Organisation-Oriented Coarse Graining and Refinement of Stochastic Reaction Networks

Chunyan Mu, Peter Dittrich, David Parker, Jonathan E. Rowe

Abstract—Chemical organisation theory is a framework developed to simplify the analysis of long-term behaviour of chemical systems. In this work, we build on these ideas to develop novel techniques for formal quantitative analysis of chemical reaction networks, using discrete stochastic models represented as continuous-time Markov chains. We propose methods to identify organisations, and to study quantitative properties regarding movements between these organisations. We then construct and formalise a coarse-grained Markov chain model of hierarchical organisations for a given reaction network, which can be used to approximate the behaviour of the original reaction network. As an application of the coarse-grained model, we predict the behaviour of the reaction network systems over time via the master equation. Experiments show that our predictions can mimic the main pattern of the concrete behaviour in the long run, but the precision varies for different models and reaction rule rates. Finally, we propose an algorithm to selectively refine the coarse-grained models and show experiments demonstrating that the precision of the prediction has been improved.

Index Terms—stochastic reaction networks, probabilistic model checking, organisation theory, coarse-graining, refinement.

1 INTRODUCTION

In this paper, we study reaction networks and chemical organisation theory, in particular, investigating the applicability of formal verification to their analysis. Reaction networks are widely used in modelling chemical and biological phenomena. For example, the BioModels Database contains more than one hundred thousand reaction network models of bio-molecular systems; additionally, large-scale processes like infection dynamics, evolutionary population dynamics, or social systems can be modelled by reaction networks. A reaction network describes the structure of interactions between system elements in a formal and intuitively clear way. However, as a consequence of non-linear interactions, feedback loops, and large state spaces the implied overall dynamics can be difficult to understand and to analyse.

Chemical organisation theory provides a way to analyse complex dynamical networks and reason about the long-term behaviour of chemical systems. The complex network is decomposed into a set of sub-networks called “organisations”. An organisation is a set of objects (for example, the species or molecules in a reaction system) which are closed and self-maintaining. Informally, closed means that no new object can be produced by the interactions within the set, and self-maintaining means that no object of the set disappears from the system, i.e., every consumed object of the set can be generated within the set. The concept of organisation allows us to lift the complex reaction network to a hierarchical structure including all stable states and states depicting accumulating molecules regarding to the organisations. The dynamics of the complex state space of the reaction network can then be mapped to movements among the set of organisations. Building a chemical organisation-based model thus helps us to model the structure and behaviour of complex reaction networks, and to simplify the dynamical analysis of the overall system.

In order to study the evolution of reaction networks, we apply probabilistic model checking, a formal verification technique for modelling and analysis of systems with stochastic behaviour. It has been used to study models across a wide range of application domains, including chemical and biological systems. Probabilistic model checking is based on the exhaustive construction and analysis of a state-based probabilistic model, typically a Markov chain or variant. In this work, we model the reaction networks as continuous-time Markov chains. Quantitative properties of interest about the system being analysed are formally specified using temporal logic. Here we use CSL (Continuous Stochastic Logic) with rewards, a quantitative extension of the temporal logic CTL.

Specifically, we apply CSL model checking of continuous-time Markov chains to investigate connections between chemical organisations using model decompositions into strongly connected components (SCCs). We develop an algorithm to automatically find organisations, and then perform a quantitative dynamical analysis in terms of organisations, asking, for example, “what is the probability of moving from one organisation to another?” or “what is the expected time to leave an organisation?” We implement our techniques as an extension of the probabilistic model checking tool PRISM, and illustrate the approaches on a set of example reaction networks.

A coarse-grained Markov chain model of hierarchical organisations for a given reaction network is then constructed as a result, and we prove that this yields safe approximations to the dynamical properties of the original...
Approximating and predicting the system behaviour over time is another direct application of our coarse-grained model, which can save significant time and space comparing with doing prediction by using the original concrete model. Specifically, we study the following question: “Given a coarse-grained model and a fixed volume of the chemical species, what is the probability distribution of the molecular population after t time units?”. Our experiments show that our prediction can mimic the main pattern of concrete behaviour in the long run, but the precision varies for different models and rates of their reaction rules. In order to improve the precision of the approximation and prediction, we develop an algorithm to selectively refine the coarse-grained models. The basic idea is that we partition the abstract states of the coarse-grained model using k-means clustering techniques. Our experiments show that the precision of the prediction is improved as a result.

This paper is an extended version of [21]. It also contains extra examples, experiments for performance comparisons and, in particular, the method for refinement of the coarse-grained models.

**Related work.** There are various approaches to modelling the dynamics of reaction networks. Feinberg and Horn [9] proposed methods to identify positive stationary states in which all molecular species are present in a network. Heinrich and Schuster [13] study network structure based on flux modes, each of which specifies a set of reaction rules that can take place at a steady state and thus implies a set of species participating in those reactions. Species relating to a flux mode were not required to be self-maintaining or closed however. We are more interested in the stationary states in which a subset of species are present, which is formalised in organisation theory [7]. In that area, the focus was typically on qualitative properties, and ODEs [8], approximating the evolution of reaction networks in continuous dynamical systems. Kreyssig et. al [15] studied the effects of small particle numbers on long-term behaviours in discrete biochemical systems. We build on their notion of discrete organisation but focus on quantitative analysis of the transitive dynamics among the organisations, which was not considered in [15]. Other approaches for approximate analysis of discrete models of reaction networks include the use of Linear Noise Approximation [4], the Central Limit Approximation [3] and “sliding window” abstractions [29].

**Outline.** This paper is organised as follows. Section 2 gives an overview of probabilistic model checking. Sections 3 and 4 present the details of modelling chemical reaction networks as CTMCs and introduce definitions for building connections with chemical organisation theory. Section 5 proposes methods for a quantitative organisation-based analysis. Section 6 formalises the definition of the organisation-based interval coarse-grained model, which safely approximates the probabilistic behaviours of the system, but may suffer from over-estimation. Section 7 demonstrates how to use our organisation-oriented coarse-grained model to predict system behaviour over time evolution. For the purpose of improving the precision of the approximation and prediction, Section 8 proposes an algorithm to selectively refine the coarse-grained models, and shows how precision of the analysis is improved by the refinement. Section 9 draws conclusions.

## 2 Probabilistic Model Checking

Probabilistic model checking is a variant of model checking [6], a well-established formal method to automatically verify the correctness of real-life systems. Classical model checking answers the question of whether the behaviour of a given system satisfies a property or not. It thus requires two inputs: a description of the system and a specification of one or more required properties of that system, normally in temporal logic (such as CTL or LTL).

