Trend-based Analysis of a Population Model of the AKAP Scaffold Protein

Oana Andrei and Muffy Calder

School of Computing Science, University of Glasgow, G12 8RZ, UK,
Oana.Andrei@glasgow.ac.uk,
Muffy.Calder@glasgow.ac.uk

Abstract. We formalise a continuous-time Markov chain with multi-dimensional discrete state space model of the AKAP scaffold protein as a crosstalk mediator between two biochemical signalling pathways. The analysis by temporal properties of the AKAP model requires reasoning about whether the counts of individuals of the same type (species) are increasing or decreasing. For this purpose we propose the concept of stochastic trends based on formulating the probabilities of transitions that increase (resp. decrease) the counts of individuals of the same type, and express these probabilities as formulae such that the state space of the model is not altered. We define a number of stochastic trend formulae (e.g. weakly increasing, strictly increasing, weakly decreasing, etc.) and use them to extend the set of state formulae of Continuous Stochastic Logic. We show how stochastic trends can be implemented in a guarded-command style specification language for transition systems. We illustrate the application of stochastic trends with numerous small examples and then we analyse the AKAP model in order to characterise and show causality and pulsating behaviours in this biochemical system.

1 Introduction

In the recent years biochemical networks have become an important application area for modelling approaches and analysis techniques developed in theoretical computer science. Our approach to modelling and analysing biochemical networks is stochastic processes, continuous-time Markov chains (CTMCs) in particular, which allow new quantitative analysis in addition to the traditional simulation afforded by ordinary differential equations (ODEs). CTMC models where states represent the counts of molecules for each biochemical species, also called molecular CMTCs, together with the behaviour analysis based on Gillespie's stochastic simulation algorithm [1], provide a faithful representation of biochemical networks. One major limitation of the molecular CTMCs is the size of the underlying state space that can easily become too large to be handled explicitly by stochastic model checking tools. CTMCs with levels [2] are based on discrete levels of concentration instead of exact molecule counts. In comparison to the molecular CTMC, the level abstraction reduces the state space, leading to models that are more amenable to model checking techniques that analyse the

entire state space. Another limitation of the molecular CTMCs is the need for precise molecular concentrations for the species and details about the reactions, whereas CTMCs with levels allow for greater abstraction and relative quantities.

We focus here on modelling the scaffold protein AKAP and its role as a mediator of the crosstalk between the cyclic AMP (cAMP) and the Raf-1/MEK/ERK signalling pathways. The behaviour of this biochemical system is complex and still under study in the laboratory. Following discussions with laboratory scientists, we have developed a CTMC with levels model, which we believe to be the first formal model of the system. This modelling paradigm is well-suited to the AKAP system because the experimental data gathered so far are relative rather than exact. In other words, exact rates of reactions are unknown, but their relative rates are known; for example, some are known to be about three times faster than others, etc. Typical questions and properties conjectured by laboratory scientists include if *increasing concentration levels of molecule A lead to decreasing concentration levels of molecules B and C*, or confirming the *pulsating behaviour suggested by the lab experiments*. In order to formalise these conjectured properties in the AKAP model we define *stochastic trends*.

Stochastic trends stem from modelling biochemical networks but they can be more generally applied to Markov Population Processes (MPPs) – continuous-time Markov chains where states record the counts of individuals in each *colony* of a *population* [3–6]. MPPs can be used for modelling in a wide variety of application domains, including, for example, computer networks, chemical reactions networks, and ecology networks. Birth-death processes are simple MPPs. In particular, molecular CMTCs and CTMCs with levels are examples of MPPs. Many key questions to ask of Markov population models involve *trends*. For example, is a particular colony *increasing/decreasing*, is the change *strict, weak, etc.*, or if we get *more* individuals in colony A, will colony B then *decrease*? Analysis of such logical properties by model checking requires a suitable representation of trends. We propose an approach based on formulating the probabilities of transitions that increase (resp. decrease) colony counts in a stochastic model.

Related work. The concept of a trend in a discrete or continuous deterministic setting is well established (e.g. slope or first-order derivative), but less so in a stochastic setting. First-order derivatives have been considered previously in the context of model checking biochemical systems. For example in BIOCHAM [7, 8], oscillatory properties are analysed using queries expressed as formulae in LTL with constraints over real numbers. Such formulae are interpreted over traces of states and a state includes not only the concentration value of each molecular species but also the value of its first-order derivative. This analysis applies to BIOCHAM *deterministic* semantics, where the underlying model has exactly one trace and therefore the concept of a trend is encapsulated by the first-order derivative. In this paper we consider the concept of trend in a stochastic setting, and without explicitly storing the trend in a state variable. Oscillating behaviours can be formulated either as temporal formulae [9–11] in CTL, PCTL or CSL or based on a system of differential equations [12]. However, for the AKAP model we have to deal with incomplete data about the reaction rates.

Stochastic trends provide a preliminary analysis technique when only partial information is provided on the reaction rates such as a reaction rate is of the order of some other reaction rate. Trend formulae are very closely related to the trend variable approach [13]. One advantage of trend formulae over trend variables is that the use of trend formulae does not increase the size of the state space. Moreover, our analysis is forward-looking, or *a priori*, based on the probability (over all possible transitions) for a colony to increase (resp. decrease), whereas trend variables imply an *a posteriori* analysis based on behaviour that has already occurred. In Sect. 5.3 we will give an in depth comparison between trend formulae and trend variables.

Contributions. This paper is an extension of previous work [14] and focuses on introducing stochastic trends as an analysis technique for MPPs in general, models of biochemical networks in particular. The contributions of the paper are twofold:

- Stochastic trend formulae for characterising the probability of increasing or decreasing colonies that can be used to extend the set of state formulae in temporal logics such as Continuous Stochastic Logic, along with an encoding of trend formulae in the guarded-command modelling language of the PRISM probabilistic model checker.
- A CTMC with level model of the AKAP scaffold protein as a mediator of the crosstalk between the cyclic AMP and the Raf-1/MEK/ERK signalling pathways, and the use of stochastic trends to characterise causality and pulsating behaviours in the AKAP model.

Outline. The next section reviews the definition and basic concepts of continuous-time Markov chains (CTMCs), Markov population processes (MPPs) and CTMCs with levels. We also review the reagent-centric modelling style of MPPs, biochemical systems in particular, and their representation in the modelling language of the PRISM model checker, and the temporal logic Continuous Stochastic Logic for expressing properties about CTMCs in PRISM. Section 3 presents the biological model of the AKAP scaffold protein and in Sect. 4 we define the associated CTMC with level model. We introduce stochastic trends in Sect. 5 and use them for analysing the behaviour of the AKAP system in Sect. 6. We give our conclusions and directions for future work in Sect. 7.

2 Preliminaries

In the following, we assume some familiarity with continuous-time Markov chains, see for example [15–17].

2.1 Continuous-time Markov Chains

Definition 2.1. *A* (labelled) continuous-time Markov chain *(CTMC) is a tuple* (S, s_0, R, L) *where* S *is a countable set of states,* $s_0 \in S$ *the initial state,*

$R : S \times S \to \mathbb{R}_{\geq 0}$ the transition rate matrix, AP a finite set of atomic propositions, and $L : S \to 2^{AP}$ the labelling function associating to each state in S the set of atomic propositions from AP that are valid in that state.

The transition rates determine the probability of transitions to be completed within a certain amount of time following the negative exponential distribution: when $R(s, s') > 0$, then the probability of this transition to be triggered within t time units equals $1 - e^{-R(s,s') \cdot t}$. The time spent in state s before any transition is triggered is exponentially distributed with parameter:

$$E(s) = \sum_{s' \in S} R(s, s').$$

$E(s)$ is called the *exit rate* of state s. For a given state s, there is a race between outgoing transitions from s if there are more than one state s' such that $R(s, s') > 0$. If the exit rate of a state is equal to 0 then no transition can be fired from it and the state is called *absorbing*. The time-abstract probability of a state s' to be the next state to which a transition is made from state s is computed by a *transition probability function* $\mathbf{P} : S \times S \to [0, 1]$ as follows:

$$\mathbf{P}(s, s') = \begin{cases} \frac{R(s,s')}{E(s)} & \text{if } E(s) \neq 0 \\ 1 & \text{if } E(s) = 0 \text{ and } s = s' \\ 0 & \text{otherwise} \end{cases}$$

This transition probability function, together with the state space S, initial state s_0 and labelling function L define a discrete-time Markov chain (DTMC) embedded in the CTMC.

An infinite *path* of a CTMC is a sequence $s_0 t_0 s_1 t_1 \ldots$ such that $R(s_i, s_{i+1}) > 0$ and $t_i \in \mathbb{R}_{>0}$ denotes the time spent in state s_i for all $i \geq 0$. A finite path is a sequence $s_0 t_0 s_1 t_1 \ldots s_{k-1} t_{k-1} s_k$ such that s_k is an absorbing state. A *self-loop* transition is a single transition going back to the same state it fired from. A *cycle* is a path beginning and ending with the same state.

2.2 Markov Population Processes

A *population* is a collection of individuals grouped into *colonies* or categories based on common features. Markovian population processes (MPPs) [3–6] are continuous-time Markov chains that express demographic processes such as birth and immigration (addition of individuals), death (removal of individuals) or emigration (transfer of individuals between colonies). The characteristic feature of MPPs is given by their states which enumerate the counts of individuals in every colony.

Definition 2.2. *A* Markov population process (MPP) *is a continuous-time Markov chain $\mathcal{M} = (S, s_0, R, AP, L)$ with S defined as a set of n-dimensional states of the form $s = (x_1, \ldots, x_n)$ with $n \geq 1$ the number of colonies in the population and x_i a non-negative integer representing the number of individuals in colony i, for all i, $1 \leq i \leq n$.*

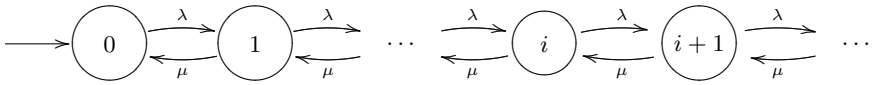

Fig. 1. An $M/M/1$ queue with initial state 0, user arrival rate λ and user departure rate μ.

