Available online at www.sciencedirect.com

SciVerse ScienceDirect

Electronic Notes in Theoretical Computer Science 284 (2012) 105–124

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Model Decomposition and Stochastic Fragments

Tatjana Petrov[1] Arnab Ganguly[2] Heinz Koeppl[3]

Automatic Control Lab, ETH Zurich
Switzerland

Abstract

In this paper, we discuss a method for decomposition, abstraction and reconstruction of the stochastic semantics of rule-based systems with conserved number of agents. Abstraction is induced by counting fragments instead of the species, which are the standard entities of information in molecular signaling. The rule-set can be decomposed to smaller rule-sets, so that the fragment-based dynamics of the whole rule-set is exactly a composition of species-based dynamics of smaller rule-sets. The reconstruction of the transient species-based dynamics is possible for certain initial distributions. We show that, if all the rules in a rule set are reversible, the reconstruction of the species-based dynamics is always possible at the stationary distribution. We use a case study of colloidal aggregation to demonstrate that the method can reduce the state space exponentially with respect to the standard, species-based description.

Keywords: cell signaling, continuous-time Markov chain, lumpability

Introduction

Internal dependencies of multi-site post-translational modifications [22] and conformational changes [4,21] of signaling proteins reflect the rich internal logic of proteins. Since chemical kinetics [11] operates on states which are based on descriptions of full molecular complexes, often times a model becomes too complex to analyze. This calls for decomposition techniques, i.e. determining the effective dimension of the state-space. Authors in [2,5,7,9] proposed approaches where they first constructed a large state-space and then reduced it. In [14], we however took a bottom-up approach and observed the effective degrees of freedom of each agent, denoted as *agent*

⋆ Heinz Koeppl and Arnab Ganguly acknowledge the support from the Swiss National Science Foundation, grant no. $PP00P2\_128503$. Tatjana Petrov acknowledges the support from SystemsX.ch, the Swiss Initiative in Systems Biology.

[1] Email: tpetrov@control.ee.ethz.ch

[2] Email: gangulya@control.ee.ethz.ch

[3] Email: koeppl@ethz.ch

doi:10.1016/j.entcs.2012.05.018

views. This is because the language Kappa for specifying reactions allows for symbolic encoding of reactants by using site-graphs instead of structureless variables. The decomposition is performed by detecting that, for example, modifications over one site of the agents' interface never condition the state of another site on the interface of the same agent. The equivalent observation is exploited in the framework of stochastic fragments [8], [9] – where we directly, without performing decomposition, observe entities of information that are more abstract than the standard species and are called *stochastic fragments*.

To illustrate the idea behind the decomposition and stochastic fragments, consider a programming module A that contains two Boolean variables x and y. The values of variables change as a discrete-time stochastic process, so that the next value of each variable is conditioned only on the current value of that same variable. Assume that the module can be instantiated more times, and that all instances are running in parallel. For example, consider two instances of module A: A^1 and A^2. Let $Z_n \in \{(i_1,i_2,i_3,i_4) | i_1+i_2+i_3+i_4 = 2\}$ represent the state (i_1,i_2,i_3,i_4) at time n, with i_1 instances of A setting (x,y) to $(0,0)$, and i_2, i_3, i_4 instances of A setting (x,y) to $(0,1)$, $(1,0)$ and $(1,1)$ respectively. Due to the independent updates of variables x and y, we can decompose module A to two smaller modules– A_x, that contains only the updates of variable x, and A_y, that contains the updates of variable y. Let the random variables $X_n \in \{0,1,2\}$ and $Y_n \in \{0,1,2\}$ represent the number of x and y variables that are set to 1 at time $n \in \mathbb{N}$. The independence of x and y allows us to compute the correct joint probability of, for example, states with one variable x set to 1, and one variable y set to 1: $P((X,Y)_n = (1,1)) = P(X_n = 1)P(Y_n = 1)$. The sites x and y that are taking value 1 may belong to the same instance of A, that is, $Z_n = (0,1,1,0)$, or to different instances of A, that is, $Z_n = (1,0,0,1)$. Hence, $P((X,Y)_n = (1,1)) = P(Z_n \in \{(0,1,1,0),(1,0,0,1)\})$. Finally, knowing that there is one variable x set to 1, and one variable y set to 1, the conditional probability that they belong to the same instance of A is $P(Z_n = (1,0,0,1) | X_n = 1 \text{ and } Y_n = 1) = 0.5$. Along these lines, we show that the decompositions of rule-based systems [14] give rise to counting fragments [9] instead of species, and that we can effectively reconstruct information about the concrete system by only analyzing the abstract one.

We start by encoding the rule-based models and assigning the stochastic semantics to it. In Section 2, we detail how to encode the rule-based models. In Section 3, we define the fragment-based abstraction, and how to decompose the rule set into smaller independent units. In Theorem 3.6, we demonstrate how these two frameworks relate. In Section 4, we use a model of colloidal aggregation, to demonstrate that the method can exponentially reduce the state space. Finally, in Section 5, we review the practical aspects of using the fragment-based abstraction. In particular, the probability distributions over the species-based system can be reconstructed from the fragment-based abstraction for certain initial distributions. We show in Theorem 5.2 that the reconstruction is applicable on a set of reversible rules, regardless of the initial distribution, because the underlying process is a non-explosive, irreducible CTMC with a stationary distribution.

1 Preliminaries

We embed the framework of classical stochastic chemical kinetics into the formalism of labelled transition systems (LTS) [18]. The stochastic semantics of an LTS is defined as a continuous-time Markov chain (CTMC in further text).

Definition 1.1 *(Interpreted labelled transition system – ILTS)* A labelled transition system (LTS) is a tuple $M = (S, L, a)$, where

- S is a finite set of states,
- L is a finite set of labels,
- $a : S \times L \times S \to \mathbb{R}_{\geq 0}$ is the activity of a transition.

Let (X_t) be a CTMC over the state space S with the generator matrix

$$W(s, s') = \begin{cases} \sum_{l \in L} a(s, l, s') & \text{for } s \neq s' \\ -\sum_{l \in L, s'' \in S} a(s, l, s'') & \text{for } s = s'. \end{cases}$$

For any pair of states $(s, s') \in S \times S$, there will be at most one label $l \in L$, such that $a(s, l, s') > 0$, that is, the one which enables the transition from s to s'.

In order to assign a set of properties to states of a LTS, we introduce the set of Boolean variables *Var*. A property is encoded by a corresponding valuation: $x : \mathit{Var} \to \{0, 1\}$. The *interpretation function* $\mathcal{L} : S \to \wp(\mathit{Var} \to \{0, 1\})$ assigns to each state $s \in S$ a set of valuations. The *interpreted LTS* (ILTS) $M_{\mathcal{L}}$ is *well-defined*, if the sets of properties assigned to different states are disjoint: if $s \neq s'$, then $\mathcal{L}(s) \cap \mathcal{L}(s') = \emptyset$.

Two ILTS's can be composed by a cross-product operator, if their sets of labels, and their sets of variables are mutually disjoint.

Definition 1.2 *(Cross-product of two ILTS)* Consider two ILTS: $M_{1,\mathcal{L}_1} = (S_1, L_1, a_1)$ and $M_{2,\mathcal{L}_2} = (S_2, L_2, a_2)$, interpreted over a set of variables Var_1 and Var_2 respectively, such that $L_1 \cap L_2 = \emptyset$, and $\mathit{Var}_1 \cap \mathit{Var}_2 = \emptyset$. The product $M_{\mathcal{L}} = M_{1,\mathcal{L}_1} \times M_{2,\mathcal{L}_2}$ is an ILTS $M_{\mathcal{L}} = (S, L, a)$ defined over the set of variables $\mathit{Var} = \mathit{Var}_1 \uplus \mathit{Var}_2$ (the symbol $\uplus$ denotes disjoint union), such that

- $S = S_1 \times S_2$,
- $L = L_1 \uplus L_2$,
- $a((s_1, s_2), l_1, (s_1', s_2)) = a_1(s_1, l_1, s_1')$, and $a((s_1, s_2), l_2, (s_1, s_2')) = a_2(s_2, l_2, s_2')$, for $s_1, s_1' \in S_1$, $s_2, s_2' \in S_2$, $l_1 \in L_1$ and $l_2 \in L_2$.
- $\mathcal{L}((s_1, s_2)) = \mathcal{L}_1(s_1)|^{\mathit{Var}} \cap \mathcal{L}_2(s_2)|^{\mathit{Var}}$, where notation $\mathcal{L}_1(s_1)|^{\mathit{Var}}$ denotes the extension of the function $\mathcal{L}_1$ to a set of variables Var [4].

If the generators of $M_{\mathcal{L}_1}$ and $M_{\mathcal{L}_2}$ are W_1 and W_2, then the generator matrix of $M_{\mathcal{L}}$ equals their Kroenecker sum $W = W_1 \oplus W_2$ [3]. In other words, the stochastic process assigned to $M_{\mathcal{L}}$ can be seen as processes $M_{1,\mathcal{L}_1}$ and $M_{2,\mathcal{L}_2}$ running in parallel.

[4] Formally, $\mathcal{L}_1(s_1)|^{\mathit{Var}} = \{x \in (\mathit{Var} \to \{0, 1\}) \mid x|_{\mathit{Var}_1} \in \mathcal{L}_1(s_1)\}$.

We will need the notion of isomorphic LTS's when considering the generator matrices of the underlying CTMC's.

Definition 1.3 *(isomorphic LTS's)* We say that two LTS $M_1 = (S_1, L_1, a_1)$ and $M_2 = (S_2, L_2, a_2)$ are isomorphic, written $M_1 \cong M_2$, if their generators are equivalent, i.e. if there is a bijection $\alpha : S_1 \to S_2$, such that $W_1(s_1, s_1') = W_2(\alpha(s_1), \alpha(s_1'))$ for all $s_1, s_1' \in S_1$.