In probabilistic model checking, the models are extended with information about the likelihood that transitions take place. In practice, these models are usually Markov chains or Markov decision processes. In this work, we model the reaction systems as continuous-time Markov chains (CTMCs). Properties expressed in temporal logic are also of a quantitative nature. For instance, instead of verifying that “species A eventually vanishes”, we ask that “what is the probability of species A eventually vanishing?”. In this work, we use the temporal logic Continuous Stochastic Logic (CSL) [1], [2].

The remainder of this section reviews some preliminary definitions for the probabilistic model checking techniques that we use in this paper.

### 2.1 Continuous-Time Markov Chains

Continuous-Time Markov Chains are widely used in fields such as performance analysis or systems biology to model systems with stochastic real-time behaviour. Formally, we define them as follows.

**Definition 1 (CTMC).** A CTMC is a tuple $A = (Q, Q_0, \Delta, L)$, where: $Q$ is a finite set of states; $Q_0 \subseteq Q$ is the set of initial states; $\Delta : Q \times Q \to \mathbb{R}_{\geq 0}$ is the transition rate matrix; $L : Q \to 2^{AP}$ is a labelling function assigning, to each state $q \in Q$, a set of atomic propositions, from a set $AP$, that are true in $q$.

The transition rate matrix $\Delta$ assigns a rate to each pair of states in the CTMC, which is used as the parameter of an exponential distribution.

### 2.2 Continuous Stochastic Logic

In this work, the probabilistic temporal logic CSL (Continuous Stochastic Logic) is used to formally represent properties of reaction networks. It was originally introduced by Aziz et al. [1] and extended by Baier et al. [2]. The extended version allows for the specification of reward (or cost) properties, to reason about rewards (or costs) that have been attached to a CTMC. The extended version of CSL that we use allows us to represent properties such as “the probability of all of species $A$ degrading within $t$ time units is at most 0.1” or “the expected time elapsed before a $B$ molecule first appears is at most 10”.

**Definition 2 (CSL syntax).** An (extended) CSL formula is an expression $\Psi$ derived from the grammar:

$$\Psi ::= \text{true} | p | \neg \Psi | \Psi \land \Psi | P_{\text{occ}}(\Psi \cup \Psi) | S_{\text{occ}}(\Psi) | R_{\text{occ}}(\Phi \Psi)$$
where \( p \in AP \) is an atomic proposition, \( \lambda \in [0,1] \) is a probability threshold, \( r \in \mathbb{R}_{\geq 0} \) is a reward threshold, \( \vartriangleleft \in \{ <,\leq,\geq,> \} \) and \( I \) is an interval of \( \mathbb{R}_{\geq 0} \).

CSL formulas are evaluated over the states of a Markov chain. A state \( \text{CSL} \) formulas are evaluated over the states of a Markov chain. A state within interval \( I \) is satisfied at some future time point within interval \( I \), and that \( \Psi \) is true up until that point. Common derived operators include: “eventually” \( \diamond \Psi := \text{true} U I \) and “always” \( \square \Psi := \neg \diamond \neg \Psi \). For example, \( \text{P}_{\leq 0}(\square \Psi) \equiv \text{P}_{\geq -1}(\diamond \neg \Psi) \).

The \( S \) operator describes the steady state (long-run) behaviour of the CTMC. The formula \( S_{\leq 0}(\psi) \) specifies that the steady-state probability of being in a state satisfying \( \psi \) meets the bound \( \infty \). The \( R \) operator is used for reward properties: \( \text{R}_{\text{rew}}(\diamond \Psi) \) is true from state \( q \) if the expected reward cumulated before a state satisfying \( \Psi \) is reached meets the bound \( \infty \). Rewards and costs are treated identically: here, we will use the \( R \) operator to formalise properties about the expected time elapsing before an event’s occurrence.

We omit a full definition of the semantics of CSL with respect to a Markov chain. Full details can be found in, for example, [2].

3 Modelling Reaction Networks with CTMCs

A reaction network consists of a set of molecules (or, molecular species to be more precise) and a set of reaction rules.

Definition 3. A reaction network is a pair \( (\mathcal{M}, \mathcal{R}) \) consisting of a set of possible molecular species \( M \) and a set \( \mathcal{R} \subseteq \mathcal{P}(\mathcal{M}) \times \mathcal{P}(\mathcal{M}) \) of possible reactions among those species, where \( \mathcal{P}(\mathcal{M}) \) denotes the set of all multisets of elements over the set \( \mathcal{M} \). For a reaction \( (R,P) \in \mathcal{R} \), the multisets \( R \) and \( P \) denote the reactants and products of the reaction, respectively, and we write \( R(s) \) and \( P(s) \) for the number of molecules of species \( s \) consumed by (reactants) and produced by (products) the reaction, respectively.

For simplicity, we write \( s_1 + s_2 + \cdots + s_n \rightarrow s'_1 + s'_2 + \cdots + s'_{n'} \) instead of \( \{s_1, s_2, \ldots, s_n\} \rightarrow \{s'_1, s'_2, \ldots, s'_{n'}\} \) ∈ \( \mathcal{R} \) to denote the existence of a reaction.

There are multiple ways in which we can obtain a dynamical model given a reaction network. One way is to consider (real-valued) concentrations of each molecular species and then represent the (deterministic) behaviour of the reactions as a set of ordinary differential equations. Here, we take a discrete, stochastic view of the network, modelling the (integer-valued) population count of each species and considering its evolution as a stochastic process, and in particular as a continuous-time Markov chain [12]. The latter is particularly appropriate when the numbers of molecules can be assumed to be relatively small in practice, and is the approach that we take in this work.

Furthermore, we will also assume that the reaction network is executing within a finite volume, which is modelled by limiting the total number \( N_{\text{max}} \in \mathbb{N} \) of molecules that can be present at any given time [15]. We also need to define the rates at which reaction events occur in the CTMC. To retain a general approach, we allow an arbitrary function \( \tau_r \) from reactant populations to rate values for each reaction \( r \). A typical default, which we use in some, but not all, of our examples, is the law of mass-action, multiplying the number of molecules of each reactant by a fixed kinetic rate associated with the reaction (and assuming the stoichiometric coefficient of each reactant is at most one). This gives \( \tau_r(q) = \lambda_r \cdot \prod_{s \in R} q(s) \) with \( \lambda_r \) being a kinetic rate constant for reaction \( r \).