We define an MPP either *component-wise* according to Def. 2.2, through a *state-transition graph* if the state space is relatively small, or via a *set of reactions* if we model a biochemical network. The graphical state-transition notation we use for an MPP is the usual one for CTMCs: a directed graph with states as nodes and an edge between any pair of nodes s_i and s_j if $R(s_i, s_j) > 0$ with $R(s_i, s_j)$ the edge label; the initial state is marked by an incoming arrow with no source.

Example 2.1. The simplest example of an MPP is a *birth-death process* (BD process) defined as an MPP with one colony. In a BD process states can be indexed by non-negative integers representing the counts of individuals in the single-colony population such that state transitions occur only between neighbouring states: from i to $i+1$ or from $i+1$ to i. One straightforward application of BD processes is in queueing theory. A BD process is an example of a single server queue with an infinite buffer size – also known as the $M/M/1$ *queue* in Kendall's notation [18, 19] if the user arrival (birth) rate λ and user departure (death) rate μ are independent of the colony size. Then each state represents the number of users in the system. In Fig. 1 we depict the MPP model of such a queue.

Another application domain for MPPs is *biochemical networks*. In these networks, the species quantities are usually given in terms of concentrations. Given a biochemical network represented as a set of reactions and initial concentrations for each species, we can associate an MPP model with as many colonies as the number of different species and where both chemical species and reaction rates are expressed in terms of number of molecules, assuming that all chemical species are in the same static compartment (i.e. of constant volume V). This type of MPP model is usually referred to as *molecular CTMC* because we count the molecules. We translate a concentration c for a species X to a number of molecules equal to $C = c \cdot V \cdot N_A$ where N_A is Avogadro's number (the number of molecules contained in a mole of X).

A reaction is usually given by a stoichiometric equation:

$$r: \ \alpha_1 X_1 + \ldots + \alpha_n X_n \xrightarrow{k} \beta_1 X_1 + \ldots + \beta_n X_n \tag{1}$$

where, for all i, $1 \leq i \leq n$, non-negative integers α_i and β_i are the stoichiometric coefficients defining how many molecules of X_i are consumed and produced respectively by the reaction, k is the constant reaction rate coefficient. The species on the left and right hand side with non-zero coefficients are called *reactants* and *products* respectively. In practise we do not include species with null stoichiometric coefficients in a stoichiometric equation. Let X_i^{max} denote

the upper bound on the number of molecules X_i. Such bounds can either be obtained from experimental data or estimated by using stochastic simulation and model checking in tandem [20], or simply imposed for any species that grows infinitely (in order to guarantee a finite CTMC). The reaction can occur from a state s if $s - (\alpha_1, \ldots, \alpha_n) \geq \mathbf{0}$ and $s - (\alpha_1, \ldots, \alpha_n) + (\beta_1, \ldots, \beta_n) \leq (X_1^{max}, \ldots, X_n^{max})$, where X_i^{max} is the maximum possible number of molecules X_i. If a transition from s is taken according to this reaction, then we move to the state $s' = s - (\alpha_1, \ldots, \alpha_n) + (\beta_1, \ldots, \beta_n)$. Assuming mass-action kinetics, the transition rate is proportional to the number of affected molecules and equals $k \cdot \prod_{1 \leq i \leq n} \binom{C_i}{\alpha_i}$, with C_i denoting the number of molecules of species X_i, since we need to consider all possible combinations of individual molecules.

The combinatorics of every possible molecular count in a molecular CTMC can lead to state space explosion. Molecular CTMC models can be too large to analyse using model checking and only an analysis based on stochastic simulation becomes available, which does not construct the complete underlying state space. One way of tackling this problem is to discretise each species concentration uniformly into a number of *levels of concentration*, rather than representing by numbers of molecules. A transition from one state to another reflects changes of these levels according to a biochemical reaction. The result is a stochastic, population based model that is more abstract than the molecular CTMC and called *continuous-time Markov chain with levels* [21, 22, 2]. One advantage of using CTMC with levels is that it allows one to deal with incomplete or only relative information about molecular concentrations, often the case in experimental settings. Another advantage of CTMCs with levels over the molecular CTMC is its smaller state space, allowing the models to be more amenable to stochastic model checking.

Informally, in a CTMC with levels each species is characterised by a number of levels, equidistant from each other, with step size h. We assume that all species have the same step size. We assign to each species different concentration levels, from 0 (corresponding to null concentration) to a maximum number N. When the maximum molar concentration is M, then the step size $h = \frac{M}{N}$. Here, we assume all reactions have mass-action kinetics.

Definition 2.3 (CTMC with levels). *A* CTMC with levels *for a biochemical system is an MPP where the molecules of the same species form a colony and states represent levels of concentrations of the species. For n different species $(X_i)_{1 \leq i \leq n}$, a state is a tuple $s = (\ell_1, \ell_2, \ldots, \ell_n)$ with ℓ_i the discrete concentration level for the species X_i, for all i, $1 \leq i \leq n$. A reaction of the form given by Eq. 1 has similar firing conditions as in the case of molecular CTMCs and the rate of a transition fired by such a reaction is the product of the reaction rate coefficient adjusted for the step-size discretisation h and the concentrations of the reacting species, i.e., $\frac{k}{h} \cdot (\ell_1^{\alpha_1} \cdot h) \cdot \ldots \cdot (\ell_n^{\alpha_n} \cdot h)$, where k is the constant reaction rate coefficient and $\ell_i^{\alpha_i}$ is the discrete level of concentration of reacting species X_i with stoichiometric coefficient α_i.*

In comparison with molecular CTMCs and traditional ordinary differential equations (ODEs), CTMCs with levels models are more compact than molecular

CMTCs yet retain the stochasticity lost in the ODEs. The granularity of a CTMC with levels can be changed by decreasing the stepsize. As the stepsize decreases with the number of levels tending to infinity, the variability of the CTMC with levels model is reduced and, as predicted by Kurtz's Theorem [23] (on the relationship of the class of density dependent Markov chains and a set of ODEs), the obtained global behaviour of the CTMC with levels model tends towards that given by the ODE model [2].

A biochemical reaction does not take only the simple form of an arrival, departure or transfer event between colonies as the definition of MPP. Often there is a form of cooperative transfer from some colonies to others. An example is the following where species are transferred to and from X_1 and X_2, and X_3.

Example 2.2. Consider a simple reaction system consisting of three species X_1, X_2 and X_3 with initial molar concentrations $X_1(0) = X_2(0) = 2\,mol/l$ and $X_3(0) = 0\,mol/l$, and a forward and a backward reaction $X_1 + X_2 \xrightarrow{k_1} X_3$, $X_3 \xrightarrow{k_2} X_1 + X_2$ with $k_1 = 1.2$, $k_2 = 0.2$. If we consider $N = 3$ the maximum number of levels of concentration, the step size is $h = \frac{2}{3}\,mol/l$. The CTMC with levels modelling this system is represented in Fig. 2 with the initial state representing the initial concentration levels given by $(\lfloor \frac{X_1(0)}{h} \rfloor, \lfloor \frac{X_2(0)}{h} \rfloor, \lfloor \frac{X_3(0)}{h} \rfloor) \rfloor = (3, 3, 0)$. We convert a molar concentration $X_i(0)$ to a number of levels $\lfloor \frac{X_i(0)}{h} \rfloor$.

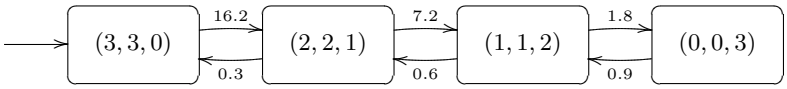

Fig. 2. CTMC with levels for the forward/backward reaction $X_1 + X_2 \overset{k_1/k_2}{\longleftrightarrow} X_3$

2.3 Modelling MPPs in PRISM

There are several languages or formalisms for specifying Markov population processes based on rate transition matrix descriptions, state-transition graphs and stoichiometric equations for chemical reactions. Several other formalisms are available for this purpose as overviewed in [6], including *guarded command models* (GCM). GCMs are textual models describing the classes of possible state transitions on colonies and take inspiration from Dijkstra's guarded-command language (GCL). Reactive Modules [24] and PRISM's specification language [25] are based on the same formalism.

We adopt the *reagent-centric modelling approach* to modelling biochemical systems [26] implemented as a PRISM specification as follows. Each of the colonies, also called reagents, in an interaction or transition is mapped to a process, whose variation reflects increase or decrease, e.g., through production or consumption, through birth, death or migration, etc. For example, the chemical reaction r_1 given in stoichiometric notation by $X_1 + X_2 \xrightarrow{k_1} X_3$ refers to

three reagents and so it is modelled by three processes, X_1, X_2 and X_3, which are then composed concurrently, synchronising on the event r_1. If we assume an underlying semantics of CTMC with levels, after the event r_1, the concentration level of X_3 is increased and those of X_1 and X_2 are decreased.

The PRISM language includes modules with local variables, action-labelled guarded commands (transitions) and multiway synchronisation of modules. Each process is implemented by a module, and the modules are composed using the multiway synchronisation operator (denoted by `||`) over all common actions.