Definition 1.4 *(ILTS: Valid abstraction)* Consider an ILTS $M_{\mathcal{L}} = (S, L, a)$, and two equivalence relations: $\sim \subseteq S \times S$ and $\sim_l \subseteq L \times L$. The ILTS $\tilde{M}_{\tilde{\mathcal{L}}} = (\tilde{S}, \tilde{L}, \tilde{a})$ is an abstraction of $M_{\mathcal{L}}$, induced by $\sim$ and $\sim_l$, such that $\tilde{S} = S_{/\sim}$, $\tilde{L} = L_{/\sim}$, and

$$\tilde{a}([s]_\sim, [l]_{\sim_l}, [s']_\sim) = \frac{1}{|[s]_\sim|} \sum_{s \in [s]_\sim, s' \in [s']_\sim, l \in [l]_{\sim_l}} a(s, l, s').$$

The lumped state $[s]_\sim$ is interpreted by the union of interpretations of the containing states: $\tilde{\mathcal{L}}([s]_\sim) = \bigcup_{s' \in [s]_\sim} \mathcal{L}(s')$.

Let $[[s]_\sim, [l]_{\sim_l}, s']$ be the number of transitions from the class $[s]_\sim$ towards the state $s' \in S$ via the labels class $[l]_{\sim_l}$. The abstraction $\tilde{M}_{\tilde{\mathcal{L}}} = (\tilde{S}, \tilde{L}, \tilde{a})$ is *valid* if

- all lumped labels establish the same activity: if $l_1 \sim_l l_2$ and $s_1, s_1', s_2, s_2' \in S$, $a(s_1, l_1, s_1') > 0$ and $a(s_2, l_2, s_2') > 0$, then $a(s_1, l_1, s_1') = a(s_2, l_2, s_2')$;
- every two lumped states have the same total activity: if $s_1 \sim s_2$, then $\sum_{l \in L, s' \in S} a(s_1, l, s') = \sum_{l \in L, s' \in S} a(s_2, l, s')$; and
- every two lumped states are backward uniform bisimilar: if $s_1 \sim s_2$, and $s \in S$, then $[[s]_\sim, [l]_{\sim_l}, s_1] = [[s]_\sim, [l]_{\sim_l}, s_2]$.

Fix $s, s' \in S$. The activity between states $[s]_\sim$ and $[s']_\sim$ via label $[l]_{\sim_l}$ of a valid abstraction can be computed as

$$\tilde{a}([s]_\sim, [l]_{\sim_l}, [s']_\sim) = a(s, l, s')[[s]_\sim, [l]_{\sim_l}, s'] \frac{|[s']_\sim|}{|[s]_\sim|}.$$

The condition imposed for an abstraction to be valid is known in the literature as a form of weak lumpability [13], or uniform backward bisimulation. In Section 2.1, the ILTS assigned to a rule-based model is such that each state is interpreted by exactly one valuation of variables from *Var*. The lumped state $[s]_\sim$ is interpreted by a union over interpretations of containing states.

The following Lemma suggests a criterion for showing that an abstraction is valid.

Lemma 1.5 *(Valid abstraction) Given an ILTS $M_{\mathcal{L}} = (S, L, a)$, and two equivalence relations: $\sim \subseteq S \times S$ and $\sim_l \subseteq L \times L$, if (i) all lumped labels establish the same activity, (ii) every two lumped states have the same total activity, and (iii) for $s, s_1, s_2 \in S$, such that $s_1 \sim s_2$, there is a bijection between the sets of predecessors of $s_1 (s_2)$ in the class $[s]_\sim$, via labels from the class $[l]_\sim$, then the abstraction $\tilde{M}_{\tilde{\mathcal{L}}} = (\tilde{S}, \tilde{L}, \tilde{a})$ is valid.*

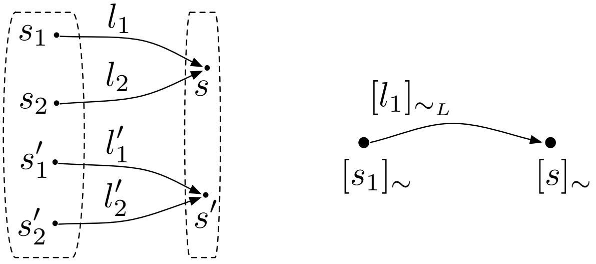


Fig. 1. Valid abstraction of LTS. (left) The LTS $M = (S,L,a)$, such that $S = \{s,s',s_1,s_1',s_2,s_2'\}$, $L = \{l_1,l_2,l_1',l_2'\}$. and $a(s,l_1,s_1) = a(s,l_2,s_2) = a(s',l_1',s_1') = a(s',l_2',s_2') = k$. The abstraction is induced by $S_{/\sim} = \{\{s,s'\},\{s_1,s_2,s_1',s_2'\}\}$ and $L_{/\sim_l} = \{\{l_1,l_2,l_1',l_2'\}\}$. (right) Since the condition from Definition 1.4 is met, the abstraction is valid, and the activity between lumped states is $a([s_1]_\sim,[l_1]_{\sim_l},[s]_\sim) = 2\frac{2}{4}k = k$.

The proof is obvious from the Dfn. 1.4.

2 Rule-based models: Boolean encoding and semantics

In this section, a rule-based system is defined, and it is shown how to associate an ILTS to it. The model is built over a set of agent names $\mathcal{A}$ and a set of site names $\mathcal{S}$. Agents typically model proteins, and sites model protein domains. Each agent has an *interface*, that is a set of sites – $\Sigma : \mathcal{A} \to \wp(\mathcal{S})$. Sites can be internal or binding sites, but not both: $\Sigma = \Sigma_i \uplus \Sigma_l$. Each site is assumed to be in one of the two modification states, denoted by 0 and 1. In particular, a binding site has a bond if and only if its modification state is 1. We use *site graphs* to formalize the model.

Definition 2.1 *(Site graph)* Site graph is a pair $G = (\mathcal{V},\mathcal{E})$, such that $\mathcal{V} \subseteq \{(A,\Sigma_i,\Sigma_l) \mid A \in \mathcal{A} \text{ and } \Sigma_i,\Sigma_l \subseteq \wp(\mathcal{S})\}$ is a set of nodes, and the set of pairs of binding sites, $\mathcal{E} \subseteq \{((A,s),(A',s')) \mid (A,\Sigma_i,\Sigma_l),(A',\Sigma_i{}',\Sigma_l') \in \mathcal{V}, s \in \Sigma_l, s' \in \Sigma_l'\}$, is a set of edges. The set of edges is a symmetric relation.

Definition 2.2 *(Annotated site graph) Annotated site graph* $(\mathcal{V},\mathcal{E})_\equiv$ is a site graph $(\mathcal{V},\mathcal{E})$, with an equivalence relation $\equiv$ over the agent-site pairs: $\equiv \subseteq \{((A,s),(A',s')) | A,A' \in \mathcal{A}, s \in \Sigma(A), s' \in \Sigma(A')\}$.

In rule-based modelling, we use site-graphs to formalize different kinds of objects (we define each of these objects formally in Sec. 2.1):

- a *contact map* is a site graph which summarizes the protein names and their possible bindings [6];
- an *annotated contact map* is an annotated site graph; two sites are grouped by the annotation relation, to formalize that their values depend on each-other (are correlated) in the behaviour of interest (which is stochastic chemical kinetics in this paper);
- A *reaction mixture map* is used for encoding one state of the system, i.e. the whole reaction mixture. It is a site-graph constructed from the contact map, by copying nodes and edges a given number of times.

2.1 Encoding reaction mixtures

Let *Var* be a set of variables assigned to each site of a given contact map:

$$Var \cong \{(A,v) | A \in \mathcal{A}, v \in \mathcal{S}\} \cup \mathcal{E}.$$

A variable that encodes the value of site v of agent A is denoted by v_A. A variable that encodes the bond between sites u and v of agents A and B is denoted by $uv_{(A,B)}$. There can be more copies of each agent in the system: assume $n(A)$ copies of agent A, then $n(B)$ copies of agent B etc. Each copy of the agent is identified by a number in its superscript: $A^1, A^2, \ldots$ A bond may exist between two sites over identified agents – (A^i, v) and (A^j, v') only if it exists between corresponding sites (A, v) and (A', v') in the contact map.

Formally, a reaction mixture map over a contact map $(\mathcal{V}, \mathcal{E})$ is a site graph $(\mathcal{V}^n, \mathcal{E}^n)$ with the set of identified agent names $\mathcal{A}^n$ and site names $\mathcal{S}$, such that

$$\mathcal{V}^n = \{(A^i, \Sigma_i(A), \Sigma_l(A)) \mid A \in \mathcal{A} \text{ and } i = 1, ..., n(A)\},$$

and the set of edges is

$$\mathcal{E}^n = \{((A^i, v), (A'^j, v')) \mid A, A' \in \mathcal{A} \text{ and } i = 1, \ldots, n(A), j = 1, \ldots, n(A')\}.$$

Let Var^n be a set of variables assigned to each site of a full contact map:

$$Var^n \cong \{(A^i, v) | A^i \in \mathcal{A}^n, v \in \mathcal{S}\} \cup \mathcal{E}^n.$$

We denote by v_A^i the variable assigned to the site v of agent A^i, and by $uv_{(A,B)}^{i,j}$ the variable assigned to the bond between sites u and v of agents A^i and B^j. One valuation of the variables from Var^n encodes one *reaction mixture*. Each state of the ILTS corresponds to one reaction mixture: $\mathcal{L}(s) \in (Var^n \to \{0, 1\})$. The ILTS counts as many states as there are valuations over Var^n, i.e. $S \cong (Var^n \to \{0, 1\})$, although typically only a small subset of them is reachable.