Definition 4 (CTMC for reaction network). Given a reaction network \( (\mathcal{M}, \mathcal{R}) \), a volume limit \( N_{\text{max}} \in \mathbb{N} \) and a rate function \( \tau_r : \mathcal{M} \to \mathbb{R}_{\geq 0} \) for each \( r \in \mathcal{R} \), we define the corresponding CTMC \( \mathcal{A} = (Q, Q_0, \Delta, L) \) where:

- \( Q = \{ q : M \to \mathbb{N} \mid \sum_{s \in M} q(s) \leq N_{\text{max}} \} \)

is the set of population counts of \( M \) and \( \Delta \) is defined as follows. For states \( q, q' \in Q \), we write \( q \xrightarrow{(R,P)} q' \) if and only if, for each species \( s \in M \), we have \( q(s) \geq R(s) \) and \( q'(s) = q(s) - R(s) + P(s) \), and \( \sum_{s \in M} q'(s) \leq N_{\text{max}} \).

Then, for any \( q, q' \in Q \), we have:

- \( \Delta(q, q') = \sum_s \{ \tau_r(q) \mid r \in \mathcal{R} \text{ and } q \xrightarrow{r} q' \} \)

We call \( r \) the transition label of \( q \xrightarrow{r} q' \).

\( Q_0 \) can be any subset of \( Q \) representing initial configurations of interest, and \( L \) can be any labelling function over \( Q \) that identifies states with relevant properties.

Each state \( q \in Q \) of the CTMC gives the number \( q(s) \) of molecules of each species \( s \in M \) currently present. For a state \( q \), we also write \( \phi(q) \) for the set of molecular species that are present, i.e., \( \phi(q) = \{ s \mid q(s) > 0 \} \), and define \( \phi(Q') = \cup_{q \in Q'} \phi(q) \) for a set of states \( Q' \subseteq Q \). We let \( Acc(q) \subseteq Q \) denote the states that are reachable from \( q \).

Example 1. Consider the reaction network \( \mathcal{A} \) with species \( M = \{ a, b \} \) and reactions \( \mathcal{R} = \{ a + b \rightarrow a + 2b, a \rightarrow 2a, b \rightarrow 2b, a \rightarrow \emptyset, b \rightarrow \emptyset \} \). Assume the volume of the system is \( N_{\text{max}} = 4 \), and that the rate of each reaction rule is the multiplication of the number of the reactants. We obtain a CTMC with 15 states (see Fig. [1]).

4 Chemical Organisation Theory and SCC Decomposition

Chemical organisation theory [7] provides a way to cope with the complex “constructive” dynamics of a reaction network by deriving a set of organisations [11], and then mapping the movement through a state space to a movement between organisations. Such an abstract view allows us to analyse and predict the dynamical behaviour of a complex reaction network more easily.

An organisation is a set of molecules that is algebraically closed and self-maintaining. A subset \( C \subseteq M \) is called “closed” if no molecules outside \( C \) can be produced by applying any reaction that uses only reactants from \( C \); a subset \( S \subseteq M \) is “self-maintaining” if all reactions that are able to fire in \( S \) can occur at certain strictly positive rates without reducing the amount of any species of \( S \).

Definition 5 (Organisation [7]). A subset of \( O \subseteq M \) is a chemical organisation if it is closed and self-maintaining,
that is, if for all \((R, P) \in \mathcal{R}, R \subseteq O\) implies \(P \subseteq O\) (closure), and there exists a strictly positive flux vector \(v > 0\) such that \(N_O \cdot v \geq 0\) with \(N_O\) being the stoichiometric matrix of the reactions that use only reactants from \(O\) (self-maintenance).

As discussed above, we model the dynamics of a reaction network as a Markov chain. A state is defined by a discrete number for each molecular species. With a limited total amount of molecules, both cases of too few and too many molecules can prevent reaction rules being fired. As a consequence, we need to define discrete organisations, and the states contributing to generate them. In the following, given a state \(q\), \(\mathcal{R}_q\) denotes the reactions firing in any of the reachable states of \(q\).

**Definition 6 (Discrete organisation and internal generator [15]).** Let \((\mathcal{M}, \mathcal{R})\) be a reaction network. A subset of species \(D \subseteq \mathcal{M}\) is called a discrete organisation if there is a state \(q \in Q\) such that: (i) \(\phi(Acc(q)) = D\) (closure); and (ii) there is sequence of transition labels \((r_1, \ldots, r_k)\) where \(r_i \in \mathcal{R}\) such that \(\bigcup_{i=1}^{k} \{r_i\} = \mathcal{R}_q\) and \(q = (r_k \circ \cdots \circ r_1)(q)\) satisfies \(\forall s \in D : q'(s) \geq q(s)\) (self-maintenance). Such a state \(q\) is called an internal generator of the discrete organisation.

**Definition 7 (Generator).** A state \(q' \in Q\) is called a generator of organisation \(D\) iff \(\exists q \in Acc(q')\) such that \(q\) is an internal generator of \(D\).

Note that, in general, the organisation \(D\) generated by a state \(q'\) is not unique. However, if \(q\) is an internal generator, there is only one organisation it generates. Unless specifically stated otherwise, we say organisation rather than discrete organisation in the rest of the paper.

**Example 2.** The discrete organisations for Example 1 are: \(\{a, b\}, \{a\}, \{b\}, \{\}\) and the corresponding generators are, respectively (cf. Fig 1):

- \(\{6, 7, 8, 10, 11, 13\}\),
- \(\{5, 6, 7, 8, 9, 10, 11, 12, 13, 14\}\),
- \(\{1, 2, 3, 4, 6, 7, 8, 10, 11, 13\}\),
- \(\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14\}\).

In order to analyse the system behaviour and perform an organisation-based quantitative analysis of the reaction network, we study the connections between chemical organisations and the decompositions into strongly connected components (SCCs) of the Markov chain.

**Definition 8 (SCC [14]).** A strongly connected component (SCC) is a maximal set of states \(T\) such that, for every pair of states \(q, q' \in T\), there is a path from \(q\) to \(q'\).

Intuitively, in the Markov chain for a reaction network, SCCs are important for an organisation-based analysis. However, some but not all SCCs correspond to organisations. In the next section, we will describe an algorithm to find organisations based on a decomposition into SCCs and then identifying those self-maintaining a set of species. We first note that bottom strongly connected components do relate to organisations.

**Definition 9 (BSCC).** A bottom strongly connected component (BSCC) is an SCC \(T\) from which no state outside \(T\) is reachable from \(T\).

**Proposition 1.** Each BSCC corresponds to a (unique) organisation, which is generated (uniquely) by any state of that BSCC.

However, there are organisations whose internal generators are not contained in any BSCC. In order to also include such organisations, we call SCCs that correspond to an organisation good SCCs.

**Definition 10 (Good SCC).** An SCC \(T\) is called good if it contains a cycle of the firing of every “possible” reaction rule, i.e., those whose reactants \(R\) appear in the SCC \((R \subseteq \{\phi(q) \mid q \in T\})\).