We illustrate the approach with the biochemical system introduced in Example 2.2. The PRISM model, depicted in Fig. 3, has three modules `X1`, `X2` and `X3`, one for each species, all modules running concurrently. Each module has the form: a state variable denoting the species concentration level, followed by commands labelled by the reactions in which the species is a reactant or product. In this example there are two commands labelled by `r1` and `r2`. Each command has the form:

```
[label] guard -> rate : update;
```

meaning that the module makes a transition to a state described by the *update* at the given *rate* when the *guard* is true (the label is optional). The `r1`-labelled command in the first two modules decreases the number of levels by 1 and in the third module increases the number of levels by 1. Initially, there are `N` levels of `X1` and `X2` and 0 levels of `X3`. The module `Const` consists of commands labelled by the reaction labels with trivial guards and updates and the rate equal to the constant reaction rate coefficient. All `r1`-labelled transitions synchronise and the resulting transition occurs with a rate equal to the product of the individual rates, i.e. `(k1/h)*(X1*h)*(X2*h)`.

2.4 Stochastic Model Checking

Since MPPs are CTMCs, we use Continuous Stochastic Logic (CSL) [15] as a temporal logic for specifying properties about their stochastic behaviour. CSL is a stochastic extension of the Computational Tree Logic (CTL) allowing one to express a probability measure of the satisfaction of a temporal property in either transient or in steady-state behaviours. The formulae of CSL are state formulae and their syntax is the following:

$$\begin{aligned} &\textit{State formula } \Phi ::= \textit{true} \mid a \mid \neg\Phi \mid \Phi \wedge \Phi \mid \mathcal{P}_{\bowtie p}[\Psi] \mid \mathcal{S}_{\bowtie p}[\Psi] \\ &\textit{Path formula } \Psi ::= \mathbf{X}\,\Phi \mid \Phi\,\mathbf{U}^{\mathrm{I}}\,\Phi \end{aligned}$$

where a ranges over a set of atomic propositions AP, $\bowtie \in \{\leq, <, \geq, >\}$, $p \in [0,1]$, and I is an interval of $\mathbb{R}_{\geq 0}$. There are two types of CSL properties: transient (of the form $\mathcal{P}_{\bowtie p}[\Psi]$) and steady-state (of the form $\mathcal{S}_{\bowtie p}[\Psi]$). A formula $\mathcal{P}_{\bowtie p}[\Psi]$ is true in state s, denoted by $s \models \mathcal{P}_{\bowtie p}[\Psi]$, if the probability that Ψ is satisfied by the paths starting from state s meets the bound $\bowtie p$. A formula $\mathcal{S}_{\bowtie p}[\Psi]$ is true in a state s if the steady-state (long-run) probability of being in a state which satisfies Ψ meets the bound $\bowtie p$. The path formulae are constructed using the

```
ctmc

const double max_conc = 2;
const int N = 4;
const double h = max_conc/N;
const double k1 = 1.2;
const double k2 = 0.2;

module X1
  X1 : [0..N] init N;
  [r1] (X1>0) -> (X1*h) : (X1'=X1-1);
  [r2] (X1<N) -> (1) : (X1'=X1+1);
endmodule

module X2
  X2 : [0..N] init N;
  [r1] (X2>0) -> (X2*h) : (X2'=X2-1);
  [r2] (X2<N) -> (1) : (X2'=X2+1);
endmodule
```

```
module X3
  X3 : [0..N] init 0;
  [r1] (X3<N) -> (1) : (X3'=X3+1);
  [r2] (X3>0) -> (X3*h) : (X3'=X3-1);
endmodule

module Const
  [r1] true -> (k1/h) : true;
  [r2] true -> (k2/h) : true;
endmodule

system
      X1 || X2 ||  X3 || Const
endsystem
```

Fig. 3. PRISM program for the forward/backward reaction $X1 + X2 \overset{k_1/k_2}{\longleftrightarrow} X3$

$\mathbf{X}$ (next) operator and the $\mathbf{U}^{\mathrm{I}}$ (time-bounded until) operator. Informally, the path formula $\mathbf{X}\,\Phi$ is true on a path starting in s if Φ is satisfied in the next state following s in the path, whereas $\Phi_1\,\mathbf{U}^{\mathrm{I}}\,\Phi_2$ is true on a path ω if Φ_2 holds at some time instant in the interval I in a state s' in ω and at all preceding time instants Φ_1 holds. This is a minimal set of operators for CSL. The operators *false*, disjunction and implication can be derived using basic logical equivalences. Two more path operators are available as syntactic sugar:

- the *eventually* operator $\mathbf{F}$ (future) where $\mathbf{F}^{\mathrm{I}}\,\Phi \equiv \mathit{true}\,\mathbf{U}^{\mathrm{I}}\,\Phi$, and
- the *always* operator $\mathbf{G}$ (globally) where $\mathbf{G}^{\mathrm{I}}\,\Phi \equiv \neg(\mathbf{F}^{\mathrm{I}}\,\neg\Phi)$.

If $I = [0, \infty)$, then the temporal operators $\mathbf{U}$, $\mathbf{F}$, $\mathbf{G}$ are no longer time-bounded, hence we omit the interval superscript notation in this situation.

The model checking problem of a state formula Φ being satisfied in an MPP is denoted by $M, s_0 \models \Phi$. We omit the initial state s_0 when it is obvious.

The PRISM probabilistic model checker [17] has a property specification language based on the temporal logics PCTL, CSL, LTL and PCTL*, including extensions for quantitative specifications and rewards. PRISM allows one to express a probability measure that a temporal formula is satisfied. The bound $\bowtie p$ may not be specified, in which case a probability is calculated in PRISM. Thus these two additional properties $\mathcal{P}_{=?}[\Psi]$ and $\mathcal{S}_{=?}[\Psi]$ are available: the results of the verification of such formulae are the expected probabilities for the satisfaction of the path formula denoted by Ψ.

3 The AKAP Scaffold Protein

In this section we give an overview of the AKAP scaffold protein and its mediating role in the crosstalk between the cAMP and Raf-1/ERK/MEK signalling pathways. The behaviour of this system is complex and still under study in the laboratory.

In intracellular signal transduction pathways, scaffolds are proteins exhibiting two main functions [27]. Namely, a scaffold protein *anchors* particular proteins in specific intracellular locations for receiving signals or transmitting them, and it provides a catalytic function by increasing the output of a signalling cascade or decreasing the response time for a faster output under certain circumstances.

3.1 Species

Figure 4 illustrates the *species* involved in the biochemical system and their interactions in the AKAP model with emphasis on the AKAP's anchoring role as positions on the scaffold are filled or unfilled. The species involved are: cyclic adenosine monophosphate (cAMP); protein kinase A (PKA); Raf-1 with two phosphorylation sites of interest, Serine 338 (Ser338) and Serine 259 (Ser259); phosphodiesterase 8 (PDE8A1); phosphatase PP. The left-hand side of Fig. 4 shows an unfilled scaffold with free PDE8A1, and the right-hand side shows a filled scaffold.

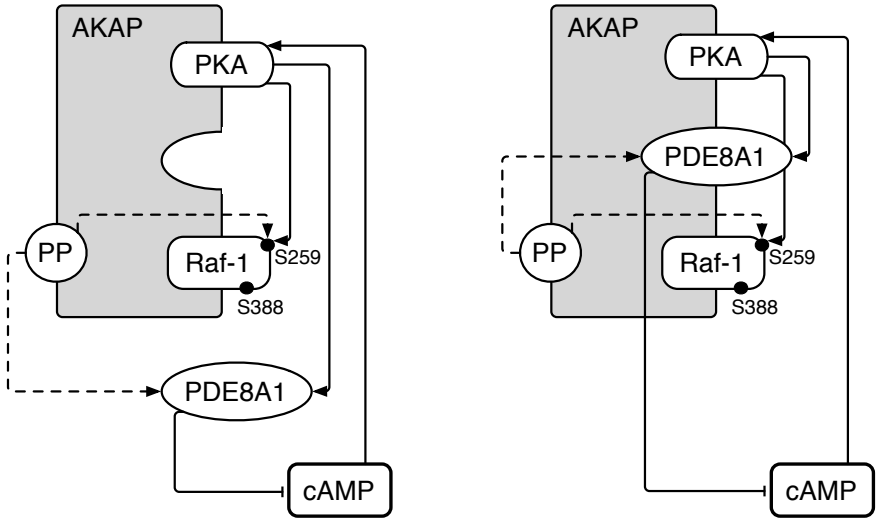

Fig. 4. Interactions between cAMP, unfilled AKAP scaffold, free PDE8A1 and filled scaffold where: $A \to B$ means A activates or phosphorylates B, $A \dashrightarrow B$ stands for A dephosphorylates B, $A \dashv B$ means A degrades B.

3.2 Behaviour

If the concentration of cAMP rises above a given threshold, cAMP activates PKA by binding to it. Activated PKA catalyses the transfer of phosphates to the phosphorylation site Ser259 of Raf-1. The site Ser338 of Raf-1 is said to be inhibited when Ser259 is phosphorylated. Only when Ser338 becomes phosphorylated, the pathway Raf-1/MEK/ERK is activated (and say that Raf-1 is active) and the signalling cascade begins.

The catalytic function of PKA sometimes couples with the AKAP, by binding PKA together with phosphodiesterase PDE8A1 on the scaffold to form a complex that functions as a signal module. Under these conditions, as the cell is stimulated, cAMP activates PKA, and then PKA is responsible for the activation of PDE8A1 (by phosphorylation). PDE8A1 degrades cAMP, but if phosphorylated, PDE8A1 degrades more cAMP, hence rapidly reducing the amount of cAMP that can activate PKA. This leads to a feedback mechanism for downregulating PKA.

Discussions with laboratory scientists revealed the following expectations, or conjectures, about the AKAP system behaviour.

