\tau)$ the cylinder set of abstract traces of $\mathscr{S}^\sharp$ which is obtained by replacing each state q in τ with $\beta^{\mathscr{Q}}(q)$ and each transition label λ in τ with $\beta^{\mathscr{L}}(\lambda)$. Then (e.g. see [7] for a formal proof), (i) *soundness*: the density distribution of a cylinder set $\tau^\sharp$ of traces of $\mathscr{S}^\sharp$ is equal to the sum of the density distributions of τ' for any cylinder set τ' of traces of $\mathscr{S}$ such that $\beta^{\mathscr{T}}(\tau') = \tau^\sharp$ and (ii) *state completeness*: given a cylinder set $\tau^\sharp$ of traces of $\mathscr{S}^\sharp$ ending in the state $q^\sharp \in \mathscr{Q}^\sharp$, then, for any state $q, q' \in \mathscr{Q}$ such that $\beta^{\mathscr{Q}}(q) = q^\sharp = \beta^{\mathscr{Q}}(q')$, we have: $P(q \mid \tau^\sharp)\gamma^{\mathscr{Q}}(q') = P(q' \mid \tau^\sharp)\gamma^{\mathscr{Q}}(q)$, where for any abstract cylinder set $\tau^\sharp$ of traces and any state $q^\sharp$, $P(q, \tau^\sharp)$ denotes the conditional probability of being in the state q after having computed a trace in the (finite) join of the cylinder sets τ such that $\beta^{\mathscr{T}}(\tau) = \tau^\sharp$. It is worth noting that the soundness and the state-completeness properties hold only because the initial distribution is invariant by 1-to-1 re-indexing of protein identifiers. In general, the abstractions which are defined by backward bisimulations are correct only for specific initial distributions. Thus, they provide only a weak lumpability criterion [11].

We can abstract our system furthermore. Firstly, we can investigate the flow of information between binding sites of the proteins of kind P. Indeed, we notice that no rule mentions simultaneously a site in the list $x_1, \ldots, x_{n_x}$ and a site in the list $y_1, \ldots, y_{n_y}$. As a consequence we can safely split each protein P into two parts, and forget which pair of parts

belongs to the same protein. This abstraction can be formalized [7] by a triple $(\beta^{\mathcal{D}^\sharp}, \beta^{\mathcal{L}^\sharp}, \gamma^{\mathcal{D}^\sharp})$ of mappings as well, where $\beta^{\mathcal{D}^\sharp}$ maps a multi-set of chemical species into a multi-set of macrospecies (or *fragments* of chemical species).

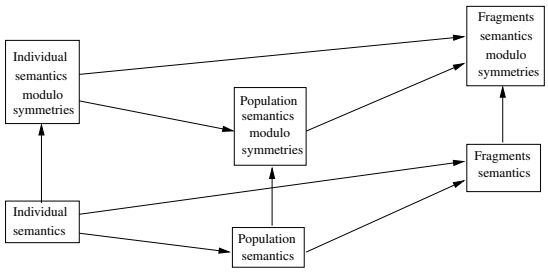

FIGURE 2. Hierarchy of stochastic semantics.

Secondly, we can notice that each site within the list $x_1,.,x_{n_x}$ (resp. within the list $y_1,\ldots,y_{n_y}$) has the same capability of interactions. Thus, the relevant information is how many sites within the list $x_1,\ldots,x_{n_x}$ and how many sites within the list $y_1,\ldots,y_{n_y}$ are bound in a protein P, i.e. it is not necessary to know for each of the sites whether it is bound or not. This abstraction amounts to consider that the sites within the list $x_1,\ldots,x_{n_x}$ (resp. within the list $y_1,\ldots,y_{n_y}$) are bound in increasing order, and can be formalized thanks to a triple of mappings [8].

Abstractions can be composed sequentially [7], or combined with a symmetric, commutative, and associative product operator [8]. In the latter case, the result is the most precise abstraction which is at least as abstract as the two abstractions being combined. This abstraction is well-defined modulo 1-to-1 renaming of abstract elements and abstract transition labels. Moreover, this product distributes over sequential composition. This way, we get the hierarchy of stochastic semantics which is given in Fig. 2.

CONCLUSION

We have summarized a framework for reducing the combinatorial explosion in stochastic models for signaling pathways. We have shown on a small case study how to reduce the set of observables, from $1+2^{n_x+n_y}$ chemical species, to $1+(n_x+1)+(n_y+1)$ fragments of chemical species, while preserving a strong relation between the different levels of abstractions (essentially, the density distribution of a trace $\tau^\sharp$ at a more abstract level, is equal to the collective density distribution of all the traces τ at a more concrete level which match the trace $\tau^\sharp$). Our framework is highly extensive, since abstractions can be composed sequentially, or combined in a associative and commutative way. So far, we have considered abstractions driven by the control flow of information among binding sites, or by the similarities between the capabilities of interactions of some binding sites. Other abstractions can be considered in the further work.

ACKNOWLEDGMENTS

The contribution of Ferdinanda Camporesi and Jérôme Feret was partially supported by the AbstractCell ANR-Chair of Excellence. Heinz Koeppl acknowledges the support from the Swiss National Science Foundation, grant no. 200020-117975/1. Tatjana Petrov acknowledges the support from SystemsX.ch, the Swiss Initiative in Systems Biology.

REFERENCES

1. V. Danos, and C. Laneve, *Theoretical Computer Science* **325** (2003).
2. M. L. Blinov, J. R. Faeder, and W. S. Hlavacek, *Bioinformatics* **20** (2004).
3. N. M. Borisov, N. I. Markevich, B. N. Kholodenko, and E. D. Gilles, *Biophysical Journal* **89** (2005).
4. H. Conzelmann, J. Saez-Rodriguez, T. Sauter, B. N. Kholodenko, and E. D. Gilles, *BMC Bioinformatics* **7** (2006).
5. J. Feret, V. Danos, J. Krivine, R. Harmer, and W. Fontana, *Proceedings of the National Academy of Sciences* **106** (2009).
6. V. Danos, 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.
7. J. Feret, H. Koeppl, and T. Petrov, *International Journal of Software and Informatics*, to appear.
8. F. Camporesi, J. Feret, H. Koeppl, and T. Petrov, "Combining model reductions," in *MFPS XXVI*, to appear.
9. P. Buchholz, *Journal of Applied Probability* **31** (1994).
10. P. Buchholz, *Theoretical Computer Science* **393** (2008).
11. G. Rubino, and B. Sericola, *Stochastic processes and their applications* **vol. 45, no 1**, 115–125 (1993).