**Example 3.** All SCCs are good in Example 1.

Clearly, some generators can contribute to multiple organisations. This makes it more difficult to decompose the Markov into its sets of generators. However, internal generators located in good SCCs contribute uniquely to an organisation.
Proposition 3. Given a good SCC $T$, let $A = \phi(T)$, if $A$ is closed, then $A$ is a discrete organisation, then $\{q \mid q \in T\}$ is the set of internal generators of $A$.

Example 4. In Example 1, the internal generators of organisations $\{a, b\}$, $\{a\}$, $\{b\}$ and $\{\}$ are $\{6, 7, 8, 10, 11, 13\}$, $\{5, 9, 12, 14\}$, $\{1, 2, 3, 4\}$ and $\{0\}$, respectively.

5 ORGANISATION-BASED ANALYSIS OF REACTION NETWORKS

In this section, we propose techniques for quantitative organisation-based analysis of reaction networks. We first introduce an algorithm to find the set of organisations for a specific reaction network. We then use probabilistic model checking to analyse quantitative properties regarding the dynamics of the network with respect to its organisations. Such organisation-based quantitative analyses can be used to construct the structure of reaction-based coarse-grained model, and provide a framework to approximate the complex dynamical behaviours of the original reaction networks in our next step.

5.1 Finding Organisations

Computing the organisations of a reaction network requires an analysis of the strongly-connected components of its Markov chain’s underlying transition graph. Since every state in a good SCC is a generator of an organisation, we identify good SCCs to find the organisations of the reaction network. Algorithm 1 presents the procedure for finding organisations of a given reaction network modelled as a CTMC. It is based on the following procedures:

- Tarjan($A$) returns the set of strongly-connected components of the Markov chain $A$, using Tarjan’s SCC algorithm [23] on the underlying digraph;
- findGoodSCCs(SCC) returns the “good” part SCC$_G$ of $A$ in which each possible reaction rule is able to be fired;
- find a set of closed molecules appearing in each $scc \in$ SCC$_G$, and its relevant internal generators i.e., states in $scc$ which generate the organisation.

5.2 Organisation-Based Probabilistic Analysis

We now illustrate, via several examples, how we derive quantitative organisation-based properties of reaction networks. We implemented the organisation and generator detection process described above in the PRISM model checker, along with a translator that converts descriptions of reaction networks into the PRISM modelling language to allow construction of the corresponding CTMC. Organisation-based properties of the network, such as probabilities (bounds or average) of the movements among organisations, or the expected time to leave or stay at an organisation, are computed using CSL formulae.

Example 5. Consider the reaction network with molecular species $M = \{a, b\}$ and reactions rules include: $a + b \rightarrow a + b$, $a \rightarrow 2a$, $b \rightarrow 2b$, $a \rightarrow 0$, $b \rightarrow 0$} with stochastic rates: $\gamma_a \cdot 2b$, $\alpha \cdot 2a$, $\beta \cdot 2b$, $(\gamma_a)^2$, $(\beta)^2$, respectively, where $\gamma_a$ denotes the number of molecules of species $a$ (note that this example does not assume the law of mass action).

The resulting model is described in the PRISM modelling language. It consists of a keyword describing the model type (ctmc), a set of constants, and a single module whose state is represented by a set of finite-ranging variables. Each variable stores the number of each molecular species. The behaviour of the module is specified by a set of guarded commands of the form $[y] \rightarrow r : u$. This command is interpreted as: if the predicate $y$ is true, then the system is updated by command $u$. Command $u$ comprises one or more statements of the form $x' = \ldots$ indicating how the value of variable $x$ is updated. The rate at which this occurs is given by $r$, which will be attached to the corresponding transition label in the underlying CTMC.

We show the PRISM language model for the example in Fig. 2. The resulting CTMC has 66 states and 201 transitions, and there are 4 SCCs ($\{a > 0, b > 0\}$, $\{a > 0, b = 0\}$, $\{a = 0, b > 0\}$, $\{a = b = 0\}$) with 1 BSCC ($\{a = b = 0\}$).

The first property we consider is the probability of moving between organisations. Specifically, the probability of moving from $O_1$ to $O_2$ can be specified in CSL as $P_{\exists t} \{0 \cup O_2\}$, where $0_1$ and $0_2$ are atomic propositions labelling states which represent internal generators of organisations $O_1$ and $O_2$. This refers to the probability of the event where $O_2$ is reached for the first time after leaving $O_1$, and is supported directly by PRISM.

In this example, all SCCs are good and each (good) SCC generates exactly one organisation. To visualise the movement between organisation, we analyse the property above for each pair of organisations and construct the abstract transition graph shown in Fig. 3. Blocks are labelled
with organisations and, for each possible transition between organisations, we show the range of probabilities (over all states in the source organisation) and the average value (over the same set of states).

We also consider the expected time to leave (the generators of each organisation). The CSL property to specify states in the source organisation and the average value of all organisations, we show the range of probabilities (over all states, and visits generators of organisations, and the expected time to leave them. For instance, the generators of organisation "a", with probability 0.56, in average and [0.0021, 0.1733] in bounds, and visits generators of organisation "b", with probability 0.044 in average and [0.0021, 0.1733] in bounds.)

Finally, we consider the effect of making some constructive perturbation to the reaction network, by adding rules to create species with a small rate. Fig. 4 shows the results of the same analysis described above for the following constructive perturbation: \{∅ → a, ∅ → b\} both with reaction rate \(γ = 0.01\). The result shows that, generating a and b with a small rate can cause an upward movement and slightly affect the system’s behaviour. The expected time to stay in each organisation decreases since the movement flow becomes slightly faster. This meets our intuition: the upward flow introduced by the constructive perturbation leads to a smoother flow of the system.

Example 6. Consider now the reaction network with \(M = \{a, b, c, d\}\) and \(R = \{a + b → a + 2b, a + d → a + 2d, b + c → 2c, c → b, b + d → c, b → ∅, c → ∅, d → ∅\}\) and rates (say \(R_1\)) as: \(γ_a * γ_b, γ_a * γ_d, γ_b * γ_c, γ_c, γ_b * γ_d, γ_b, γ_c, γ_d\) respectively.

Fig. 5 shows the structure of the CTMC for \(N_{max} = 5\). Even for a small volume \(N_{max} = 5\), the structure is quite complex: 126 states, 386 transitions, 28 SCCs and 6 BSCCs.

Fig. 6 illustrates, in the same fashion as above, the transition probabilities between all SCCs of the CTMC, and the expected time to leave them. Note that not all SCCs are good SCCs in this example: we highlight good SCCs in colour in Fig. 6.