Causal relation between concentration fluctuations. We define *causality* to mean: assuming *more* A (*less* A) denotes increasing (resp. decreasing) concentration levels for a species A, the implication "*more* $A \Rightarrow$ *less* B" means that a decrease in B's concentration level is necessarily preceded by an increase in A's concentration level. Laboratory scientists expect that increasing concentration level of phosphorylated PDE8A1 leads to a cascade of changes in the concentration levels of the other reactants: decreasing concentration levels of cAMP and active PKA, and an increase in the activity of Raf-1 – due to lower levels of phosphorylated Raf-1 at site Ser259. Informally, we express this causality relation by the following relationship:

$$\textit{more}\ \mathsf{pPDE8A1}\ \Rightarrow \textit{less}\ \mathsf{cAMP}\ \Rightarrow \textit{less active}\ \mathsf{PKA}\ \Rightarrow \textit{more active}\ \mathsf{Raf\text{-}1}$$

Pulsating behaviour. Time courses from laboratory experiments suggest the presence of a pulsating behaviour in the system. The pulsations ensure that the state of the Raf-1 pathways alternates between active and inactive, which is a desirable behaviour because very long periods of activity or inactivity may increase the risk of disease. In the current model we do not consider explicitly interactions between cAMP and Raf-1. However, the system is not closed and we include an exogenous interaction represented by the diffusion of cAMP. We conjecture this makes the system exhibit a pulsating behaviour corresponding to the feedback mechanism for the downregulation of PKA, coupled with the diffusion of cAMP. Note that we call such a behaviour *pulsating*, not *oscillating*: oscillation assumes fluctuation around a given value, but the current partial data do not provide us with such a value.

4 MPP Model for the AKAP System

We define a CTMC with levels model for the AKAP system based on *combinations* of the species represented in Fig. 4. An overview of the model follows.

4.1 Scaffolded Species

The AKAP scaffold has three positions to be filled in order by PKA, site Ser259 of Raf-1 and PDE8A1 respectively, with the third one not necessarily filled. We define an abstraction over these species consisting of different combinations in order to encode the context of reactions as follows:

- for filled scaffold: S[PKA *'s state*][Ser259 *'s state*][PDE8A1 *'s state*]
- for unfilled scaffold: S[PKA *'s state*][Ser259's state].

where each state has a binary representation with 1 representing activated or phosphorylated and 0 otherwise. For instance, S100 represents a filled AKAP scaffold with active PKA and unphosphorylated Ser259 and PDE8A1, whereas S01 represents an unfilled scaffold with inactive PKA and phosphorylated Ser259. All the possible abstract species involving a scaffold are: S00, S10, S01, S11, S000, S100, S101, S110, S011, S010, S001, S111.

We also distinguish between PDE8A1 and its phosphorylated form PDE8A1 (pPDE8A1) as two different unscaffolded species. The same reasoning applies to the phosphatase PP anchored on a filled scaffold and PP on an unfilled scaffold (denoted by uPP). The remaining unscaffolded species is cAMP.

4.2 Biochemical Reactions

In Fig. 5 we list the biochemical reactions of the model. Each reaction is given by a stoichiometric equation with explicit reference to the scaffold positions (the underlying reactions have mass-action kinetics). We associate reaction rate constants (from r_1 to r_{26}) with each biochemical reaction.

The existing experimental data suggest only approximate ratios of the reaction rates. More precisely, we have some information on the ratio between the rate of PKA phosphorylating Raf-1 at site Ser259 and PDE8A1 (either on the scaffold or not). On unfilled scaffolds, PKA phosphorylates three times less unscaffolded PDE8A1 than Raf-1 at site Ser259 from the same scaffold. On filled scaffolds, PKA phosphorylates Raf-1 at Ser259 and PDE8A1 at the same rate. Consequently the relation between constant rates of the reactions involving PKA phosphorylating either PDE8A1 or Raf-1 is: $r_4 = r_5 = r_6 = r_{10} = r_{11} = 3 \cdot r_{12} = 3 \cdot r_{13}$. In addition, phosphorylated PDE8A1 degrades about three times more cAMP than PDE8A1 does, hence we deduce the following ratios between the constants rates of the reactions where PDE8A1 degrades cAMP: $r_{19} = r_{20} = r_{21} = r_{22} = r_{23} = r_{24} = 3 \cdot r_{25} = 9 \cdot r_{26}$.

cAMP **diffusion:**	PDE8A1 **phosphorylation:**
$\_ \xrightarrow{r_1}$ cAMP	S100 $\xrightarrow{r_{10}}$ S101
	S110 $\xrightarrow{r_{11}}$ S111
PKA **activation:**	S10 + PDE8A1 $\xrightarrow{r_{12}}$ S10 + pPDE8A1
S000 + cAMP $\xrightarrow{r_2}$ S100	S11 + PDE8A1 $\xrightarrow{r_{13}}$ S11 + pPDE8A1
S00 + cAMP $\xrightarrow{r_3}$ S10	
	PDE8A1 **dephosphorylation:**
Ser259 **phosphorylation:**	PP + S001 $\xrightarrow{r_{14}}$ PP + S000
S100 $\xrightarrow{r_4}$ S110	PP + S011 $\xrightarrow{r_{15}}$ PP + S010
S101 $\xrightarrow{r_5}$ S111	uPP + pPDE8A1 $\xrightarrow{r_{16}}$ PP + PDE8A1
S10 $\xrightarrow{r_6}$ S11	
	cAMP **degradation:**
Ser259 **dephosphorylation:**	S011 + cAMP $\xrightarrow{r_{19}}$ S011
PP + S010 $\xrightarrow{r_7}$ PP + S000	S001 + cAMP $\xrightarrow{r_{20}}$ S001
PP + S011 $\xrightarrow{r_8}$ PP + S001	S100 + cAMP $\xrightarrow{r_{21}}$ S100
PP + S01 $\xrightarrow{r_9}$ PP + S00	S110 + cAMP $\xrightarrow{r_{22}}$ S110
	S010 + cAMP $\xrightarrow{r_{23}}$ S010
cAMP **release:**	S000 + cAMP $\xrightarrow{r_{24}}$ S000
S111 $\xrightarrow{r_{17}}$ S011 + cAMP	pPDE8A1 + cAMP $\xrightarrow{r_{25}}$ pPDE8A1
S11 $\xrightarrow{r_{18}}$ S01 + cAMP	PDE8A1 + cAMP $\xrightarrow{r_{26}}$ PDE8A1

Fig. 5. Biochemical reactions occurring during scaffold-mediated crosstalk between cAMP and the Raf-1/MEK/ERK pathway. The notation $\mathsf{Sv_1v_2v_3}$ represents a filled scaffold with $\mathsf{v_1}$, $\mathsf{v_2}$, $\mathsf{v_3}$ denoting the activation state of the bound PKA, site Ser259 of Raf-1 and PDE8A1 respectively, i.e., 0 for inactive and 1 for active or phosphorylated. Similarly, $\mathsf{Su_1u_2}$ represents an unfilled scaffold with $\mathsf{u_1}$ and $\mathsf{u_2}$ denoting the activation state of the bound PKA and Ser259 respectively.

4.3 The PRISM Model for the AKAP System

The PRISM model consists of four modules: one module for cAMP, one module for the scaffold with 12 variables (one variable for each type of scaffold), a module for PDE8A1 and pPDE8A1, and a module for PP and uPP. The complete PRISM model can be found at `http://www.dcs.gla.ac.uk/~muffy/akap/`.

We assume that the initial concentrations for species S00, S000, PP and uPP are all equal to $12\,mol/l$, for cAMP $120\,mol/l$, for unscaffolded PDE8A1 $6\,mol/l$, and 0 otherwise. We calculate the stepsize for the CTMC with levels abstraction as $h = \frac{12}{N}$ with N the number of levels. The system is not closed as cAMP is added exogenously *from time to time*. Such interaction is needed in the model because cAMP is consumed and to avoid termination, must be replenished. We model this interaction with an extra integer variable `tick` ranging from 0 to maximum value `tick_max` (10 in our prototype). The concentration level of cAMP increases when the value of `tick` is less than `tick_max`/2 or it reaches the maximum value

(with `tick` being reset to 0). The variable `tick` is incremented by 1 whether or not diffusion takes place, i.e. when its value is greater than `tick_max`/2 but less than the maximum. We consider a default 1.0 rate for all reactions including r_4 and r_{19}, unless defined as equal or proportional to r_4 and r_{19}.

An indication of the size of the model is: for $N = 3$, we have 1 632 240 states and 12 691 360 transitions, and for $N = 5$, we have 74 612 328 states and 734 259 344 transitions.

5 Trend-based Characterisation of Transitions in an MPP

In this section we define trend formulae that describe stochastic trends of colonies, illustrate them with several examples, show how they can be encoded in the PRISM model checker, and compare them to the trend variables introduced by Ballarini and Guerriero [13].

5.1 Trend Formulae

In a similar approach taken to the definition of the transition probability function, we introduce families of functions $\mathbf{P}_{i\uparrow}$, $\mathbf{P}_{i\downarrow}$, $\mathbf{P}_{i=}$ corresponding to increasing, decreasing or constant counts of individuals in a colony i respectively, where i ranges over colony identifiers in an MPP.

Definition 5.1. *Let $\mathcal{M} = (S, s_0, R, AP, L)$ be an MPP. The probability of making a transition from a state s to a state where the count of individuals in colony i* increases *is a function $\mathbf{P}_{i\uparrow} : S \to [0,1]$ defined as the sum of all i-increasing transition rates divided by the exit rate in state s:*

$$\mathbf{P}_{i\uparrow}(s) = \begin{cases} \frac{1}{E(s)} \cdot \sum\{R(s,s') \mid s' \in S, s_i < s'_i\}, \text{ if } E(s) \neq 0 \\ 0, \qquad\qquad\qquad\qquad\qquad\quad \text{otherwise} \end{cases}$$

The functions $\mathbf{P}_{i\downarrow} : S \to [0,1]$ and $\mathbf{P}_{i=} : S \to [0,1]$ of making a transition from a state s to a state where the count of individuals in colony i decreases *or* stays constant *are defined in a similar way:*

$$\mathbf{P}_{i\downarrow}(s) = \begin{cases} \frac{1}{E(s)} \cdot \sum\{R(s,s') \mid s' \in S, s_i > s'_i\}, \text{ if } E(s) \neq 0 \\ 0, \qquad\qquad\qquad\qquad\qquad\quad \text{otherwise} \end{cases}$$

$$\mathbf{P}_{i=}(s) = \begin{cases} \frac{1}{E(s)} \cdot \sum\{R(s,s') \mid s' \in S, s_i = s'_i\}, \text{ if } E(s) \neq 0 \\ 0, \qquad\qquad\qquad\qquad\qquad\quad \text{otherwise} \end{cases}$$

As expected, we have $\mathbf{P}_{i\uparrow}(s)+\mathbf{P}_{i\downarrow}(s)+\mathbf{P}_{i=}(s) = 1$ for all $s \in S$ with $E(s) \neq 0$.