2.2 Encoding transitions

The dynamics of a rule-based model is given by a set of rules. A classical chemical reaction consists of a left-hand-side (lhs), a right-hand-side (rhs), and a rate. The lhs is a set of reactants, which can transform to a set of products, and the transformation occurs at a velocity depending on the reaction rate. Similarly, the lhs and rhs of a rule are sets of agents with different values of sites in their interfaces.

Assumptions. The rule-based model we present here is inspired by a rule-based modelling framework Kappa [6], but we restrict to the following assumptions:

(i) An agent appears at most once in a rule;

(ii) An agent cannot be created or deleted by a rule.

Consider the set of propositional formulae $\mathcal{P}$ over variables *Var* generated by the grammar $p \equiv 0 \mid 1 \mid a \in Var \mid \neg p \mid p \wedge p$. We denote by Var_p a subset of variables *Var* that occur in the proposition p. The satisfaction region of the formula p is denoted by $[\![p]\!] = \{x \mid \text{state } x \text{ satisfies proposition } p\}$.

Definition 2.3 *(Rule)* A *rule* is a triple $(p, q, k) \in \mathcal{P} \times \mathcal{P} \times \mathbb{R}_0$ such that $Var_p = Var_q$.

Definition 2.4 *(Rule-based system)* A rule-based system $\mathcal{B} = (\mathcal{V}, \mathcal{E}, n, \mathcal{R})$ is defined by (i) a reaction mixture map $(\mathcal{V}^n, \mathcal{E}^n)$, and (ii) a set of rules $\mathcal{R} = \{R_1, \ldots, R_m\}$.

Two sites in a contact map are in *stochastic annotation* if they both appear in some rule.

Definition 2.5 *(Contact map: stochastic annotation, [8])* Given a rule-based system $\mathcal{B} = (\mathcal{V}, \mathcal{E}, n, \mathcal{R})$ over a contact map $(\mathcal{V}, \mathcal{E})$, its stochastic annotation is the least reflexive and symmetric relation $\equiv \subseteq \{((A, v), (A', v')) | A, A' \in \mathcal{A}, v, v' \in \mathcal{S}\}$, such that

- each two sites that form an edge are correlated: $\mathcal{E} \subseteq \equiv$, and
- if $R = (p, q, k) \in \mathcal{R}$ and $\{v_A, v'_A\} \subseteq Var_p$, then $((A, v), (A, v')) \in \equiv$, and
- the restriction of $\equiv$ to sites of the same agent is transitive: for any agent $A \in \mathcal{A}$, such that $v, v', v'' \in \Sigma(A)$, if $((A, v), (A, v')) \in \equiv$ and $((A, v), (A, v'')) \in \equiv$, then $((A, v'), (A, v'')) \in \equiv$.

Let us now consider a reaction mixture map $(\mathcal{V}^n, \mathcal{E}^n)$ and let $\equiv_s \subseteq Var^n \times Var^n$ be the least equivalence relation such that $(A^i, v) \equiv_s (B^j, v')$ if $[A = B,\ i = j$ and $(A, v) \equiv (A, v')]$ or $[A \neq B$ and $(A, v) \equiv (B, v')]$. The equivalence classes $[v_A]_{\equiv_s} \in Var_{/\equiv_s}$, that are induced by the stochastic annotation on the set of variables, are called *stochastic fragments.*

Rules are defined over the set of variables Var. On the other hand, a reaction mixture is defined over the set of variables Var^n. The application of a rule to a reaction mixture is formalized through a concept of agent identification.

Definition 2.6 *(Agent identification)* The agent identification function $\nu : Var \to Var^n$ assigns to each agent's variable an identified version of it, in such a way that one agent's site variables are mapped to that agent's same identified version: if $u, v \in \Sigma(A)$, and $\nu(u) \in \Sigma(A^i)$, then also $\nu(v) \in \Sigma(A^i)$. Given a proposition $p \in \mathcal{P}$ over the set of variables Var, the same proposition with variables renamed by agent identification function ν is denoted by $p[/\nu]$. The state s, interpreted by $\mathcal{L}(s)$, *satisfies* the lhs of the rule, if for some identification function ν, it holds that $\mathcal{L}(s) \in [\![p[/\nu]]\!]$.

For example, if a variable $v_A \in Var$ denotes value of site v in agent A, it can be identified by using instead a variable $\nu(v_A) = v_A^1 \in Var^n$. After agent identification, a proposition $p = \neg v_A$ becomes $p[/\nu] = \neg v_A^1$. A bond variable $uv_{(A,B)} \in Var$ can be identified by a variable $\nu(uv_{(A,B)}) = uv_{(A,B)}^{2,4} \in Var^n$. The proposition $p' = \neg u_A \wedge v_A$ becomes, for example, $p'[/\nu] = \neg u_A^3 \wedge v_A^3$. Note that, by definition, an identification such as $p'[/\nu] = \neg u_A^3 \wedge v_A^5$ is impossible.

Application of a rule to a reaction mixture can be done, if after some agent identification, the lhs of the rule is satisfied by that reaction mixture's interpretation function. After the rule application, the reaction mixture is updated accordingly, so as to satisfy the rhs of the rule. The transition is labelled by the name of the rule accompanied with the identification function.

Definition 2.7 *(ILTS of a rule-based system)* Given a rule-based system $\mathcal{B} = (\mathcal{V}, \mathcal{E}, n, \mathcal{R})$, the ILTS $M_{\mathcal{L}} = (S, L, a)$ assigned to the set of rules $\mathcal{R}$ in interpretation $\mathcal{L}$, written also $M_{\mathcal{L}} \vDash \mathcal{R}$, is defined by

- a state space $S = \{s_1, s_2, \ldots\} \cong (Var^n \to \{0,1\})$,
- set of labels $L = \{(R, \nu) |$ rule $R = (p, q, k) \in \mathcal{R}$ and identification $\nu : Var_p \to Var^n\}$,
- for any two states $s, s' \in S$, a rule $R = (p, q, k)$, and an identification function $\nu : Var \to Var^n$, the activity of transition from state s to state s' via label (R, ν) is equal to k, i.e. $a(s, (R, \nu), s') = k$, if and only if:

(i) $\mathcal{L}(s) \in [\![p[/\nu]]\!]$, i.e. state s satisfies the lhs of the rule,
(ii) $\mathcal{L}(s') \in [\![q[/\nu]]\!]$, i.e. state s' satisfies the rhs of the rule, and
(iii) if no variable gets identified to a site $v \in Var^n$, then its value remains unchanged after the rule application.

The defined ILTS has dynamics which coincides with the standard way of defining stochastic chemical kinetics over a continuous-time Markov chain [11],[1],[8].

Example 2.8 Consider the following set of rules:

$$\begin{aligned} R_1 &: \neg x_A \to x_A \ \ (k_1) \\ R_1^- &: x_A \to \neg x_A \ \ (k_1^-) \\ R_2 &: \neg y_A \to y_A \ \ (k_2) \\ R_2^- &: y_A \to \neg y_A \ \ (k_2^-). \end{aligned}$$

where $R : p \to q \ \ (k)$ denotes a rule $R = (p, q, k)$. If there are two copies of agent A, i.e. $n(A) = 2$, there are two different agent identifications for the rule R_1:

$$\begin{aligned} (R_1, A \mapsto A^1) &: \neg x_A^1 \to x_A^1 \ \ (k_1), \text{ and} \\ (R_1, A \mapsto A^2) &: \neg x_A^2 \to x_A^2 \ \ (k_1). \end{aligned}$$

The contact map is a site graph $(\mathcal{V}, \mathcal{E})$ with agent names $\mathcal{A} = \{A\}$ and site names $\mathcal{S} = \{x, y\}$; Set of nodes is given by $\mathcal{V} = \{(A, \{x, y\}, \{\})\}$, and edges $\mathcal{E} = \{\}$. The set of variables associated to the rule set is $Var = \{x_A, y_A\}$. Since no rule involves both variables x_A and y_A, the stochastic fragments are $Var_{/\equiv_s} = \{\{x_A\}, \{y_A\}\}$.

For $n(A) = 2$, the reaction mixture map is a site-graph $(\mathcal{V}^n, \mathcal{E}^n)$, where $\mathcal{V}^n = \{A^1, A^2\}$, and $\mathcal{E}^n = \{\}$. For $n(A) = 2$, the set of variables to encode one state of a CTMC is $Var^n = \{x_A^1, y_A^1, x_A^2, y_A^2\}$. Therefore, interpreting state $s \in S$ of ILTS $M_{\mathcal{L}} = (S, L, a)$ assigned to the rule set $\mathcal{R}$ is such that $\mathcal{L}(s) \in (\{x_A^1, y_A^1, x_A^2, y_A^2\} \to \{0,1\}) \cong \{0,1\}^4$. For example, the state s, with $\mathcal{L}(s) = (0,0,0,0)$, denotes the mixture where all site values are set to 0. A part of the ILTS that models rule-based system of this example is shown in Fig.2d.