For instance, the SCC labelled as \(\{∅ → a, ∅ → b\}\) cannot be fired within the SCC and it is therefore not good. In addition, the SCC labelled as \(\{105; 0, 25\}\) is also not a good one. There are two states in this SCC: state 99 \((a = 2, b = 0, c = 1, d = 1)\) and state 105 \((a = 2, b = 1, c = 1, d = 1)\). The set of molecules appearing in this node is closed, but reaction rules such as \(c → ∅\) cannot be fired within the SCC and it is therefore not good. In addition, the SCC labelled as \(\{12, 27; 0, 25\}\) is also not a good one. It contains state 12 \((a = 0, b = 0, c = 2, d = 1)\) and state 27 \((a = 0, b = 1, c = 1, d = 1)\). The set of molecules appeared in this node is closed, but reaction rule \(c → ∅\) cannot be fired locally, i.e., this decay will only introduce transitions to other SCCs. Similar cases can happen for some of the other reaction rules.

Fig. 7 presents the transition probabilities between good SCCs only, and the expected time to leave them. Note that multiple good SCCs can contribute to the generation of one organisation. For instance, both good SCCs labelled 65... and 98... contribute to organisation \(\{a, b, c\}\). Based on this graph, we can build up the transition graph over organisations.

Fig. 8 presents the transition probabilities between (internal generators of) organisations, and the expected time to leave each of them. It helps us to understand the movement between organisations and can be viewed as an abstract model capturing the behaviour of the reaction network at the level of organisations.
In addition, we also present transition graphs over the lattice of molecules (states in which a set of molecules in the lattice with positive numbers) for a quantitative analysis for organisations from a different point of view, see Fig. 6. The transition probabilities are given in bound. Specifically, the probability of movement from \( \{a, b, c\} \) to \( \{a, b\} \) can be specified as: \( P_{x}[(a > 0 \land b > 0 \land c > 0) \ \cup \ (a > 0 \land b > 0 \land c = 0)] \). Note that Fig. 5 and Fig. 6 can be used to build a coarse-grained model from a different perspective.

6 ORGANISATION-ORIENTED INTERVAL MARKOV CHAIN

The organisation-oriented transition graph generated by the quantitative analysis can be used to build a coarse-grained model (with either interval based or average based probabilistic transitions). Such a coarse-grained model can mimic
the complex reaction behaviours of the reaction network in an abstract way, whose state space and movement structure are much smaller. We can then perform approximation, prediction, and quantitative analysis upon the coarse-grained model in stead of the complex concrete model. This section focuses on formalising the interval-based organisation coarse-grained model. Specifically, our quantitative analysis computes an organisation-oriented interval Markov chain, in which each abstract state is specified by a set of internal generators of an organisation, and the abstract transition provides the information about the uncertainty of the abstract behaviours of the system. Probabilities of moving from one abstract state to another are given by the lower and upper bounds, which provides the under and over approximation of the concrete probabilities.

**Definition 11 (Organisation-oriented interval Markov chain).** An organisation-oriented interval Markov chain is a tuple $\mathcal{A}_I^\sharp = (Q^\sharp, Q_0^\sharp, \Delta^\sharp, L)$, where

- $Q^\sharp$ is a finite set of abstract states, each of which $q^\sharp ∈ Q^\sharp$ is a set of internal generators of an organisation $\alpha: q^\sharp \subseteq G_\alpha(o)$;
- $Q_0^\sharp \subseteq Q^\sharp$ is the set of initial abstract states;
- $\Delta^\sharp : Q^\sharp × Q^\sharp \rightarrow [lb, ub]$ is the abstract transition matrix, s.t. $\Delta^\sharp(q^\sharp, q'^\sharp) = [lb, ub]$, where $lb$ and $ub$ are the lower and upper bound of a set of concrete probabilistic transitions: $\{\Delta(q, q') \mid q ∈ q^\sharp, q' ∈ q'^\sharp\}$ specified in the relevant concrete model $\mathcal{A}$ respectively;
- $L : Q^\sharp \rightarrow 2^{AP}$ is a labelling function over $Q^\sharp$ that identifies properties of interest.

An abstract path is an execution of the organisation-oriented interval Markov chain.

**Definition 12 (Abstract path).** An abstract path $\omega^\sharp$ is a non-empty sequence of states $q_0^\sharp q_1^\sharp \ldots$, where $q_i^\sharp ∈ Q^\sharp$ and $\forall i. \Delta^\sharp(q_i^\sharp, q_{i+1}^\sharp) \subseteq (0, k]$ where $0 < k ≤ 1$. The set of all finite and infinite paths of $\mathcal{A}_I^\sharp$ starting in state $q_0^\sharp$ are denoted as: $\text{Path}_{\mathcal{A}_I^\sharp}(q_0^\sharp)$ and $\text{Path}_{\mathcal{A}_I^\sharp}(q_0^\sharp)$ respectively.

**Definition 13 (Probability bounds of abstract paths).** The lower (Prob$^{-}$) and upper bound (Prob$^{+}$) of the probability of a finite abstract path $\omega_{fin}^\sharp$ starting from state $q_0^\sharp$ are respectively:

$$\text{Prob}_-^\sharp(\omega_{fin}^\sharp) \triangleq \begin{cases} 1 & \text{if } n = 0 \\ \text{Prob}_-^\sharp(\omega_{fin}^\sharp) & \text{otherwise} \end{cases}$$

$$\text{Prob}_+^\sharp(\omega_{fin}^\sharp) \triangleq \begin{cases} 1 & \text{if } n = 0 \\ \text{Prob}_+^\sharp(\omega_{fin}^\sharp) & \text{otherwise} \end{cases}$$

where $n$ denotes the length of the abstract path, $\omega_i^\sharp$ denotes the $i^{th}$ element of $\omega^\sharp$.

We focus on the reachability properties, for instance, the probability bounds of reaching or moving to an organisation of interests from another.

**Definition 14 (Reachability properties).** Let $\mathcal{A}_I^\sharp$ be an organisation-based interval Markov chain. The lower
and upper bound of the probability of reaching an abstract state \( q'' \) from \( q' \) are computed by:

\[
\text{Reach}^{-}_{A_f}(q', q'') = \min \left( \sum_{\omega^f \in \text{Path}_{A_f}(q', q'')} \text{Prob}_{q'}(\omega^f) | \omega^f_0 = q' \land \exists i \geq 0. \omega^f_i = q'', 1 \right)
\]

\[
\text{Reach}^{+}_{A_f}(q', q'') = \min \left( \sum_{\omega^f \in \text{Path}_{A_f}(q', q'')} \text{Prob}_{q'}(\omega^f) | \omega^f_0 = q' \land \exists i \geq 0. \omega^f_i = q'', 1 \right).
\]

Our organisation-oriented interval Markov chain should safely approximate the concrete CTMC describing the probabilistic behaviours of the system.