Definition 5.2 (Trend formulae). *A* trend formula *θ is a boolean predicate over $\mathbf{P}_{i\uparrow}(s)$, $\mathbf{P}_{i\downarrow}(s)$ and $\mathbf{P}_{i=}(s)$, where $s \in S$, of one of the following forms:*

$$\theta ::= f(s) = p \mid f(s) > p \mid f(s) = f'(s) \mid f(s) > f'(s) \mid \neg\theta \mid \theta \wedge \theta$$
$$\forall f, f' \in \{\mathbf{P}_{i\uparrow}, \mathbf{P}_{i\downarrow}, \mathbf{P}_{i=}\}, \forall s \in S, \forall p \in [0,1]$$

Using the above elementary trend formulae, we define a derived set of trend formulae consisting of inequalities such as $\mathbf{P}_{i\uparrow} \leq p$ or $\mathbf{P}_{i\uparrow} \geq \mathbf{P}_{i\downarrow}$ and the following auxiliary named trends.

Definition 5.3 (Auxiliary trend formulae). *We say that in a state s* the stochastic trend *of a colony i is:*

- strictly increasing *if*

$$i{\Uparrow}(s) \triangleq (\mathbf{P}_{i\uparrow}(s) > \mathbf{P}_{i\downarrow}(s)) \wedge (\mathbf{P}_{i\uparrow}(s) > \mathbf{P}_{i=}(s)) \text{ is true}$$

- strictly decreasing *if*

$$i{\Downarrow}(s) \triangleq (\mathbf{P}_{i\downarrow}(s) > \mathbf{P}_{i\uparrow}(s)) \wedge (\mathbf{P}_{i\downarrow}(s) > \mathbf{P}_{i=}(s)) \text{ is true}$$

- weakly increasing *if* $i{\uparrow}(s) \triangleq \mathbf{P}_{i\uparrow}(s) > \mathbf{P}_{i\downarrow}(s)$ *is true*
- weakly decreasing *if* $i{\downarrow}(s) \triangleq \mathbf{P}_{i\downarrow}(s) > \mathbf{P}_{i\uparrow}(s)$ *is true*
- very weakly increasing *if* $i{\uparrow^{=}}(s) \triangleq \mathbf{P}_{i\uparrow}(s) \geq \mathbf{P}_{i\downarrow}(s)$ *is true*
- very weakly decreasing *if* $i{\downarrow_{=}}(s) \triangleq \mathbf{P}_{i\downarrow}(s) \geq \mathbf{P}_{i\uparrow}(s)$ *is true*
- constant *if* $i^{=}(s) \triangleq (\mathbf{P}_{i=}(s) > \mathbf{P}_{i\downarrow}(s)) \wedge (\mathbf{P}_{i=}(s) > \mathbf{P}_{i\uparrow}(s))$ *is true*
- equi *if* $i{\updownarrow^{=}}(s) \triangleq (\mathbf{P}_{i\uparrow}(s) = \mathbf{P}_{i\downarrow}(s)) \wedge (\mathbf{P}_{i\downarrow}(s) = \mathbf{P}_{i=}(s))$ *is true*

We illustrate the use of trend formulae in the next section.

5.2 Trend-based Properties in CSL

We use trend formulae in CSL formulae for reasoning over changes in particular colony counts. Therefore, we extend the set of state formulae in CSL to include trend formulae as modalities of arity 0. The definition of path formulae does not change.

$$\begin{aligned} &\textit{State formula } \Phi ::= \textit{true} \mid a \mid \Phi \wedge \Phi \mid \theta \mid \mathcal{P}_{\bowtie p}[\Psi] \mid \mathcal{S}_{\bowtie p}[\Psi] \\ &\textit{Path formula } \Psi ::= \mathbf{X}\,\Phi \mid \Phi\,\mathbf{U}^{\mathrm{I}}\,\Phi \end{aligned}$$

The semantics of trend formulae is defined as $s \models \theta$ if and only if $\theta(s) \equiv \textit{true}$.

In the following we illustrate several CSL properties using trend formulae on two examples.

Example 5.1. Consider MPP C_1 defined in Fig. 6 with one colony whose count ranges from 1 to 5 individuals and states from s_0 to s_6. The initial state is $s_0 = 4$. We encode the MPP using two variables: i for the colony index and k for the state index. Then for instance, the evaluation of the transition probability functions $\mathbf{P}_{i\uparrow}$, $\mathbf{P}_{i\downarrow}$ and $\mathbf{P}_{i=}$ in state s_0 gives 0, 1 and 0 respectively (hence $i{\Uparrow}(s_0)$ is true), while in state s_1, functions $\mathbf{P}_{i\uparrow}$, $\mathbf{P}_{i\downarrow}$ and $\mathbf{P}_{i=}$ evaluate to $\frac{1}{2}$, $\frac{1}{2}$ and 0 respectively. In state s_2 the trend of i is strictly increasing since $\mathbf{P}_{i\uparrow}(s_2) = 1$. In state s_4 the probability of a decreasing count is $\frac{2}{3}$ and of a constant count is $\frac{1}{3}$.

In the following CSL *experiments* we use the variable s to range over states (indexed by k), with k ranging from 0 to 6:

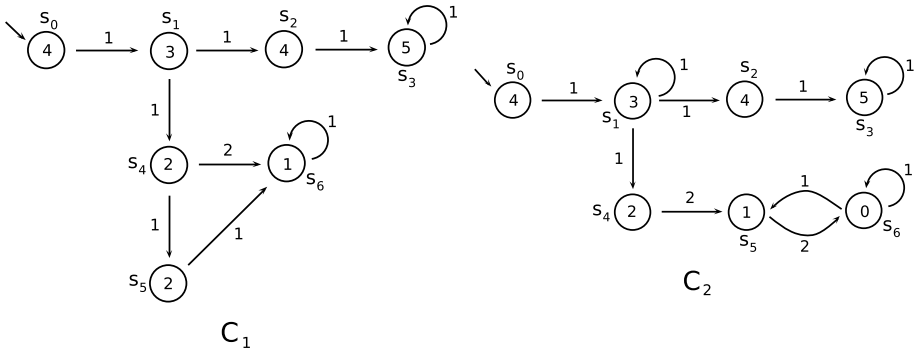

Fig. 6. Markov population process C_1 and C_2 both having initial state s_0.

- The probabilities of reaching a state s_k where the trend θ is true, with θ ranging over $i{\Uparrow}$, $i{\uparrow}$, $i{\uparrow^{=}}$, $i{\Downarrow}$, $i{\downarrow}$, $i{\downarrow_{=}}$, $i^{=}$ and $i{\updownarrow^{=}}$, are listed in Table 1.
- Eventually all states are stochastically very weakly increasing:
  $C_1 \models \mathcal{P}_{\geq 1}[\mathbf{F}\,(\mathcal{P}_{>0}[\mathbf{G}\,(i{\uparrow^{=}})])]$ returns *true*.
- Not all states are eventually weakly increasing:
  $C_1 \models \mathcal{P}_{\leq 0}[\mathbf{F}\,(\mathcal{P}_{>0}[\mathbf{G}\,(i{\uparrow})])]$ returns *true*.
- Eventually all states are stochastically very weakly decreasing:
  $C_1 \models \mathcal{P}_{\geq 1}[\mathbf{F}\,(\mathcal{P}_{>0}[\mathbf{G}\,(i{\downarrow_{=}})])]$ returns *true*.
- The probability that all states are very weakly decreasing:
  $C_1 \models \mathcal{P}_{=?}[\mathbf{G}\,(i{\downarrow_{=}})]$ returns 0.5.

Table 1. Model checking CSL formulae for C_1 that compute the probability of reaching a state s_k where the trend θ is true, with θ ranging over trend formulae and k ranging from 0 to 6

Model checking CSL formula	k						
	0	1	2	3	4	5	6
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i{\Uparrow}) \wedge (s = k))]$	0	0	0.5	0	0	0	0
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i{\uparrow}) \wedge (s = k))]$	0	0	0.5	0	0	0	0
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i{\uparrow^{=}}) \wedge (s = k))]$	0	1	0.5	0.5	0	0	0.5
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i{\Downarrow}) \wedge (s = k))]$	1	0	0	0	0.5	0.167	0
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i{\downarrow}) \wedge (s = k))]$	1	0	0	0	0.5	0.167	0
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i{\downarrow_{=}}) \wedge (s = k))]$	1	1	0	0.5	0.5	0.167	0.5
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i^{=}) \wedge (s = k))]$	0	0	0	0.5	0	0	0.5
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i{\updownarrow^{=}}) \wedge (s = k))]$	0	0	0	0	0	0	0

- Eventually i will strictly decrease for some time with a non-zero probability until it reaches a constant trend:
  $C_1 \models \mathcal{P}_{\geq 1}[\mathbf{F}\,(\mathcal{P}_{>0}[(i{\Downarrow})\,\mathbf{U}\,(i^{=})])]$ returns *true*.
- Eventually i will strictly increase for some time with a non-zero probability until it reaches a constant trend:
  $C_1 \models \mathcal{P}_{\geq 1}[\mathbf{F}\,(\mathcal{P}_{>0}[(i{\Uparrow})\,\mathbf{U}\,(i^{=})])]$ returns *true*.