Example 2.9 Consider the following set of rules:

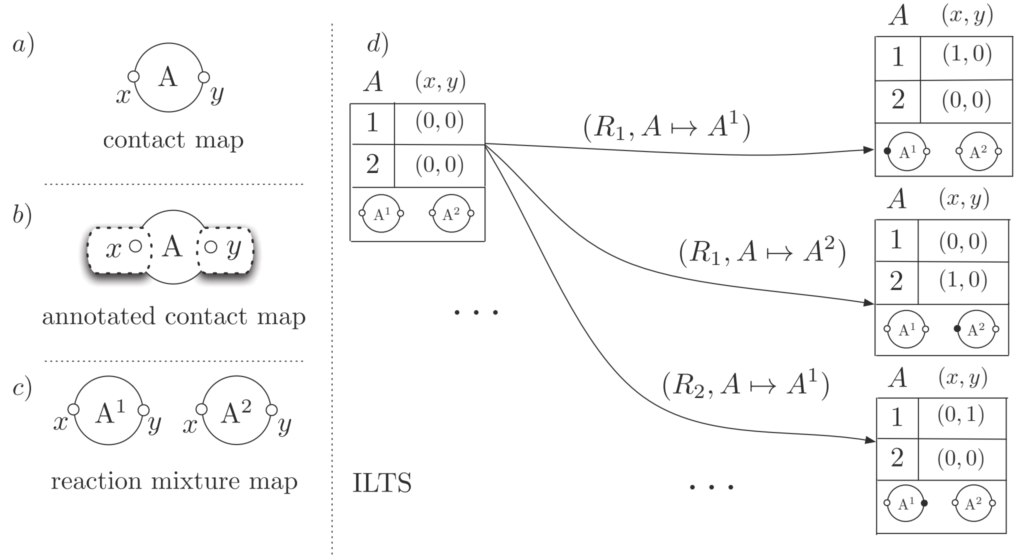
$$
\begin{aligned}
R_1 &: \neg b_A, \neg a_B, \neg ba_{(A,B)} \to b_A, a_B, ba_{(A,B)} \ (k_1)\\
R_1^- &: b_A, a_B, ba_{(A,B)} \to \neg b_A, \neg a_B, \neg ba_{(A,B)} \ (k_1^-)\\
R_2 &: \neg c_B, \neg b_C, \neg cb_{(B,C)} \to c_B, b_C, cb_{(B,C)} \ (k_2)\\
R_2^- &: c_B, b_C, cb_{(B,C)} \to \neg c_B, \neg b_C, \neg cb_{(B,C)} \ (k_2^-).
\end{aligned}
$$

The contact map is a site graph $(\mathcal{V}, \mathcal{E})$ with agent names $\mathcal{A} = \{A, B, C\}$ and site names $\mathcal{S} = \{b, a, c\}$; Set of nodes is $\mathcal{V} = \{(A, \{\}, \{b\}), (B, \{\}, \{a, c\}), (C, \{\}, \{b\})\}$, and the set of edges $\mathcal{E}$ is the symmetric closure of the set $\{((A, b), (B, a)), ((B, c), (C, b))\}$.

For $n(A) = 1$, $n(B) = 2$ and $n(C) = 1$, the set of identified agents is $\mathcal{A} = \{A^1, B^1, B^2, C^1\}$, the reaction mixture map is a site-graph $(\mathcal{V}^n, \mathcal{E}^n)$, where $\mathcal{V}^n = \{(A^1, \{b\}, \{\}), (B^1, \{\}, \{a, c\}), (B^2, \{\}, \{a, c\}), (C^1, \{\}, \{b\})\}$. The set of associated variables is $Var = \{b_A, a_B, ab_{(A,B)}, c_B, b_C, bc_{(B,C)}\}$. Interpreting state $s \in S$ of ILTS $M_{\mathcal{L}} = (S, L, a)$ assigned to the rule set $\mathcal{R}$ is a function

$$\mathcal{L}(s) \in (\{b_A^1, a_B^1, a_B^2, ab_{(A,B)}^{1,1}, ab_{(A,B)}^{1,2}, c_B^1, c_B^2, b_C^1, b_C^2, bc_{(B,C)}^{1,1}, bc_{(B,C)}^{2,1}\} \to \{0, 1\}).$$

For example, the state s, such that $\mathcal{L}(s) = (1, 1, 0, 1, 0, 1, 0, 1, 1, 0)$ encodes the mixture with one complex between agents A^1, B^1 and C^1, and a free B^2 agent. The stochastic fragments are $Var_{/\equiv_s} = \{\{b_A, a_B, ab_{(A,B)}\}, \{c_B, b_C, bc_{(B,C)}\}\}$.



A	(x, y)
1	$(0, 0)$
2	$(0, 0)$

A	(x, y)
1	$(1, 0)$
2	$(0, 0)$

A	(x, y)
1	$(0, 0)$
2	$(1, 0)$

A	(x, y)
1	$(0, 1)$
2	$(0, 0)$

Fig. 2. Illustration for Example 2.8: a) contact map (CM), b) annotated contact map (ACM), c) reaction mixture map, for $n(A) = 2$, d) A part of the ILTS assigned to the rule-based system.

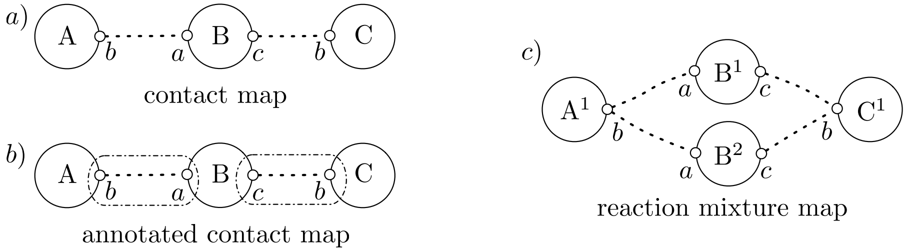


Fig. 3. Illustration for Example 2.9: a) contact map (CM), b) annotated contact map (ACM), c) reaction mixture map, for $n(A) = 1$, $n(B) = 2$, $n(C) = 1$.

3 From model decomposition to stochastic fragments

Let $M_\mathcal{L}$ be an ILTS of a rule-based system $\mathcal{B} = (\mathcal{V}, \mathcal{E}, n, \mathcal{R})$. We introduce two valid abstractions of $M_\mathcal{L}$: (i) a species-based abstraction, that is a standard level of observation in classical chemical kinetics, and (ii) a fragment-based abstraction, specific to rule-based models, first introduced in [9,8]. Both abstractions are induced by the following labels' lumping:

Definition 3.1 *(Lumping labels)* Two labels $l_1, l_2 \in L$ are lumped by relation $\sim_l \subseteq L \times L$ if and only if they are created by identifying agents of the same rule: given a rule $R \in \mathcal{R}$, and two identification functions $\nu_1, \nu_2 : Var \to Var^n$, it holds that $l_1 = (R, \nu_1) \sim_l (R, \nu_2) = l_2$.

Species-based abstraction is done by lumping the states, which are equivalent up to permutation over agents' identifiers. For example, if a state has one agent A with interface $(0,0)$, and one agent A with interface $(1,1)$, it does not matter which interface is of agent A^1, and which of A^2 (for example, see states s_1 and s_2 in Fig.4).

Definition 3.2 Let $\{\sigma_A \in (\{1,\dots n(A)\} \to \{1,\dots n(A)\}\}_{A\in\mathcal{A}}$ be a family of permutations over the set $\{1,\dots n(A)\}$, identifiers of agent A. Each such family of permutations induces another permutation, over the set of variables Var^n, $\Phi_\sigma : Var^n \to Var^n$:

$$\Phi_\sigma(w) = \begin{cases} v_A^{\sigma_A(i)} & \text{if } w = v_A^i \\ uv_{(A,B)}^{\sigma_A(i),\sigma_B(j)} & \text{if } w = uv_{(A,B)}^{i,j}. \end{cases}$$

Definition 3.3 *(Species-based abstraction)* The states s and s' are lumped, i.e. $s{\sim_p}s'$, if there exists a family of permutations over its identifiers: $\{\sigma_A \in (\{1,\dots n(A)\} \to \{1,\dots n(A)\}\}_{A\in\mathcal{A}}$, such that

$$\text{for all } u \in Var^n, \text{ it holds that } \mathcal{L}(s)(u) = \mathcal{L}(s')(\Phi_\sigma(u)).$$

Let the equivalence relation $\sim_p \subseteq S \times S$ be the transitive closure of $\sim_p$. The species-based abstraction $M^p_{\mathcal{L}_p} = (S^p, L^p, a^p)$ is an abstraction of $M_\mathcal{L}$ induced by lumping states with $\sim_p$ and lumping labels with $\sim_l$.

In fragment-based abstractions, two states are lumped if we can permute the identifiers of agents, so that the parts (fragments) of their interfaces match site values. In Example 2.8, there are two fragments: $\mathit{Var}_{/\equiv_s} = \{\{x_A\},\{y_A\}\}$. A state with agent A^1 of interface $(0,0)$ and agent A^2 of interface $(1,1)$ (state s_1 in Fig.3), and an agent A^1 of interface $(0,1)$ and agent A^2 of interface $(1,0)$ (state s_3 in Fig.3) are lumped by relation $\sim_f$.

Definition 3.4 *(Fragments-based abstraction)* Let $(\mathcal{V},\mathcal{E})_{\equiv}$ be the contact map of a rule-based system $\mathcal{B} = (\mathcal{V},\mathcal{E},n,\mathcal{R})$, with stochastic annotation $\equiv$ which induces a set of stochastic fragments $\mathit{Var}_{/\equiv_s} = \{\mathit{Var}_1,\ldots,\mathit{Var}_l\}$. The states s and s' are lumped, i.e. $s{\sim_f}s'$, if there exist l families of permutations over its identifiers, $\{\sigma^1_A\}_{A\in\mathcal{A}}$, $\{\sigma^2_A\}_{A\in\mathcal{A}}$, $\ldots$, $\{\sigma^l_A\}_{A\in\mathcal{A}}$, such that for $i=1,2,\ldots,l$,

$$\text{for all } u \in \mathit{Var}_i^n\text{, it holds that } \mathcal{L}(s)(u) = \mathcal{L}(s')(\Phi_{\sigma^i}(u)).$$

Let the equivalence relation $\sim_f \subseteq S\times S$ be a transitive closure of $\sim_f$. The fragment-based abstraction $M^f_{\mathcal{L}_f} = (S^f, L^f, a^f)$ is an abstraction of $M_{\mathcal{L}}$ induced by lumping of the states and labels with $\sim_f$ and $\sim_l$ respectively.