**Theorem 1 (Soundness of the abstract semantics).** Let \( A_f^j \) and \( A \) be the coarse-grained model and the relevant concrete model of a reaction network respectively, \( \forall q^j = Q, q'' = Q' \in Q^j \subseteq Q \), we have:

\[
\text{Reach}^{-}_{A_f^j}(q_1^j, q_2^j) \leq \text{Reach}^{-}_{A}(Q, Q'),
\]

\[
\text{Reach}^{+}_{A_f^j}(q_1^j, q_2^j) \geq \text{Reach}^{+}_{A}(Q, Q').
\]

**Proof:** Let \( \omega^j \) denote an abstract path starting from \( q^j \) and reaching \( q'' \). For any \( \omega^f \in \text{Path}_{A_f}(q', q'') \), such as \( \omega^f_0 = q', \omega^f_{\omega_0^f} = q'' \), assume \( |\omega^j| = n \in \mathbb{N} \), and let \( \omega \in \text{Path}_{A}(q, q') \) denote a concrete path starting from a state in \( Q \) and reaching a state \( Q' \), we have:

\[
\text{Reach}^{-}_{A_f}(q_1^j, q_2^j) = \sum_{\omega^f \in \text{Path}_{A_f}(q', q'')} \text{Prob}_{q'}(\omega^f) \leq \sum_{\omega \in \text{Path}_{A}(q, q')} \text{Prob}_{q'}(\omega) = \text{Reach}^{-}_{A}(Q, Q'),
\]

\[
\text{Reach}^{+}_{A_f}(q_1^j, q_2^j) = \sum_{\omega^f \in \text{Path}_{A_f}(q', q'')} \text{Prob}_{q'}(\omega^f) \geq \sum_{\omega \in \text{Path}_{A}(q, q')} \text{Prob}_{q'}(\omega) = \text{Reach}^{+}_{A}(Q, Q').
\]

Similarly, we have \( \text{Reach}^{+}_{A_f}(q^j, q'') \geq \text{Reach}^{+}_{A}(Q, Q') \).

---

**Example 7.** Consider again the reaction network described in Example 6.

- by applying the coarse-grained model shown in Fig. 8, we can calculate the probability of movement from \( q_{(a,b,c,d)}^j \) to \( q_{(a,b)}^j \): \( 0.1506 \); the probability of movement from \( q_{(a,b,c,d)}^j \) to \( q_{(a,c)}^j \): \( 0.2314 \);
- by applying the concrete model shown in Fig. 5 we obtain the probability of movement from \( Q_{(a,b,c,d)} \) to \( Q_{(a)} \): \( 0.1776 \); the probability of movement from \( Q_{(a,b,c,d)} \) to \( Q_{(a)} \): \( [1, 1] \).

Note that our interval coarse-grained model safely approximates the concrete one, but may suffer from over-estimation.

### 7 Predicting System Behaviour over Time

This section presents an application of our organisation-oriented coarse grained model. We address the following problem: given a reaction network and a fixed number of the maximum population of the system, construct the average-based organisation coarse-grained model \( A^\circ \) (focus on the
average transition probabilities between abstract states for simplicity and intuition, this can be replaced by interval-based transitions directly), can we predict the behaviour of the system at any future time using $\mathcal{A}_t^\ast$? Fig. 10 captures the idea of using an organisation-based coarse grained model to approximate the concrete one. In the concrete world, $\mathcal{A}_t$ denotes the concrete model at time $t$, $f$ denotes the dynamical transition function over $\mathcal{A}_t$, and $\mathcal{A}_t^\ast$ denotes the concrete model after $\Delta t$ time units. $g_o$ denotes the organisation based coarse graining function, which maps the concrete model $\mathcal{A}_t$ (c.f. $\mathcal{A}_t^\ast$) to the average coarse-grained model $\mathcal{A}_t^\ast$ (c.f. $\mathcal{A}_t^{\ast\ast}$); $g_o$ denotes the coarse-graining dynamical transition function on $\mathcal{A}_t^\ast$.

### 7.1 Time evolution of the reaction networks via master equations

We apply the traditional “master equation” approach to calculate the stochastic time evolution of the reaction network. We briefly review the main features of the master equation transition function over $\mathcal{A}_t$, and $\mathcal{A}_t + \Delta t$ denotes the concrete model after $\Delta t$ time units; $g_o$ denotes the organisation based coarse graining function, which maps the concrete model $\mathcal{A}_t$ (c.f. $\mathcal{A}_t^\ast$) to the average coarse-grained model $\mathcal{A}_t^\ast$ (c.f. $\mathcal{A}_t^{\ast\ast}$); $f^\ast$ denotes the coarse-graining dynamical transition function on $\mathcal{A}_t^\ast$.

\[
\frac{dP}{dt}(t) = \sum_{j=0,j\neq i}^{n} \frac{1}{E_j} P_j(t).
\]

Sometimes, we use $P_{i|o \in O}(t)$ to denote $P_i(t)$ for readability. We therefore build a set of equations for all $i$. By solving the set of equations, we can obtain the distributions of the system at any future time.

**Example 8.** Consider again Example 8 $\mathcal{M} = \{a, b, c, d\}$, $\mathcal{R} = \{a + b \rightarrow a + 2b, a + d \rightarrow a + 2d, b + c \rightarrow 2c, c \rightarrow b, b + d \rightarrow c, b \rightarrow 0, c \rightarrow 0, d \rightarrow 0\}$. From the coarse-grained model shown in Fig. 8 ($N_{\text{max}} = 5$), we construct the master equations as follows:

\[
\begin{align*}
\frac{dP_{a,b,c,d}(t)}{dt} &= -\frac{1}{0.99} P_{a,b,c,d}(t) \\
\frac{dP_{a,b,c}(t)}{dt} &= -\frac{1}{1.1} P_{a,b,c}(t) + 0.436 - 0.39 \frac{1}{0.99} P_{a,b,c,d}(t) \\
\frac{dP_{a,b}(t)}{dt} &= -\frac{1}{1.08} P_{a,b}(t) + 0.71 - 0.59 \frac{1}{0.99} P_{a,b,c}(t) \\
\frac{dP_{a,c}(t)}{dt} &= -\frac{1}{1.08} P_{a,b,c}(t) + 0.409 - 0.59 \frac{1}{0.99} P_{a,b,c,d}(t) \\
\frac{dP_{a}(t)}{dt} &= -\frac{1}{1.25} P_{a,b,c,d}(t) + 0.184 - 0.73 \frac{1}{0.99} P_{a,b,c}(t) + \frac{1}{125} P_{a,b,c}(t)
\end{align*}
\]

We solve the above equations and show in Fig. 11 a comparison between the time evolution of the reaction network via master equation simulation on (a) the organisation-based average coarse-grained model (see Fig. 8), and (b) the original concrete model (see Fig. 5). The former (6 states) takes less than a second, whereas the latter (126 states) takes approximately 1.5 minutes.