Example 5.2. Consider now the MPP C_2 defined in Fig. 6. Note that C_2 has infinite paths including infinite loop-free paths (i.e., infinite paths without self-loops). We analyse a set of CSL queries using trends that are more complex than those from Example 5.1. Again, k ranges from 0 to 6.

- Eventually a state with an equi trend is reached (more precisely s_1):
  $C_2 \models \mathcal{P}_{\geq 1}[\mathbf{F}\,(i{\updownarrow}^{=})]$ returns *true*, whereas in C_1 this query returns *false*.
- The probability of reaching a state having a very weakly increasing trend and in the next state the trend is strictly decreasing with non-zero probability:
  $C_2 \models \mathcal{P}_{=?}[\mathbf{F}\,((i{\uparrow}^{=}) \wedge \mathcal{P}_{>0}[\mathbf{X}\,((i{\Downarrow}) \wedge (s = k))])]$ returns 1 for $k = 4$, 0.5 for $k = 5$, and 0 otherwise.
  If we restrict the probability of a strictly decreasing next state to at least 0.5, then the probability of $C_2 \models \mathcal{P}_{=?}[\mathbf{F}\,((i{\uparrow}^{=}) \wedge \mathcal{P}_{>0.5}[\mathbf{X}\,((i{\Downarrow}) \wedge (s = k))])]$ is 0.5 for $k = 5$, and 0 otherwise.
- The probability that eventually all states are stochastically very weakly decreasing: $C_2 \models \mathcal{P}_{=?}[\mathbf{F}\,\mathcal{P}_{>0}[\mathbf{G}\,(i{\downarrow}_{=})]]$ returns 0.5, whereas in C_1 the same query returns 1.
- The probability that eventually i will strictly decrease for some finite time with a non-zero probability until it reaches a constant trend:
  $C_2 \models \mathcal{P}_{=?}[\mathbf{F}\,\mathcal{P}_{>0}[(i{\Uparrow})\,\mathbf{U}\,(i^{=})]]$ returns 0.5.
- Eventually i will have a strictly decreasing trend for some time until reaching a constant trend and then, with probability greater than 0.5, will show an increasing trend:
  $C_2 \models \mathcal{P}_{\geq 1}[\mathbf{F}\,\mathcal{P}_{>0}[(i{\Downarrow})\,\mathbf{U}\,\mathcal{P}_{>0}[(i^{=})\,\mathbf{U}\,\mathcal{P}_{>0.5}[i{\Uparrow}]]]]$ returns *true*.
- The probability that always a decreasing trend of i eventually leads to an increasing trend and vice versa:
  $C_2 \models \mathcal{P}_{=?}[\mathbf{G}\,(((i{\Downarrow}) \Longrightarrow \mathcal{P}_{>0}[\mathbf{F}\,i{\Uparrow}]) \wedge ((i{\Uparrow}) \Longrightarrow \mathcal{P}_{>0}[\mathbf{F}\,i{\Downarrow}]))]$ returns 0.5.
- The long-run probability that a decreasing trend of i eventually leads to an increasing trend and vice versa:
  $C_2 \models \mathcal{S}_{\geq 1}[((i{\Downarrow}) \Longrightarrow \mathcal{P}_{>0}[\mathbf{F}\,i{\Uparrow}]) \wedge ((i{\Uparrow}) \Longrightarrow \mathcal{P}_{>0}[\mathbf{F}\,i{\Downarrow}])]$ returns *true*.
- Variable i has a constant trend in the long-run, more specifically when state s_3 is reached: $C_2 \models \mathcal{S}_{=?}[i^{=}]$ returns 0.5.

Since we can define i-increasing/decreasing/constant functions for DTMCs, the trend formulae approach presented in this section is also applicable to DTMC models and PCTL formulae.

5.3 Trend Formulae *vs.* Trend Variables

An approach closely related to trend formulae is described in [13] in the context of modelling and analysis of biochemical systems. It is based on associating two boolean variables inc_X and dec_X to each species X in order to record, for each possible transition, if the value of X increases or decreases respectively; if the variable X is not updated by a transition, neither are the associated variables inc_X and dec_X. The aim is to analyse behavioural queries such as monotonic and oscillatory trends in models of biochemical systems. In our preliminary work [28] we had a similar approach based on adding one integer variable drv_X to each species X; the value of drv_X is updated at the the same time as the value of X and it denotes the sign of the change of X: 1 for increasing, -1 for decreasing and 0 otherwise. In the following we identify two major differences between the trend variable approach and our trend formulae approach.

State Space Size. Trend formulae do not increase the size of the state space, a well-known issue for the trend variable approach. To support this claim let us first give a constructive definition of a single-colony MPP enriched with trend variables. Let $M = (S, s_0, R, L)$ be a single-colony MPP; then the corresponding MPP with trend variables $M' = (S', s'_0, R', L')$ is constructed as follows:

1. Add the initial state $s'_0 = (s_0, \mathbf{t}, \mathbf{t})$ to S'.
2. For all states $i, j \in S$ with $R(i, j) > 0$ and $(i, inc, dec) \in S'$ with $inc, dec \in \{\mathbf{t}, \mathbf{f}\}$, add (j, inc', dec') to S' where:

$$inc' = \begin{cases} \mathbf{t}, & \text{if } i < j \\ \mathbf{f}, & \text{if } i > j \\ inc, & \text{if } i = j \end{cases} \qquad dec' = \begin{cases} \mathbf{f}, & \text{if } i < j \\ \mathbf{t}, & \text{if } i > j \\ dec, & \text{if } i = j \end{cases}$$

3. $R((i, inc, dec), (j, inc', dec')) = R(i, j)$ for all $(i, inc, dec), (j, inc', dec') \in S'$.
4. $L'(i, inc, dec) = L(i)$ for all $(i, inc, dec) \in S'$.

Proposition 5.1. *Given a single-colony MPP M, the state size of the MPP M' obtained from M by enriching it with trend variables is greater or at least equal to the state size of M.*

Proof. We prove that $|S'| \geq |S'|$ by identifying two types of transitions in M that increase the state space:

- If $R(i, j) > 0$ for $i > j$ and $(i, inc, dec) \in S'$, then $(j, \mathbf{f}, \mathbf{t}) \in S'$. If $R(k, j) > 0$ for $k < j$ and $(k, inc', dec') \in S'$, then $(j, \mathbf{t}, \mathbf{f}) \in S'$. In this case M' has two distinct states for the same colony count of j, one more than M has.
- If $R(i, j) > 0$ and $R(j, i) > 0$ with $i > j$ and $(i, inc, dec), (j, inc', dec') \in S'$, then $(j, \mathbf{f}, \mathbf{t}), (i, \mathbf{t}, \mathbf{f}) \in S'$. If $(inc, dec) \neq (\mathbf{t}, \mathbf{f})$ then M' has two distinct states for the same colony count of i, one more than M has; the same reasoning goes for the state j in M.

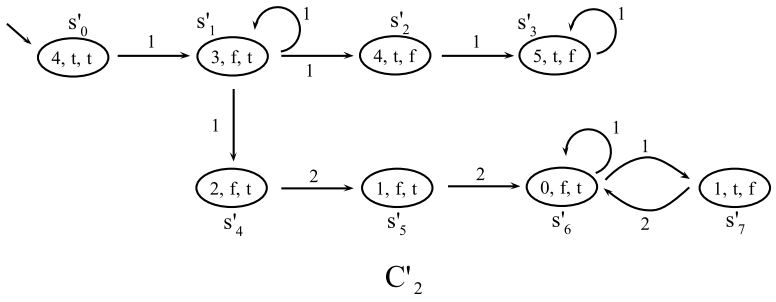

Fig. 7. Markov population process C_2' with initial state $s_0' = (4, \mathbf{t}, \mathbf{t})$ obtained from C_2 by enriching the states with two boolean variables keeping track of the increasing or decreasing trend of the first component of the state.

Consider the following simple example. We add trend variables to the MPP C_2 defined in Fig. 6 to obtain the MPP C_2' depicted in Fig. 7. Notice that C_2' has one additional state and one additional transition, due to the cycle between states s_5 and s_6 in C_2: if we consider a path in C_2' starting from the initial state $(4, \mathbf{t}, \mathbf{t})$, when we first reach the state where $i = 1$ the trend variables *inc* and *dec* are set to **f** and **t** respectively because the value of i decreases strictly; but when the state $i = 1$ (i.e., state s_7') is the state visited from $i = 0$ (i.e., state s_6'), then *inc* and *dec* are set to **t** and **f** respectively since the value of i increases from 0 to 1. Hence in C_2' there are two states with $i = 1$ but different values for the trend variables *inc* and *dec*.

The result above can be generalised for MPPs with several colonies. Therefore if a state occurs multiple times along an execution path or along different paths, the size of the state space may increase. In addition, the size of each state increases by the two boolean trend variables, for each colony in the MPP.

A Priori and A Posteriori Trend Computation. Trend variables provide an *a posteriori* detection of a stochastic trend, whereas trend formulae an *a priori* detection. More precisely, if a transition from state s to state s' increases (decreases) the counts in a colony, the trend variable approach detects in state s' the increasing (resp. decreasing) trend, whereas the trend formulae approach detects the trend in state s, i.e., *prior to the transition*. When deciding to analyse an MPP using trends, one has to decide which type of detection of the trend best suits the problem. Note also that the values of trend variables associated with a colony variable A are not updated during a transition if the value of A is not changed by the transition. This notion of monotonicity corresponds, in our framework, to weak monotonicity, more precisely to *very weakly increasing/decreasing* trends.