Lemma 3.5 *Abstractions $M^p_{\mathcal{L}_p}$ and $M^f_{\mathcal{L}_f}$ are valid. Moreover, equivalence relation $\sim_f$ is coarser than $\sim_p$: $\sim_p \subseteq \sim_f$.*

Proof. We first show that the abstraction $M^p_{\mathcal{L}_p}$ is valid. The proof for showing that the abstraction $M^f_{\mathcal{L}_f}$ is valid is similar. Consider two species-based states $[s_1]_{\sim_p},[s_2]_{\sim_p} \in S_{/\sim_p}$, and a rule $R\in\mathcal{R}$ which can be applied to some state in $[s_1]_{\sim_p}$. Let $s_2,s_2' \in [s_2]_{\sim_p}$. By Dfn. 1.4, it suffices to show that there is a bijection between the set of applications of rule R from a lumped state $[s_2]_{\sim_p}$ towards a state s_2 and the set of applications of rule R from a lumped state $[s_2]_{\sim_p}$ towards state s_2'. By Dfn.3.3, there exists a permutation of agent identifiers $\sigma_A : \{1,\ldots n(A)\}\to\{1,\ldots n(A)\}$, such that for all $u\in \mathit{Var}^n$, it holds that $\mathcal{L}(s_2)(u) = \mathcal{L}(s_2')(\Phi_\sigma(u))$.

Let $s_1 \in [s_1]_{\sim_p}$ be a state such that $a(s_1,(R,\nu),s_2) = k$, that is, application of the rule $R=(p,q,k)$ can be done on state s_1 via agent identification function $\nu : \mathit{Var}\to \mathit{Var}^n$. By Dfn. 2.7, it means that $\mathcal{L}(s_1)\in [\![p[/\nu]]\!]$, $\mathcal{L}(s_2)\in [\![q[/\nu]]\!]$, and all variables that are not identified keep the same values: if $v\in \mathit{Var}^n \setminus \{\nu(u) | u\in \mathit{Var}\}$, then $\mathcal{L}(s_1)(v) = \mathcal{L}(s_2)(v)$.

Let the identification function $\nu' : \mathit{Var}\to\mathit{Var}^n$ be such that it first maps a site u by function ν, and then permutes the identifiers by function Φ_σ:

$$\nu'(v) = \Phi_\sigma(\nu(v)).$$

Let s_1' be such that $\mathcal{L}(s_1')\in[\![p[/\nu']]\!]$ and all variables that are not identified by ν' keep the same values: if $v\in\mathit{Var}^n\setminus\{\nu'(u)|u\in\mathit{Var}\}$, then $\mathcal{L}(s_1')(v) = \mathcal{L}(s_2')(v)$. Then, $a(s_1',(R,\nu'),s_2') = k$. Moreover, $s_1{\sim_p}s_1'$, because, by construction of the function ν', for all $u\in\mathit{Var}^n$, it holds that $\mathcal{L}(s_1)(u) = \mathcal{L}(s_1')(\Phi_\sigma(u))$.

Now we show that $\sim_p \subseteq \sim_f$. Take two states $s,s'\in S$ and assume that $s{\sim_p}s'$. Then, by Dfn.3.3, there exists a permutation of agent identifiers $\sigma_A : \{1,\ldots n(A)\}\to$

$$s_1 = \begin{array}{|c|c|} \multicolumn{1}{c}{A} & \multicolumn{1}{c}{(x,y)} \\ \hline 1 & (0,0) \\ \hline 2 & (1,1) \\ \hline \end{array} \qquad s_2 = \begin{array}{|c|c|} \multicolumn{1}{c}{A} & \multicolumn{1}{c}{(x,y)} \\ \hline 1 & (1,1) \\ \hline 2 & (0,0) \\ \hline \end{array} \qquad s_3 = \begin{array}{|c|c|} \multicolumn{1}{c}{A} & \multicolumn{1}{c}{(x,y)} \\ \hline 1 & (0,1) \\ \hline 2 & (1,0) \\ \hline \end{array}$$

$$\mathcal{L}(s_1) = (0,0,1,1) \quad \mathcal{L}(s_2) = (1,1,0,0) \quad \mathcal{L}(s_3) = (0,1,1,0)$$

Fig. 4. Three states $s_1, s_2, s_3 \in S$, and their interpretations. States s_1 and s_2 are lumped in species-based abstraction, i.e. $s_1{\sim_p}s_2$. The witness permutation is $\sigma(1) = 2, \sigma(2) = 1$. This is because $\mathcal{L}(s_1)(x_A^1) = \mathcal{L}(s_2)(x_A^2)(= 0)$, $\mathcal{L}(s_1)(y_A^1) = \mathcal{L}(s_2)(y_A^2)(= 0)$, $\mathcal{L}(s_1)(x_A^2) = \mathcal{L}(s_2)(x_A^1)(= 1)$ and $\mathcal{L}(s_1)(y_A^2) = \mathcal{L}(s_2)(y_A^1)(= 1)$. Stochastic annotation gives classes of variables $Var_{/\equiv_s} = \{\{x\},\{y\}\}$. Species s_1 and s_3 are not lumped in the species-based abstraction, i.e. $(s_1, s_3) \notin {\sim_p}$, but they are lumped in the fragment-based abstraction: $s_1{\sim_f}s_3$. The permutations $\sigma_{\{x\}}(1) = 2, \sigma_{\{x\}}(2) = 1$ and $\sigma_{\{y\}}(1) = 1, \sigma_{\{y\}}(2) = 2$ justify lumping states s_1 and s_3 by relation $\sim_f$.

$\{1,\dots n(A)\}$, such that for all $u \in Var^n$, it holds that $\mathcal{L}(s_1)(u) = \mathcal{L}(s_1')(\Phi_\sigma(u))$. Take

$$\{\sigma_A^1\}_{A\in\mathcal{A}} = \{\sigma_A^2\}_{A\in\mathcal{A}} = \ldots = \{\sigma_A^l\}_{A\in\mathcal{A}} = \{\sigma_A\}_{A\in\mathcal{A}}$$

Then it holds that for all $u \in Var_i^n$, $\mathcal{L}(s)(u) = \mathcal{L}(s')(\Phi_{\sigma^1}(u))$, for $i = 1,\dots,l$, and hence $s_2{\sim_f}s_2'$. □

We now show a complementary viewpoint to the fragment-based abstraction: it is a result of a particular composition operator over the species-based abstractions of appropriately chosen smaller sets of rules. More specifically, the ILTS $M_\mathcal{L}$ can be represented as a cross-product of smaller ILTS such that each of the small ILTS is assigned to a subset of rules. To do so, each of the two chosen subsets of rules must be independent, in the sense that they operate on mutually disjoint sets of sites. Finally, we show that the fragment-based abstraction is a cross-product of species-based abstractions of smaller ILTS's. The theorem is illustrated in Fig.5.

Theorem 3.6 *(Decomposing ILTS, Prop. 4.2, [14] extended)* Let $M_\mathcal{L}$ be the ILTS assigned to a rule-based system $\mathcal{B} = (\mathcal{V},\mathcal{E},n,\mathcal{R})$, and $\mathcal{R} = \mathcal{R}_1 \uplus \ldots \uplus \mathcal{R}_m$ the largest partitioning of the rule-set to smaller ones, such that each two subsets of rules have mutually disjoint sets of variables. Then, $M_\mathcal{L}$ can be decomposed in the following form:

$$M_\mathcal{L} = M_{1\mathcal{L}_1} \times \ldots \times M_{m\mathcal{L}_m},$$

where ILTS $M_{i,\mathcal{L}_i}$ $(1 \le i \le m)$ is an ILTS assigned to a set of rules $\mathcal{R}_i$. Moreover, fragment-based abstraction of $M_{\mathcal{L}_f}^f$ is isomorphic to a cross-product of the species-based abstractions of all the smaller ILTS's – $M_{i\mathcal{L}_i}^p$'s:

$$M_{\mathcal{L}_f}^f \equiv M_{1\mathcal{L}_1}^p \times \ldots \times M_{m\mathcal{L}_m}^p.$$

Lemma 3.7 *Let $M_\mathcal{L} = (S,L,a)$, $M_{\mathcal{L}_f}^f = (S^f,L^f,a^f)$, and $M_{i\mathcal{L}_i}^p = (S_i^p,L_i^p,a_i^p)$. Let $v_i \in Var$ be a variable involved in the subset of rules $\mathcal{R}_i$. The partition class $[v_i]_{\equiv_s} = Var_i \subseteq Var$ contains exactly the set of variables that appear in the subset of rules $\mathcal{R}_i$.*