Similarly, Fig. 12 and 13 present experimental results for the case of $N_{\text{max}} = 10$ with different reaction rates. The relevant master equations are:

\[
\begin{align*}
\frac{dP_{a,b,c,d}(t)}{dt} &= -\frac{1}{0.99} P_{a,b,c,d}(t) \\
\frac{dP_{a,b,c}(t)}{dt} &= -\frac{1}{1.1} P_{a,b,c}(t) + 0.4836 - 0.39 \frac{1}{0.99} P_{a,b,c,d}(t) \\
\frac{dP_{a,b}(t)}{dt} &= -\frac{1}{1.08} P_{a,b}(t) + 0.71 - 0.59 \frac{1}{0.99} P_{a,b,c}(t) \\
\frac{dP_{a,c}(t)}{dt} &= -\frac{1}{1.08} P_{a,b,c}(t) + 0.409 - 0.59 \frac{1}{0.99} P_{a,b,c,d}(t) \\
\frac{dP_{a}(t)}{dt} &= -\frac{1}{1.25} P_{a,b,c,d}(t) + 0.184 - 0.73 \frac{1}{0.99} P_{a,b,c}(t) + \frac{1}{125} P_{a,b,c}(t)
\end{align*}
\]

and

\[
\begin{align*}
\frac{dP_{a,b,c,d}(t)}{dt} &= -\frac{1}{0.99} P_{a,b,c,d}(t) \\
\frac{dP_{a,b,c}(t)}{dt} &= -\frac{1}{1.1} P_{a,b,c}(t) + 0.4836 - 0.39 \frac{1}{0.99} P_{a,b,c,d}(t) \\
\frac{dP_{a,b}(t)}{dt} &= -\frac{1}{1.08} P_{a,b}(t) + 0.71 - 0.59 \frac{1}{0.99} P_{a,b,c}(t) \\
\frac{dP_{a,c}(t)}{dt} &= -\frac{1}{1.08} P_{a,b,c}(t) + 0.409 - 0.59 \frac{1}{0.99} P_{a,b,c,d}(t) \\
\frac{dP_{a}(t)}{dt} &= -\frac{1}{1.25} P_{a,b,c,d}(t) + 0.184 - 0.73 \frac{1}{0.99} P_{a,b,c}(t) + \frac{1}{125} P_{a,b,c}(t)
\end{align*}
\]

for rates $R_1$: \{a \rightarrow b, b \rightarrow c, c \rightarrow b, b \rightarrow d, a \rightarrow 2b, c \rightarrow 2c, c \rightarrow 2d\} and $R_2$: \{a \rightarrow 2b, b \rightarrow 4c, c \rightarrow 2c, b \rightarrow 2d, a \rightarrow 2b, c \rightarrow 2c, d \rightarrow 2d\}$ respectively.

\[E_i = \int_0^\infty P_i(t)dt = \int_0^\infty e^{-\alpha_i t} dt = \frac{1}{\alpha_i},\]
Note that the prediction by applying the abstract model produces a similar pattern of the concrete behaviour over time for this case. The precision of the results varies regarding to different models and rates of reaction rules, however the basic pattern of behaviours can be captured but the time and space spent is much smaller.

8 Refining the Coarse Graining

Our organisation-based coarse-grained model helps us to scale the complex model but may lose some precision for analysis and prediction of its time evolution (see the comparison presented in Fig. 12). To address this concern, in this section, we propose a method to refine our coarse-grained model for the purpose of improving the precision of the analysis and the prediction. The idea is that we split each abstract state with respect to the time to reach a stable state from it, since the speed to reach a stable state also heavily affects the behaviours of the reaction network system in the long term. Note that a bottom abstract state is an abstract state of the coarse-grained model without outgoing abstract transitions, which is stable and will stay there forever. A bottom abstract state is essentially a set of bottom SCCs regarding to the internal generators of an organisation, which is called “bottom SCC class”.

Definition 15 (Bottom SCC class). An organisation-based abstract state $q^*$ is called a bottom SCC class if it consists of a set of bottom SCCs.

We propose an algorithm to refine the organisation-based coarse-grained model. Specifically, we make a parti-
tion over the abstract state into a pair, one of which reaches a bottom SCC class quickly while the other reaches it slowly. Thus each abstract state is split into $2^n$ states if there is a path to reach $n$ bottom SCC class. Specifically, we use the k-means ($K = 2$) refinement techniques to make the partitions.

We briefly review the basic concept of k-means clustering \[20\] here for our purpose of refining the abstract states. Given a set of observations \(x_1, x_2, \ldots, x_n\), each of which is a d-dimensional real vector, k-means clustering proposes to partition the $n$ observations into $k(\leq n)$ sets $S = \{S_1, S_2, \ldots, S_k\}$ such that the sum of distance functions of each point in the cluster to the K center is minimised, i.e., it aims to find: \(\arg\min_S \sum_{i=1}^{k} \sum_{x \in S_i} \|x - \mu_i\|^2\), where $\mu_i$ is the mean of points in $S_i$. The basic procedure of the standard k-means algorithm \[19\] alternatively proceeds between two steps until the assignments in the first step no longer change: (i) assign each observation to the cluster whose mean produces the least squared Euclidean distance; and (ii) calculate the new means to be the centroids of the observations in the new clusters.