We illustrate the difference between the increasing trend computed using trend variables and computed using the trend formulae for the MPP C_1 from Fig. 6. Let C_1' be the MPP resulting from adding trend variables to C_1, as depicted in Fig. 8. Now consider the temporal property $\phi =$ *"eventually the value*

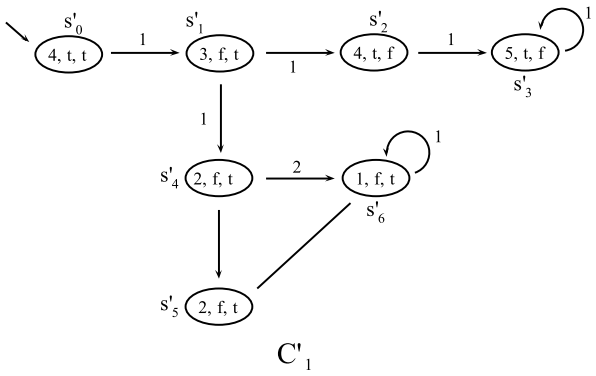

Fig. 8. MPP C_1' with initial state $s_0' = (4, \mathbf{t}, \mathbf{t})$ obtained from C_1 by adding trend variables.

of i will increase". In CSL with trends formulae this property is specified as $\mathcal{P}_{=?}[\mathbf{F}\,(\theta_{up} \wedge (s = k))]$ for MPP C_1 with θ_{up} either strictly increasing, weakly increasing or very weakly increasing trend and k ranging from 0 to 6. Then $\phi' = \mathcal{P}_{=?}[\mathbf{F}\,(inc \wedge (s' = k))]$ is the property corresponding to ϕ we want to check for C_1'. The results of model checking ϕ and ϕ' in PRISM are given in Table 2. Notice that in C_1' an increasing trend is found in the initial state only because the variables *inc* and *dec* are set initially to *true*. Otherwise, an increasing trend is detected in C_1' in state s_2' because the transition from s_1' to s_2' increased the value of i: the conclusion that the trend is increasing in state s_2' was established when the transition to be triggered was already chosen. Therefore we call this analysis *a posteriori*. The same reasoning can be applied to the increasing trend in state s_3'. The detection of state s_2 as a state with increasing trend in C_1 is performed *a priori* any possible transition and this trend corresponds to the increasing trend detected in state s_3' of C_1'. With trend formulae we detect the states with the highest probability of moving to a state where the value of i is increased. The corresponding state of the increasing trend of s_2' in C_1' is s_1 in C_1 having a very weakly increasing trend. The strictly increasing and weakly increasing trends require that the probability of i to increase in state s_1 is strictly greater than the probability to decrease, which is not the case because $\mathbf{P}_{i\uparrow}(s_1) = \mathbf{P}_{i\downarrow}(s_1) = 0.5$. The very weakly increasing trend is detected in states s_3 and s_6 where, because of the cycle, the value of i remains unchanged.

The trend formulae approach permits expressing different concepts that cannot be expressed with trend variables such as several types of monotonicity or, for instance, the following property for C_1: *"What is the probability to eventually reach a state where i will most probably increase and in the next state will most probably decrease"*. By model checking $C_1 \models \mathbf{P}_{=?}[\mathbf{F}\,((i\uparrow^{=}) \wedge \mathcal{P}_{>0}[\mathbf{X}\,(i\Downarrow)])]$ we obtain probability 1 as the probability of taking the path starting in the initial state that reaches the state s_1 where $i = 3$ has the highest chances of increasing and in the next state to decrease strictly. But if we consider the CSL formula

Table 2. Comparison of checking trend formulae and trend variable properties in C_1 and C_1'

Model checking CSL formula	k						
	0	1	2	3	4	5	6
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i\Uparrow) \wedge (s = k))]$	0	0	0.5	0	0	0	0
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i\uparrow) \wedge (s = k))]$	0	0	0.5	0	0	0	0
$C_1 \models \mathcal{P}_{=?}[\mathbf{F}\,((i\uparrow^{=}) \wedge (s = k))]$	0	1	0.5	0.5	0	0	0.5
$C_1' \models \mathcal{P}_{=?}[\mathbf{F}\,(inc \wedge (s' = k))]$	1	0	0.5	0.5	0	0	0

with trend variables, $C_1' \models \mathbf{P}_{=?}[\mathbf{F}\,(inc \wedge \mathcal{P}_{>0}[\mathbf{X}\,dec])]$, the result is probability 0 because it detects the increasing trend in state s_2', where $i = 4$ and from the next state the trend is only increasing.

6 Trend-based Analysis of the AKAP System

In this section we apply trend formulae in the analysis of AKAP system. We formalise in CSL the causality and fluctuation properties and model check them in PRISM. Stochastic trend formulae are essential for expressing these properties. The key question is which trends best encode *more* X and *less* X for X a colony. Consider the statement *more* X. In order to express an increase in the concentration of X, we rule out decreasing concentrations but consider transitions that do not change the concentration. Therefore the trend we choose to encode *more* X is the weakly increasing trend $X\uparrow$. The same reasoning applies to encoding *less* X by $X\downarrow$.

6.1 Causality Relation

A causality relation between two events can be formalised as a temporal query using the *necessarily preceded* or *requirement* pattern [29]. This pattern represents an ordering relation between two events, the occurrence of the latter being conditioned by the occurrence of the former: *a state ϕ is reachable and is necessarily preceded all the time by a state φ*. The associated CTL formula is $\mathbf{EF}\phi \wedge (\mathbf{AG}((\neg\varphi) \Rightarrow \mathbf{AG}(\neg\phi)))$, where $\mathbf{A}$ and $\mathbf{E}$ are temporal operators quantifying universally and existentially over paths respectively.

Consider now the causal relation stated in Section 3 for the AKAP model:

$$\textit{more}\ \mathsf{pPDE8A1} \Rightarrow \textit{less}\ \mathsf{cAMP} \text{ and } \textit{less}\ \textit{active}\ \mathsf{PKA}$$

The two CSL state formulae encoding the two sides of the implication above are:

$$\begin{aligned} \varphi_1 &\triangleq \mathsf{pPDE8A1}\uparrow \\ \phi_1 &\triangleq (\mathsf{cAMP}\downarrow) \wedge (\textit{active}\ \mathsf{PKA}\downarrow) \end{aligned}$$

where the concentration of *active* PKA is given by the sum of concentrations of all scaffold combinations with 1 in the first position: S10, S11, S100, S110, S101 and S111. Employing basic proposition equivalences, we translate the requirement pattern for the cause φ_1 and effect ϕ_1 into CSL to obtain the following formula which was checked as *true* for our PRISM model:

$$\mathcal{P}_{>0}[\mathbf{F}\phi_1] \wedge \mathcal{P}_{\geq 1}[\mathbf{G}((\neg\varphi_1) \Rightarrow \mathcal{P}_{\leq 0}[\mathbf{G}\phi_1]))]$$

We can express a tighter causality relation between increasing concentration levels of pPDE8A1 ($\varphi_2 \triangleq$ pPDE8A1 $\uparrow$) and decreasing levels of cAMP ($\phi_2 \triangleq$ cAMP$\downarrow$) using the following formula checked as *true* for our PRISM model:

$$\mathcal{P}_{\geq 1}[\mathbf{F}\,((\neg\varphi_2 \wedge \neg\phi_2)\,\mathbf{U}\,(\mathcal{P}_{\geq 1}[(\varphi_2 \wedge \neg\phi_2)\,\mathbf{U}\,\mathcal{P}_{>0}[\mathbf{X}\,\phi_2]]))]$$

This formula stands for "*more* pPDE8A1 $\Rightarrow$ *less* cAMP" in the notation introduced in Sect. 3 and it states that there is a time interval where the trend of pPDE8A1 is not decreasing and the trend of cAMP is not decreasing until the trend of pPDE8A1 starts increasing and soon after the trend of cAMP starts decreasing. A similar CSL formula can be employed in order to show that "*less* pPDE8A1 $\Rightarrow$ *more* cAMP" and "*less* cAMP $\Rightarrow$ *less active* PKA".

6.2 Pulsating Behaviour

An oscillating behaviour of a variable assumes a fluctuation of the value of the variable around a given value k. Oscillation and its expression as temporal formulae in CTL and PCTL have been studied in [10] and informally described as *always in the future, the variable x departs from and reaches the values k infinitely often*. The corresponding CTL formula is $\mathbf{AG}(((x = k) \Rightarrow \mathbf{EF}(x \neq k)) \wedge ((x \neq k) \Rightarrow \mathbf{EF}(x = k)))$. In the context of BIOCHAM [9], a weaker form of oscillation properties expressed in CTL is used with the symbolic model checker NuSMV; the oscillating behaviour is approximated by the necessary but not sufficient formula $\mathbf{EG}((\mathbf{EF}\,\neg\varphi) \wedge (\mathbf{EF}\,\varphi))$ expressing that there exists a path where at all time points whenever φ is true it becomes eventually false, and whenever it is false it becomes eventually true.

We are interested in *pulsating* behaviour, i.e. no fixed k. We therefore consider oscillations (around 0) of the values of some variables. We refer to this approximate oscillating behaviour as *pulsation*. The CSL formulae describing pulsations of cAMP, *active* PKA and pPDE8A1 are the following:

$$\mathcal{P}_{\geq 1}[\mathbf{G}(((\text{cAMP}\uparrow) \Rightarrow \mathcal{P}_{>0}[\mathbf{F}(\text{cAMP}\downarrow)]) \wedge ((\text{cAMP}\downarrow) \Rightarrow \mathcal{P}_{>0}[\mathbf{F}(\text{cAMP}\uparrow)]))]$$

$$\mathcal{P}_{\geq 1}[\mathbf{G}(((\textit{active PKA}\uparrow) \Rightarrow \mathcal{P}_{>0}[\mathbf{F}(\textit{active PKA}\downarrow)]) \wedge ((\textit{active PKA}\uparrow) \Rightarrow \mathcal{P}_{>0}[\mathbf{F}(\textit{active PKA}\downarrow)]))]$$

$$\mathcal{P}_{\geq 1}[\mathbf{G}(((\text{pPDE8A1}\uparrow) \Rightarrow \mathcal{P}_{>0}[\mathbf{F}(\text{pPDE8A1}\downarrow)]) \wedge ((\text{pPDE8A1}\downarrow) \Rightarrow \mathcal{P}_{>0}[\mathbf{F}(\text{pPDE8A1}\uparrow)]))]$$

and they were all checked as *true* for our model using PRISM.