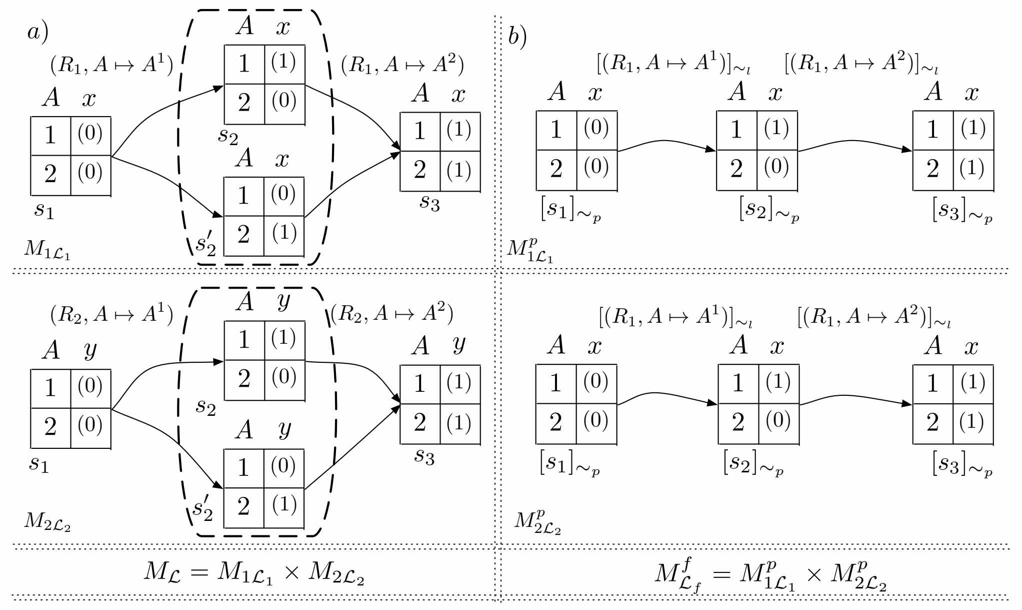


Fig. 5. Decomposition for Example 2.8: rule-set $\mathcal{R} = \{R_1, R_1^-, R_2, R_2^-\}$ is split into two smaller of sets of rules: $\mathcal{R}_1 = \{R_1, R_1^-\}$ and $\mathcal{R}_2 = \{R_2, R_2^-\}$. a) An ILTS $M_{1\mathcal{L}_1}$ assigned to rule-set $\mathcal{R}_1$, and an ILTS $M_{2\mathcal{L}_2}$ assigned to rule-set $\mathcal{R}_2$. b) The species-based projection $M^p_{1\mathcal{L}_1}$; the population-based projection $M^p_{2\mathcal{L}_2}$. The Theorem 3.6 states that $M^f_{\mathcal{L}_f} = M^p_{1\mathcal{L}_1} \times M^p_{2\mathcal{L}_2}$.

Proof. Recall that for any two variables $u, v \in \mathit{Var}$, they are correlated by relation $\equiv_s$, i.e. $u \equiv_s v$ if and only if they belong to the same subset of rules, e.g. $\mathcal{R}_i$. Due to the Dfn. 2.5, for some rule $R = (p, q, k)$, either $u, v \in \mathit{Var}_p$, or because of the transitive closure, there is a sequence of rules $R_1 = (p_1, q_1, k_1), ..., R_l = (p_l, q_l, k_l)$, such that $u \in \mathit{Var}_{p_1}$ and $v \in \mathit{Var}_{p_l}$, but $\mathit{Var}_{p_i} \cap \mathit{Var}_{p_{i+1}} \neq \varnothing$ for $i = 1, \ldots, l-1$. Conversely, if two variables $u, v \in \mathit{Var}$ appear in the subset of rules $\mathcal{R}_i$, then $u \equiv_s v$, because the partitioning $\mathcal{R} = \mathcal{R}_1 \uplus \ldots \uplus \mathcal{R}_m$ is assumed to be the *largest* one where subsets of rules do not share variables. □

Proof. (Thm. 3.6) Let $\mathit{Var}_i^n \subseteq \mathit{Var}$ be the set of variables that identify agents from the set Var_i, and thus $\mathit{Var}^n = \mathit{Var}_1^n \uplus \ldots \uplus \mathit{Var}_l^n$. Consider the function $\alpha : S \to S_1 \times \ldots \times S_m$ that takes a state s with valuation $\mathcal{L}(s)$, and picks the states $s_1 \in S_1, ..., s_m \in S_m$ which have the corresponding valuations of subsets of variables $\mathit{Var}_1^n, \ldots, \mathit{Var}_l^n$: for all $v \in \mathit{Var}_i$, let $\mathcal{L}_i(s_i)(v) = \mathcal{L}(s)(v)$. Then,

$$s \sim_f s' \text{ if and only if } s_1 \sim_p s'_1, s_2 \sim_p s'_2, \ldots s_m \sim_p s'_m,$$

because the i-th witness family of permutations $\{\sigma_A^i\}_{A \in \mathcal{A}}$ for $s \sim_f s'$ is exactly the witness family of permutations for $s_i \sim_p s'_i$. Hence, the function $\tilde{\alpha} : S^f \to S_1^p \times \ldots \times S_m^p$

$$\tilde{\alpha}([s]_{\sim_f}) = ([s_1]_{\sim_p}, ..., [s_m]_{\sim_p}) \text{ so that } s = \alpha(s_1, \ldots, s_m).$$

is a well-defined bijective function.

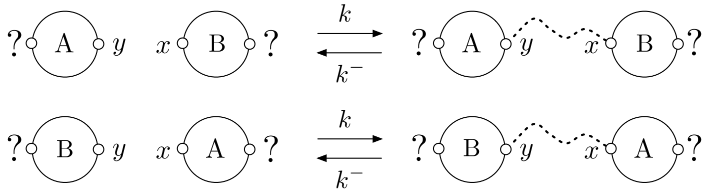


Fig. 6. Example 4.1–colloidal aggregation model.

Moreover, we show that the rate between states $s^f, s^{f'} \in S^f$ is equal to the rate between the states $\tilde{\alpha}(s^f), \tilde{\alpha}(s^{f'}) \in S_1^p \times ... \times S_m^p$. Let $[s]_{\sim_f}, [s']_{\sim_f} \in S^f$, a rule $R \in \mathcal{R}_i$, and assume that for some identification function $\nu : Var \to Var^n$, we have that $a(s,(R,\nu),s') = k$. The transition with label (R,ν) will happen only in the ILTS $M^p_{i\mathcal{L}_i}$, because rules from other classes do not intersect with the variables from $\mathcal{R}_i$, and thus Var_p. Therefore, $a([s]_{\sim_f}, [(R,\nu)]_{\sim_l}, [s']_{\sim_f}) = a_i(([s_1]_{\sim_p}, \ldots, [s_i]_{\sim_p}, \ldots, [s_l]_{\sim_p}), [(R,\nu)]_{\sim_l}, ([s_1]_{\sim_p}, \ldots, [s_i']_{\sim_p}, \ldots, [s_l]_{\sim_p}))$. □

4 Case study: Colloidal aggregation

In the following, we illustrate the framework for a simple model of colloidal aggregation [15]. Such aggegration dynamics represent the simplest form of self-assembly – a process ubiquitous in molecular cell biology. Microtubuli assembly [10], actin filament polymerization [19] or prion replication [20] fall into this class - to name but a few. With this case-study, we demonstrate that using a fragment-based abstraction instead of the standard, species-based abstraction brings an exponentially smaller state space.

Example 4.1 Consider a system with particles of type A and B, each having two sites, x and y. Whenever two complexes encounter, they may form a bond between a free site y and a free site x, at rate k [5] . The bond can be released at rate k^-. In graphical notation, the model is summarized in Fig.6.

The set of agent types is $\mathcal{A} = \{A, B\}$, and the set of site types is $\mathcal{S} = \{x, y\}$. The contact map is $G = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V} = \{(A, \varnothing, \{x,y\}), (B, \varnothing, \{x,y\})\}$, $\mathcal{E} = \{((A,y),(B,x)), ((A,x),(B,y))\}$. Annotated contact map is a contact map $G_\equiv$, with annotation $\equiv = \{((A,y),(B,x)), ((A,x),(B,y))\}$. Set of variables is $Var = \{x_A, y_A, x_B, y_B, yx_{(B,A)}, yx_{(A,B)}\}$, and the set of rules is $\mathcal{R} = \{R_1, R_1^-, R_2, R_2^-\}$, where

$$\begin{aligned}
R_1 &: \neg y_A, \neg x_B, \neg yx_{(A,B)} \to y_A, x_B, yx_{(A,B)}@k,\\
R_1^- &: y_A, x_B, yx_{(A,B)} \to \neg y_A, \neg x_B, \neg yx_{(A,B)}@k^-,\\
R_2 &: \neg y_B, \neg x_A, \neg yx_{(B,A)} \to y_B, x_A, yx_{(B,A)}@k,\\
R_2^- &: y_B, x_A, yx_{(B,A)} \to \neg y_B, \neg x_A, \neg yx_{(B,A)}@k^-.
\end{aligned}$$

[5] The rate of binding does not depend on the size of interacting complexes, as it is often assumed for aggregation models.

4.1 Abstractions

Reachable species can be categorized into two types: *chains*– with two free sites and *rings*– with no free sites. We say that a chain or a ring is of length i if it has i agents in total. Chains can be classified into four different kinds, depending on which sites are free. Let the chains be denoted by C_i^{AB} (free site x of agent A, and site y of agent B), C_i^{BA}, C_i^{AA}, C_i^{BB}, and let C_i° be the ring of length i. Given $n_A = n_B = n$ copies of each of the agents, the species is either C_i^{AB}, C_i^{BA}, or C_i°, for $i = 2, 4, \ldots, 2n$, or C_i^{AA}, C_i^{AB}, for $i = 1, 3, \ldots, 2n-1$. All states lumped in the species-based abstraction are abstracted by the same multiset of reachable species.