In our scenario, for all concrete states in an abstract state $q^s \in Q^s$, we calculate the expected time to reach a bottom SCC class, say $q^s_B$, and get a set of observations $S = \{t_1, t_2, \ldots, t_n\}$. k-means clustering partitions the set of observations $S$ into $K = 2$ sets: $S = \{S_1, S_2\}$, such that the sum of distance functions of each point in the cluster to the K center is minimised. The algorithm is formalised in Algorithm 3 and 2. The basic procedure is that, for any $q \in \bar{Q}$, the refined model is presented in Fig. 14 (the coarser model, but for $N = 5$ was shown previously in Fig. 8). To perform a prediction using our refined model, we construct the master equations as follows:

\[
\begin{align*}
\frac{dP_{(a,b,c,d)}}{dt} \times (t) &= -\frac{1}{1.0292} P_{(a,b,c,d)}(t) + \frac{0.4065}{1.411} P_{(a,b,c,d)}(t) + \frac{0.5037}{1.411} P_{(a,b,c,d)}(t) + \frac{0.9064}{1.411} P_{(a,b,c,d)}(t) \\
\frac{dP_{(a,b)}}{dt} \times (t) &= -\frac{1}{1.0292} P_{(a,b)}(t) + \frac{0.4065}{1.411} P_{(a,b)}(t) + \frac{0.5037}{1.411} P_{(a,b)}(t) + \frac{0.9064}{1.411} P_{(a,b)}(t) \\
\frac{dP_{(a,d)}}{dt} \times (t) &= -\frac{1}{1.0292} P_{(a,d)}(t) + \frac{0.4065}{1.411} P_{(a,d)}(t) + \frac{0.5037}{1.411} P_{(a,d)}(t) + \frac{0.9064}{1.411} P_{(a,d)}(t) \\
\frac{dP_{(b,c)}}{dt} \times (t) &= -\frac{1}{1.0292} P_{(b,c)}(t) + \frac{0.4065}{1.411} P_{(b,c)}(t) + \frac{0.5037}{1.411} P_{(b,c)}(t) + \frac{0.9064}{1.411} P_{(b,c)}(t) \\
\frac{dP_{(c,d)}}{dt} \times (t) &= -\frac{1}{1.0292} P_{(c,d)}(t) + \frac{0.4065}{1.411} P_{(c,d)}(t) + \frac{0.5037}{1.411} P_{(c,d)}(t) + \frac{0.9064}{1.411} P_{(c,d)}(t) \\
\frac{dP_{(a,b,c)}}{dt} \times (t) &= -\frac{1}{1.0292} P_{(a,b,c)}(t) + \frac{0.4065}{1.411} P_{(a,b,c)}(t) + \frac{0.5037}{1.411} P_{(a,b,c)}(t) + \frac{0.9064}{1.411} P_{(a,b,c)}(t) \\
\frac{dP_{(a,b,c,d)}}{dt} \times (t) &= -\frac{1}{1.0292} P_{(a,b,c,d)}(t) + \frac{0.4065}{1.411} P_{(a,b,c,d)}(t) + \frac{0.5037}{1.411} P_{(a,b,c,d)}(t) + \frac{0.9064}{1.411} P_{(a,b,c,d)}(t)
\end{align*}
\]

Fig. 14: Example 6 $N_{\text{max}} = 10$, refined coarse-grained model

9 CONCLUSIONS

This work investigates the combination of chemical organisation theory and probabilistic model checking for the
Algorithm 3: Refining the coarse-grained model

| Data: Organisation-based coarse-grained model $A^2(Q^2, Q^2, \Delta^2, L)$ |
| Result: Refined model $A^2_p(Q^2, Q^2, \Delta^2, L_p)$ |

$Q^2_p = Q^2 \cap \{\} ;$

$\Delta^2_p = \{\} ;$

$L_p = \{\} ;$

for each bottom SCC $q^2_B \in Q^2_B$

for each abstract state $q^2 \in (Q^2 \setminus q^2_B)$ do

\[(q^{2_p}, q^2_B) \leftarrow \text{partitionAbsState}(q^2, q^2_B) ; \]

$Q^2_p \cup \{q^{2_p}\} \cup \{q^2_B\} ;$

$L^1(q^{2_p}) = \{L(q) \mid q \in q^{2_p}\} ;$

$L^1(q^2_B) = \{L(q) \mid q \in q^2_B\} ;$

if $q^{2_p} \subseteq Q^2_p \wedge (q^2_B \neq \{\})$ then

\[Q^{2_p} \cup q^{2_p} ;\]

end

if $q^{2_p} \subseteq Q^2_p \wedge (q^2_B \neq \{\})$ then

\[Q^{2_p} \cup q^2_B ;\]

end

end

$\Delta^2_p \leftarrow \text{computeProbBetOrgs}(Q^2_p);$

return $A^2_p$.

---

**Fig. 15**: Organisation k-means clustering dynamics prediction via master equation simulation over the average coarse-grained model (a) and the concrete analysis over the original model (b) of Example 6, $N_{\max} = 10$.

---

The authors acknowledge support from the European Union through funding under FP7–ICT–2011–8 project HIERATIC.
We also thank the anonymous reviewers for their helpful and detailed comments.

References


Chunyan Mu Chunyan Mu is a Senior Lecturer in Computer Science at Teesside University She obtained her PhD from King’s College London in 2010, and worked in different roles as a post-doctoral researcher at UCL, Paris 7, and University of Birmingham before joining Teesside University in 2016. Her research interests include formal methods and security, information flow, language-based security, and quantitative verification.

Peter Dittrich Peter Dittrich (Prof. Dr. habil.) is head of the Bio Systems Analysis Group at the Institute for Computer Science, Friedrich Schiller University Jena; member of the Jena Centre for Bioinformatics and the Michael Stifel Centre for Data-Driven and Simulation Science. He is responsible for the B.Sc./M.Sc. program in Bioinformatics at the FSU Jena, ERASMUS coordinator, and faculty member of the graduate schools: HIGH-RADE (Helmholtz Centre for Environmental Research), IMPRS-gBGC (International Max Planck Research School for Global Biogeochmical Cycles), and Jena School of Microbial Communication (JSMC). He obtained his Habilitation from the Department of Mathematics and Computer Science of the FSU Jena in 2007 and his Doctoral degree from the University of Dortmund in 2001. His research interests are in systems chemistry, systems biology, chemical computing, and systems theory.

David Parker David Parker is a Reader in Computer Science at the University of Birmingham. He completed his PhD in 2003, also at the University of Birmingham, and then worked as a post-doctoral researcher, first at Birmingham and then at the University of Oxford, between 2007 and 2012. His main research interests are in the area of formal verification, with a particular focus on the analysis of quantitative aspects such as probabilistic and real-time behaviour, and he has published over 100 papers in this area. He also leads the development of the widely used probabilistic verification tool PRISM.

Jonathan E. Rowe Jonathan Rowe is a Professor in Natural Computation at the University of Birmingham. He obtained his PhD in 1991 from the University of Exeter and worked in industry for a couple of years. He then returned to academia where, amongst other things, such as multi-agent systems, artificial life and other complex adaptive systems he has studied the theory of genetic and other evolutionary algorithms. In addition to publishing many papers on the subjects, he co-authored the graduate-level text book Genetic Algorithms: Principles and Perspectives with Colin Reeves. He has also helped organize several workshops and conferences in the field, including “Parallel Problem Solving from Nature”, “Foundations of Genetic Algorithms” (FOGA) and the Dagstuhl seminars on the Theory of Evolutionary Computation.