Analysis of signalling pathways using the PRISM model checker

Muffy Calder[1], Vladislav Vyshemirsky[2], David Gilbert[2], and Richard Orton[2]

[1] Department of Computing Science, University of Glasgow, Glasgow G12 8QQ, UK
http://www.dcs.gla.ac.uk
[2] Bioinformatics Research Centre, University of Glasgow, Glasgow G12 8QQ, UK
http://www.brc.dcs.gla.ac.uk

Abstract. We describe a new modelling and analysis approach for signal transduction networks in the presence of incomplete data. We illustrate the approach with an example, the RKIP inhibited ERK pathway [1]. Our models are based on high level descriptions of continuous time Markov chains: reactions are modelled as synchronous processes and concentrations are modelled by discrete, abstract quantities. The main advantage of our approach is that using a (continuous time) stochastic logic and the PRISM model checker, we can perform quantitative analysis of queries such as *if a concentration reaches a certain level, will it remain at that level thereafter?* We also perform standard simulations and compare our results with a traditional ordinary differential equation model. An interesting result is that for the example pathway, only a small number of discrete data values is required to render the simulations practically indistinguishable.

1 Introduction

Signal transduction pathways allow cells to sense an environment and make suitable responses. External signals detected by cell membrane receptors activate a sequence of reactions, allowing the cell to recognise the signal and pass it into the nucleus. The cellular response is then activated inside the nucleus. This signalling mechanism is involved in a number of important processes, such as proliferation, cell growth, movement, apoptosis, and cell communication. The pathways are usually very complicated and embedded in more complex networks, thus manual analysis is almost impossible. Some form of modelling for computer guided analysis is required.

Our aim is to develop techniques for signal transduction pathway[3] modelling and analysis, based on incomplete, or semiquantitative data. Our models are distinctive in two ways. First, we model concentrations (rather than molecules). Second, we model incomplete data. Incompleteness is an issue because contemporary methods for biochemical experiments do not, in general, permit the measurement of absolute and continuous values of concentrations. Consequently,

[3] In this paper we use the terms pathway and network synonymously.

some existing quantitative models are over constrained. We avoid this by considering discrete, abstract concentrations. Our analysis includes simulation, but extends much further to include checking for quantitative, temporal biological queries. The models are based on high level descriptions of stochastic transition systems, i.e continuous time Markov chains (CTMCs). Reactions are modelled by synchronous processes and concentrations are modelled by discrete, abstract quantities. We use a continuous stochastic logic and the probabilistic symbolic model checker PRISM [2] to express and check a variety of temporal queries for both transient behaviours and steady state behaviours. We can also perform standard simulations and so we compare our results with a traditional ordinary differential equation model. Throughout, we illustrate our approach with an example pathway: the RKIP inhibited ERK pathway [1].

The paper is organised as follows. In section 2 we describe our example pathway the RKIP inhibited ERK pathway. Our model is developed in section 3. In section 4, we discuss different types of analysis and present three types of probabilistic, temporal queries for the model pathway. We express examples of each in the continuous stochastic logic CSL [3], and check their validity. In section 5 we show how our model compares with simulations from a MATLAB® implementation of the ordinary differential equations for the example pathway. In section 6 we discuss our results and in section 7 we review related work. We conclude in section 8.

2 RKIP and the ERK pathway

The example system we consider in this paper is the RKIP inhibited ERK pathway [1].

The ERK pathway (also called Ras/Raf, or Raf-1/MEK/ERK pathway) is a ubiquitous pathway that conveys mitogenic and differentiation signals from the cell membrane to the nucleus. An important area of experimental scientific investigation is the role the kinase inhibitor protein RKIP plays in the behaviour of this pathway. Moreover, an understanding of the functioning and structure of this pathway may lead to more general results applicable to other pathways.