The stochastic annotation induces two stochastic fragment classes: $Var_{/\equiv_s} = \{\{y_A, x_B, yx_{AB}\}, \{y_B, x_A, yx_{BA}\}\}$. The common feature of all states lumped in the fragment-based abstraction are: the number of bonds between y site of agent A and x site of agent B, called AB-bond type in the remaining text, and the number of bonds between y site of agent B and x site of agent A, called BA-bond type in the remaining text.

Assume given two instances of agent A and two instances of agent B, i.e. $n(A) = n(B) = 2$, and let the states $s_1, s_2 \in S$ be such that $\mathcal{L}(s_1)(yx^{1,1}_{(A,B)}) = \mathcal{L}(s_1)(yx^{1,2}_{(B,A)}) = 1$ (and all other sites are evaluated to 0), and $\mathcal{L}(s_2)(yx^{1,2}_{(A,B)}) = \mathcal{L}(s_2)(yx^{2,2}_{(B,A)}) = 1$ (and all other sites are evaluated to 0). It is easy to inspect that $s_1 \sim_p s_2$. The witness permutation over the identifiers is $\sigma_A(1) = 1$, $\sigma_A(2) = 2$, $\sigma_B(1) = 2$, $\sigma_B(2) = 1$. The witness permutation exists, because both states s_1 and s_2 have a chain of length 3, containing two agents A and one agent B, and one free B. We use the multiset notation $\{C_3^{AA}, C_1^{BB}\}$ to represent the lumped state $[s_1]_{\sim_p} = [s_2]_{\sim_p}$. In the fragment-based abstraction $M^f_{\mathcal{L}} = (S^f, L^f, a^f, S^f_0)$, induced by the stochastic annotation $\equiv'_s$, consider the state $s_3 \in S$, such that $\mathcal{L}(s_3)(yx^{1,1}_{(A,B)}) = \mathcal{L}(s_1)(yx^{2,1}_{(B,A)}) = 1$ (and all other sites are evaluated to 0). Then, $s_1 \not\sim_p s_3$, but $s_1 \sim_f s_3$, and the witness family of permutations is: $\sigma^{(1)}$, the identity function, and $\sigma^{(2)}_A(1) = 2$, $\sigma^{(2)}_A(1) = 1$, $\sigma^{(2)}_B(1) = 2$, $\sigma^{(2)}_B(2) = 1$. The corresponding population-based state $[s_3]_{\sim_p}$ is described by a multiset $\{C_2^{\circ}, C_1^{AA}, C_1^{BB}\}$.

4.2 Comparing the fragment-based and species-based abstraction

The comparison between the species-based and fragment-based abstraction is summarized in Fig. 7. For the presented estimation, we assume the same number of copies of agents A and B: $n_A = n_B = n$.

In order to count the number of reachable states, we used the approximation for the number of partitions of n, denoted by $P(n)$ (one partition of n is writing it as a sum of non-negative integers). There exists no closed-form expression for $P(n)$, but one of the well-known asymptotic limits is $P(n) \approx \frac{1}{4n\sqrt{3}} e^{\pi\sqrt{\frac{2n}{3}}}$ [12]. The connection between the number of partitions of $2n$, and the number of reachable species-based states is the following. Consider one partition $n = n_1 + \ldots + n_k$, $n_1 \leq \ldots \leq n_k$, and a state $s \in S$, such that $[s]_{\sim_p}$ is represented by a multiset $\{C^{AB}_{n_1}, \ldots, C^{AB}_{n_k}\}$. It has

	agents	reactions (rules)	number of states
SPECIES	$5n$	$2n^2$	$O(e^n)$
FRAGMENTS	2	4	$O(n^2)$

Fig. 7. Species-based and fragment-based abstraction for Example 4.1: for $n_A = n_B = n$, $2n^2$ reactions are needed to describe the model specified by 4 rules; The number of species is $5n$, and the number of stochastic fragments is two. The state space of species-based abstraction counts $O(e^n)$ states, and the state space of fragment-based abstraction counts $(n+1)^2$ states.

exactly n agents A and n agents B, so it is a reachable state in $M^p_{\mathcal{L}_p}$. Therefore, the set S^p counts at least $P(n)$ states. Note that this rough estimation can be significantly improved with more detailed combinatorial analysis.

5 Convergence properties

In this section, we reason about the practical aspect of using the fragment-based abstraction instead of the standard, species-based abstractions of rule-based models. The properties, discussed already in [8], are: (i) soundness: the probability of being in a fragment-based state is equal to the sum of probabilities of being in the corresponding species-based states, (ii) invertability: being in a fragment-based state, can one reconstruct the probabilities over the corresponding species-based states, by applying the function $\hat{\gamma}$, that is a static property of the set of rules only. The invertability property however holds only for certain initial distributions. If the modeler cannot enforce the system to start reacting in those initial states, the approach becomes useless for reconstructing the species-based dynamics.

We show here that, if all rules are reversible, then the invertability property holds at the stationary distribution, regardless of the initial distribution.

Definition 5.1 *(reversible rule)* Given a rule-based system $\mathcal{B} = (\mathcal{V}, \mathcal{E}, n, \mathcal{R})$, we say that a rule $R = (p, q, k) \in \mathcal{R}$ is reversible, if there exists a rule $R' = (p', q', k') \in \mathcal{R}$, such that $p' = q$ and $q' = p$.

Theorem 5.2 Consider $M_{\mathcal{L}}$, the ILTS of a rule-based system $\mathcal{B} = (\mathcal{V}, \mathcal{E}, n, \mathcal{R})$, its species-based abstraction $M^p_{\mathcal{L}_p} = (S^p, L^p, a^p)$, and its fragment-based abstraction $M^f_{\mathcal{L}_f} = (S^f, L^f, a^f)$. Let the corresponding stochastic processes be (X_t), (Y_t), and (Z_t).

Let $\gamma : S^f \times S^p \to [0,1]$ denote the ratio of the number of states lumped to $[s]_{\sim_p}$ and to $[s]_{\sim_f}$:

$$\hat{\gamma}([s']_{\sim_f}, [s]_{\sim_p}) := \begin{cases} \frac{|[s]_{\sim_p}|}{|[s']_{\sim_f}|} & \text{, if } s{\sim_p}s' \\ 0 & \text{, otherwise.} \end{cases} \tag{1}$$

Due to Lemma 3.5, for a given $s \in S$, the function $\gamma([s]_{\sim_f}) : S^p \to [0,1]$ is a probability distribution, i.e. $\sum_{[s']_{\sim_p} \in S^p} \hat{\gamma}([s]_{\sim_f})([s']_{\sim_p}) = 1$.

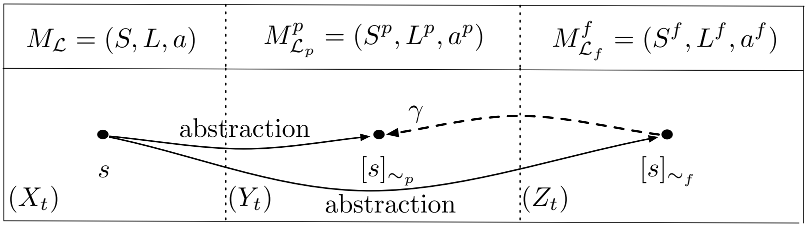

$M_{\mathcal{L}} = (S, L, a)$
$M^p_{\mathcal{L}_p} = (S^p, L^p, a^p)$
$M^f_{\mathcal{L}_f} = (S^f, L^f, a^f)$
abstraction
γ
s
$[s]_{\sim_p}$
$[s]_{\sim_f}$
(X_t)
(Y_t)
abstraction
(Z_t)

Fig. 8. ILTS $M_{\mathcal{L}}$ that models the rule-based system, its species-based abstraction $M^p_{\mathcal{L}_p}$, and its fragment-based abstraction $M^f_{\mathcal{L}_f}$. The CTMC's underlying the systems are (X_t), (Y_t) and (Z_t). Given the fragment-based state $[s]_{\sim_f}$, the function $\gamma : S^f \times S^p \to [0,1]$ is used for reconstructing the probability of being in a species-based state $[s]_{\sim_p}$, given the probability of the process Z_t being in the fragment-based state $[s]_{\sim_f}$.

Moreover, let $\gamma_t : S^f \times S^p \to [0,1]$, $t \in \mathbb{R}_{\geq 0}$ denote the conditional probability that the process Y_t is in state $[s]_{\sim_p}$, given that it is in one of the states $[s']_{\sim_p}$, that it is lumped with s in the fragment-based abstraction:

$$\gamma_t([s']_{\sim_f}, [s]_{\sim_p}) := P(Y_t = [s]_{\sim_p} | Z_t = [s']_{\sim_f}, s' \sim_f s).$$

Note that γ_t is a time-dependent variable, whereas $\hat{\gamma}$ is a constant. Then the following holds:

- (soundness) If $\gamma_0 = \hat{\gamma}$, then $P(Z_t = [s]_{\sim_f}) = P(Y_t \subseteq [s]_{\sim_f})$, and
- (invertability) If $\gamma_0 = \hat{\gamma}$, then $\gamma_t = \hat{\gamma}$, for $t \in [0, \infty)$;
- (convergence) If (Y_t) has a unique stationary distribution, then $\gamma_t \to \hat{\gamma}$, as $t \to \infty$.

The properties of soundness and invertability are discussed in our previous papers [8,9]. The detailed mathematical proofs are given in [17].

Corollary 5.3 *Given a rule-based system $\mathcal{B} = (\mathcal{V}, \mathcal{E}, n, \mathcal{R})$, if all rules in $\mathcal{R}$ are reversible, then $\gamma \to \hat{\gamma}$, when $t \to \infty$.*

Proof sketch. Since we deal with rule-based models with a conserved number of agents, the state space is finite, and hence the stationary distribution exists. If all the rules in a rule set are reversible, then the underlying CTMC is irreducible, since each transition in the CTMC is symmetric. Since the CTMC is non-explosive and irreducible, its stationary distribution is unique, and convergence of γ_t to $\hat{\gamma}$ follows ([16], Thm.3.6.2).