We can also prove that the presence of a *synchronised* pulsation with pPDE8A1 showing a very weakly increasing (decreasing) trend at the same time as cAMP

and *active* PKA follow a very weakly decreasing (increasing) trend. Consider the following two state formulae:

$$\phi_3 \triangleq (\mathsf{pPDE8A1}{\uparrow^{=}}) \wedge (\mathsf{cAMP}{\downarrow_{=}}) \wedge (\mathit{active}\ \mathsf{PKA}{\downarrow_{=}})$$
$$\phi_4 \triangleq (\mathsf{pPDE8A1}{\downarrow_{=}}) \wedge (\mathsf{cAMP}{\uparrow^{=}}) \wedge (\mathit{active}\ \mathsf{PKA}{\uparrow^{=}})$$

The following formula expressing a synchronised pulsation was checked as *true* for our model using PRISM:

$$\mathcal{P}_{\geq 1}[\mathbf{G}((\phi_3 \Rightarrow \mathcal{P}_{>0}[\mathbf{F}\,\phi_4]) \wedge (\phi_4 \Rightarrow \mathcal{P}_{>0}[\mathbf{F}\,\phi_3]))]$$

We remark that using *weakly monotonic trends* in formulae ϕ_3 and ϕ_4, the above formula would return false. Hence weakly monotonic trends are too strong to show the synchronised pulsation, whereas the *very weakly trends* validate it. The reason is that the pulsations take place modulo a very small time delay, when the probability of increasing concentrations may be equal to the probability of decreasing concentration of a species. Therefore the three species (cAMP, *active* PKA and pPDE8A1) do pulsate in a synchronised way, but only when we consider weak monotonicity.

Finally, we note that we have not used any timed operators, i.e. the bounded until operator, in this case study. This is because the system is still under investigation and currently we have only semi-quantitative information. It was therefore more relevant to consider trends within the context of unbounded temporal operators. In other applications, where rate information is more precise, time-bounded operators would be more relevant.

7 Conclusions and Future Work

We have introduced stochastic trend formulae for characterising the probability of increasing/decreasing colonies in MPP models. The probabilities are forward-looking, based on behaviour that will occur in the future. We defined a set of stochastic trend formulae and showed how to derive several formulae encapsulating useful forms of monotonicity. We extended the set of state formulae of CSL with trends formulae and we defined an encoding in the PRISM language using the PRISM formula construct, which means that there are no additional variables in the underlying state space. We compared stochastic trend formulae with stochastic trend variables, and showed the former is more tractable with respect to the state space size and the size of the states. We note that while we focus on continuous time here, similar results are easily obtained for the discrete time case.

After illustration with several small examples, stochastic trends were applied to the analysis of causality relations and pulsating behaviour in a significant biochemical signalling case study: the AKAP mediated crosstalk between the cAMP and Raf-1/ERK/MEK pathways. We believe this to be the first formal model of this system. We were able, with the use of trend formulae, to show causality and pulsations predicted by life scientists and observed in laboratory experiments.

Future work includes investigating how stochastic trends (an abstraction) over different combinations of colonies affects various relations (e.g. simulation) between MPPs.

Acknowledgements We thank George Baillie, Kim Brown and Walter Kolch from the Faculty of Biomedical & Life Science, University of Glasgow, for discussions, guidance and insight into the AKAP scaffold. We also thank the anonymous reviewers of this paper for their insightful comments on the work. This work was supported by the SIGNAL project, funded by the UK Engineering and Physical Science Research Council (EPSRC) under grant number EP/E031439/1.

References

1. Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. J. Phys. Chem. **81**(25) (1977) 2340–2361
2. Ciocchetta, F., Degasperi, A., Hillston, J., Calder, M.: Some Investigations Concerning the CTMC and the ODE Model Derived From Bio-PEPA. Electr. Notes Theor. Comput. Sci. **229**(1) (2009) 145–163
3. Bartlett, M.S.: An introduction to stochastic processes, with special reference to methods and applications. 3rd edn. Cambridge University Press (1978)
4. Kingman, J.F.C.: Markov Population Processes. Journal of Applied Probability **6** (1969) 1–18
5. Cohen, J.E.: Markov population processes as models of primate social and population dynamics. Theoretical Population Biology **3**(2) (1972) 119–134
6. Henzinger, T.A., Jobstmann, B., Wolf, V.: Formalisms for Specifying Markovian Population Models. In Bournez, O., Potapov, I., eds.: RP. Volume 5797 of Lecture Notes in Computer Science., Springer (2009) 3–23
7. Fages, F.: Temporal Logic Constraints in the Biochemical Abstract Machine BIOCHAM. In Hill, P.M., ed.: 15th International Symposium on Logic Based Program Synthesis and Transformation (LOPSTR'05). Volume 3901 of Lecture Notes in Computer Science., Springer (2005) 1–5
8. Rizk, A., Batt, G., Fages, F., Soliman, S.: On a Continuous Degree of Satisfaction of Temporal Logic Formulae with Applications to Systems Biology. In Heiner, M., Uhrmacher, A.M., eds.: 6th International Conference on Computational Methods in Systems Biology (CMSB'08). Volume 5307 of Lecture Notes in Computer Science., Springer (2008) 251–268
9. Chabrier-Rivier, N., Chiaverini, M., Danos, V., Fages, F., Schächter, V.: Modeling and querying biomolecular interaction networks. Theoretical Computer Science **325**(1) (2004) 25–44
10. Ballarini, P., Mardare, R., Mura, I.: Analysing Biochemical Oscillation through Probabilistic Model Checking. Electr. Notes Theor. Comput. Sci. **229**(1) (2009) 3–19
11. Spieler, D., Hahn, E.M., Zhang, L.: Model Checking CSL for Markov Population Models. CoRR **abs/1111.4385** (2011)
12. Júlvez, J., Kwiatkowska, M.Z., Norman, G., Parker, D.: A Systematic Approach to Evaluate Sustained Stochastic Oscillations. In Al-Mubaid, H., ed.: Proc. of the ISCA 3rd International Conference on Bioinformatics and Computational Biology (BICoB'11), ISCA (2011) 134–139

13. Ballarini, P., Guerriero, M.L.: Query-based verification of qualitative trends and oscillations in biochemical systems. Theoretical Computer Science **411**(20) (2010) 2019–2036
14. Andrei, O., Calder, M.: A Model and Analysis of the AKAP Scaffold. Electr. Notes Theor. Comput. Sci. **268** (2010) 3–15
15. Baier, C., Haverkort, B.R., Hermanns, H., Katoen, J.P.: Model-Checking Algorithms for Continuous-Time Markov Chains. IEEE Trans. Software Eng. **29**(6) (2003) 524–541
16. Baier, C., Katoen, J.P.: Principles of Model Checking. The MIT Press (2008)
17. Kwiatkowska, M.Z., Norman, G., Parker, D.: Stochastic Model Checking. In Bernardo, M., Hillston, J., eds.: SFM. Volume 4486 of Lecture Notes in Computer Science., Springer (2007) 220–270
18. Kleinrock, L.: Queueing Systems, Volume I: Theory. John Wiley, New York (1975)
19. Bolch, G., Greiner, S., de Meer, H., Trivedi, K.S.: Queueing networks and Markov chains : modeling and performance evaluation with computer science applications. 2nd edn. Wiley-Interscience (2006)
20. Ciocchetta, F., Gilmore, S., Guerriero, M.L., Hillston, J.: Integrated Simulation and Model-Checking for the Analysis of Biochemical Systems. Electr. Notes Theor. Comput. Sci. **232** (2009) 17–38
21. Calder, M., Gilmore, S., Hillston, J.: Modelling the Influence of RKIP on the ERK Signalling Pathway Using the Stochastic Process Algebra PEPA. In: T. Comp. Sys. Biology. Volume 4230 of Lecture Notes in Computer Science., Springer (2006) 1–23
22. Calder, M., Vyshemirsky, V., Gilbert, D., Orton, R.J.: Analysis of Signalling Pathways Using Continuous Time Markov Chains. In Priami, C., Plotkin, G.D., eds.: T. Comp. Sys. Biology. Volume 4220 of Lecture Notes in Computer Science., Springer (2006) 44–67
23. Kurtz, T.G.: Limit Theorems for Sequences of Jump Markov Processes Approximating Ordinary Differential Processes. Journal of Applied Probability **8**(2) (1971) 344–356
24. Alur, R., Henzinger, T.A.: Reactive Modules. Formal Methods in System Design **15**(1) (1999) 7–48
25. Kwiatkowska, M.Z., Norman, G., Parker, D.: PRISM: probabilistic model checking for performance and reliability analysis. SIGMETRICS Performance Evaluation Review **36**(4) (2009) 40–45
26. Calder, M., Hillston, J.: Process Algebra Modelling Styles for Biomolecular Processes. T. Comp. Sys. Biology **11** (2009) 1–25
27. James E. Ferrell, J.: What Do Scaffold Proteins Really Do? Sci. STKE **2000**(52) (2000) 1–3
28. Andrei, O., Calder, M.: Modelling Scaffold-mediated Crosstalk between the cAMP and the Raf-1/MEK/ERK Pathways. In: Proceedings of the PASTA'09. (2009)
29. Monteiro, P.T., Ropers, D., Mateescu, R., Freitas, A.T., de Jong, H.: Temporal logic patterns for querying dynamic models of cellular interaction networks. Bioinformatics **24**(16) (2008) 227–233