Example 5.4 (Ex.4.1 Cont'd)

Let $n_A = n_B = 2$, and consider the fragment-based state $[s]_{\sim_f} \in S^f$, that lumps all states with one bond of type AB, and one bond of type BA. There are four different population-based states which are lumped to $[s]_{\sim_f}$: a multiset containing one chain C_3^{AA} and one chain C_1^{BB}, a multiset containing one ring C_2° and two chains– C_1^{AA} and C_1^{BB}, and a multiset containing chains C_2^{AB} and C_2^{BA}. The function $\hat{\gamma}$ is

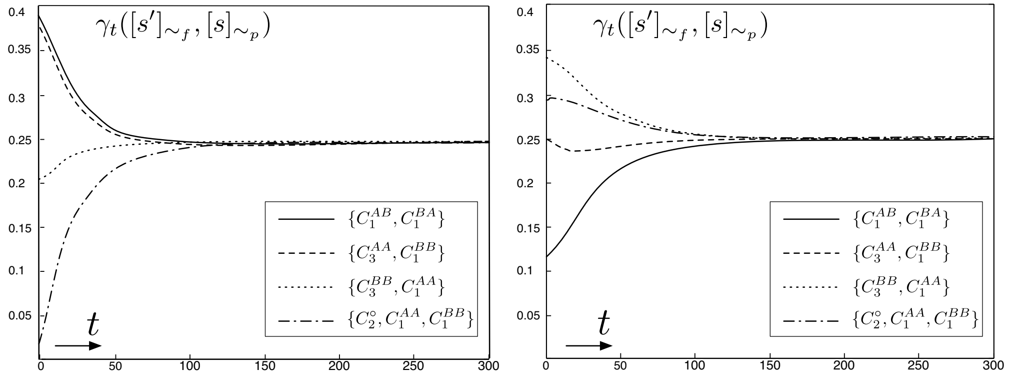


Fig. 9. Example 4.1: Convergence. The conditional probability $\gamma_t([s']_{\sim_f},[s]_{\sim_p})$ is computed for for two different random initial distributions, by a numerical integration of the CTMC's assigned to $M^p_{\mathcal{L}_p}$. We choose a fragment-based state $[s]_{\sim_f}$ which lumps all states with one bond of type AB, and one bond of type BA. The four plots correspond to the distributions of the corresponding four species-based states: $[s_1]_{\sim_p}$, containing complexes C_2^{AB} and C_2^{BA}, and $[s_2]_{\sim_p}$, containing complexes C_3^{AA} and C_1^{BB}, $[s_3]_{\sim_p}$, containing complexes C_1^{AA} and C_3^{BB}, and $[s_4]_{\sim_p}$, containing complexes C_2° and C_1^{AA} and C_1^{BB}. The convergence to $\hat{\gamma}$ (given in Eq. 2) is evident. We remark that $\hat{\gamma}$ is uniform distribution for the chosen states, but it is not uniform in a general case.

computed by using the Eq. (2) given in Thm.5.2:

$$\hat{\gamma} = \begin{pmatrix} \{C_3^{AA},C_1^{BB}\} & \{C_2^{AB},C_1^{AA},C_1^{BB}\} & \{C_2^{AB},C_2^{BA}\} & \{C_3^{BB},C_1^{AA}\} \\ 4/12 = 1/4 & 4/12 = 1/4 & 4/12 = 1/4 & 4/16 = 1/4 \end{pmatrix}. \tag{2}$$

The convergence is illustrated in Fig.9. The reasoning for computing the distribution $\hat{\gamma}$ is the following: there are four different states lumped to $\{C_3^{AA},C_1^{BB}\}$: either B^1 or B^2 is free, and either A^1 or A^2 is the one bound to a B agent via the y site. There are 16 different states lumped to $[s]_{\sim_f}$: four ways to identify the A and B which form the bond yx_{AB}, combined with one of four ways to identify the bond yx_{BA}.

It is worth noting that the function $\hat{\gamma}$ is not necessarily uniform. For example, for $n_A = n_B = 3$, the fragment-based state with 3 bonds of type AB and 3 bonds of type BA has three different population-based states, described by the multisets: $\{C_6^{\circ}\}$, $\{C_2^{\circ},C_4^{\circ}\}$ and $\{3C_2^{\circ}\}$. The $\hat{\gamma}$ function in this case is

$$\hat{\gamma} = \begin{pmatrix} \{C_6^{\circ}\} & \{C_2^{\circ},C_4^{\circ}\} & \{3C_2^{\circ}\} \\ 12/36 = 1/3 & 18/36 = 1/2 & 6/36 = 1/6 \end{pmatrix}.$$

Conclusions

We have extended the framework for analysing the stochastic semantics of rule-based models, by showing that if all the rules in a rule set are reversible, the reconstruction of the probabilities over species-based states is always possible at the stationary distribution.

The analysis is done on a novel formalism of Boolean encodings of site-graphs and interpreted labelled transition systems, rather than on syntactic analysis of Kappa expressions, as it was done in previous related works [9], [8]. We showed in this formalism a complementary viewpoint to the fragment-based abstraction: it is a result of a particular composition operator over the species-based abstractions of appropriately chosen smaller sets of rules.

Finally, we demonstrated that the state space of the fragment-based abstraction can be exponentially smaller than the one of the species-based abstraction, on an example of colloidal aggregation.

Some of the questions which we plan to address in the future work are: (i) removing the assumptions about agent birth, deletion and the same agent appearing in one rule; (ii) relaxing the decomposition criterion by exploiting additional conservation laws; (ii) a tool which computes the decomposition, the fragment-based variables, and the $\hat{\gamma}$ function; (ii) error measure for non-exact abstractions.

References

[1] Baier, C., B. Haverkort, H. Hermanns and J.-P. Katoen, *Model-checking algorithms for continuous-time Markov chains*, IEEE Transactions on Software Engineering **29** (2003), p. 2003.

[2] Borisov, N. M., N. I. Markevich, J. B. Hoek and B. N. Kholodenko, *Signaling through receptors and scaffolds: Independent interactions reduce combinatorial complexity*, Biophysical Journal **89** (2005), pp. 951—966.

[3] Buchholz, P. and P. Kemper, *Kronecker based matrix representations for large Markov models*, in: *Validation of Stochastic Systems*, Lecture Notes in Computer Science **2925**, 2004, pp. 256—295.

[4] Changeux, J.-P. and S. J. Edelstein, *Allosteric mechanisms of signal transduction*, Science **308** (2005), pp. 1424–1428.

[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 Systems Biology **7** (2006).

[6] Danos, V. and C. Laneve, *Core formal molecular biology*, Theoretical Computer Science **325** (2003), pp. 69–110.

[7] Feret, J., V. Danos, J. Krivine, R. Harmer and W. Fontana, *Internal coarse-graining of molecular systems*, Proceedings of the National Academy of Sciences of the United States of America **106** (2009), pp. 6453–6458.

[8] Feret, J., T. A. Henzinger, H. Koeppl and T. Petrov, *Lumpability abstractions of rule-based systems*, MeCBIC (2010), pp. 142–161.

[9] Feret, J., H. Koeppl and T. Petrov, *Stochastic fragments: A framework for the exact reduction of the stochastic semantics of rule-based models*, http://en.scientificcommons.org/52625730 (2010), accepted in International Journal of Software and Informatics.

[10] Gardner, M. K., B. D. Charlebois, I. M. Jánosi, J. Howard, A. J. Hunt and D. J. Odde, *Rapid microtubule self-assembly kinetics*, Cell **146** (2011), pp. 582–592.

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

[12] Hardy, G. H. and S. Ramanujan, *Asymptotic formula in combinatory analysis*, Proceedings of the London Mathematical Society **S2-17(1)** (1918), pp. 75–115.

[13] Kemeny, J. and J. L. Snell, “Finite Markov Chains,” Van Nostrand, 1960.

[14] Koeppl, H. and T. Petrov, *Stochastic semantics of signaling as a composition of agent-view automata*, Electronic Notes in Theoretical Computer Science **272** (2011), pp. 3 – 17.

[15] Meakin, P., *Models for Colloidal Aggregation*, Annual Review of Physical Chemistry **39** (1988), pp. 237–267.

[16] Norris, J. R., "Markov Chains (Cambridge Series in Statistical and Probabilistic Mathematics)," Cambridge University Press, 1998.

[17] Petrov, T., A. Ganguly and H. Koeppl, *Markov chain aggregation and its applications to combinatorial reaction networks* (in preparation), BISON group, ETH Zurich.

[18] Plotkin, G. D., *A structural approach to operational semantics*, Journal of Logic and Algebraic Programming **60-61** (2004), pp. 17–139.

[19] Pollard, T. D., *Regulation of actin filament assembly by Arp2/3 complex and formins.*, Annual review of biophysics and biomolecular structure **36** (2007), pp. 451–77.

[20] Rubenstein, R., P. Gray, T. Cleland, M. Piltch, W. Hlavacek, R. Roberts, J. Ambrosiano and J.-I. Kim, *Dynamics of the nucleated polymerization model of prion replication*, Biophysical Chemistry **125** (2007), pp. 360 – 367.

[21] Smock, R. G. and L. M. Gierasch, *Sending signals dynamically*, Science **324** (2009), pp. 198–203.

[22] Walsh, C. T., "Posttranslation Modification of Proteins: Expanding Nature's Inventory," Roberts and Co. Publisher, 2006.