We consider the pathway as described in the graphical representation of Figure 1. This figure is taken from [1], where a number of nonlinear ODEs and difference equations representing the kinetic reactions are given. We take Figure 1 as our starting point, and explain informally, its meaning. Each node is labelled by the protein (or substrate, we use the two interchangeably) it denotes. For example, Raf-1*, RKIP and Raf-1*/RKIP are proteins, the last being a complex built up from the first two. It is important to note that Raf-1*/RKIP is simply a *name*, following biochemical convention; the / symbol is not an operator (in this context). A suffix -P or -PP denotes a phosyphorylated protein, for example MEK-PP and ERK-PP. Each protein has an associated concentration, denoted by $m1$, $m2$ etc. *Reactions* define how proteins are built up and broken down. Each reaction has a rate denoted by the rate constants $k1$, $k2$, etc. These are given in the rectangles, with $kn/kn+1$ denoting that kn is the forward rate and

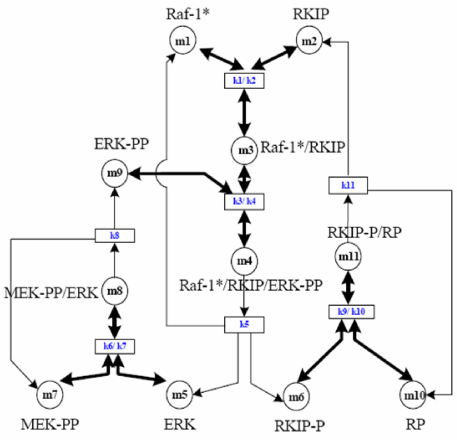

Fig. 1. RKIP inhibited ERK pathway

$kn+1$ the backward rate. So for example, Raf-1* and RKIP react (forward) with rate $k1$, and Raf-1/RKIP disassociates with rate $k2$. Initially, all concentrations are unobservable, except for m_1, m_2, m_7, m_9, and m_{10} [1].

3 Modelling signalling networks

Signalling networks describe the interaction between proteins. In this section we describe how we model the concentrations of proteins by discrete variables, and the dynamic behaviour of proteins by computational processes.

3.1 Discrete concentrations

Each protein defined in a network has a concentration which changes with time, thus $m = f(t)$, where m is a concentration of the protein and t is time. In classical approaches, ordinary differential equations are used to describe the dynamics of reactions and concentrations. But as we have indicated earlier, there is a difficulty in obtaining absolute concentration values using the methods of contemporary biochemistry. We therefore make discrete approximations to the data values and assume a set of totally ordered symbolic names representing *levels* of concentration:

$$level_0 < level_1 < \ldots < level_N$$

We additionally assume that the symbolic levels correspond to equally distributed intervals (of absolute concentrations).

3.2 Proteins as processes

We associate a concurrent, computational process with each of the proteins defined in the network and define those processes using the PRISM model checker language. This language allows the definition of systems of concurrent processes which when synchronised, denote continuous time Markov chains (CTMCs). Below, we give a very brief overview of the language, illustrating each concept with a simple example; the reader is directed to [2] for further details.

Transitions of a process are labelled with performance rates and (optional) action names. For each action, the performance rate is defined as the parameter λ of an exponential distribution of the action duration. The distribution is the "memoryless" negative exponential, that is, $P(t) = 1 - e^{-\lambda t}$ is the probability that the action will be completed before time t. A key feature is synchronisation: concurrent processes are synchronised on transitions (i.e. the transitions occur simultaneously) with common names. Transitions with distinct names are not synchronised. The performance rate for the synchronised action is the *product* of the performance rates of the synchronised processes. For example, if process A performs action α with rate λ_1, and process B performs action α with rate λ_2, then the performance rate of action α when A is synchronised with B is $\lambda_1 \cdot \lambda_2$.

As an example, consider the single reaction RAF-1* + RKIP $\rightarrow$ RAF-1*/RKIP which describes the binding of Raf-1* and RKIP proteins. Let us call this reaction "bind" and assume that the reaction kinetics is the following:

$$m_1 + m_2 \rightleftharpoons m_3$$

where m_1,m_2, and m_3 are the concentrations of Raf-1*, RKIP, and Raf-1*/RKIP respectively.

Now consider the PRISM model for this system, listed as Model 1. The model begins with the keyword *stochastic* and consists of some preliminary constants (N and R), four modules: $RAF1Process$, $RKIPProcess$, $RAF1RKIPProcess$, and $Constants$, and a system description which states that the four modules should be run concurrently. The constant N defines the number of symbolic levels for protein concentrations (in this case, $4 = N + 1 = 3 + 1$). Consider the first three modules which represent the proteins of the same name. Each module has the form: a state variable which denotes the protein concentration, followed by a single transition named *bind*. The transition has the form *condition* $\rightarrow$ *rate: assignment*, meaning when the condition is true, then perform the assignment at the given rate. The rate for transitions of the first two modules is protein concentration multiplied by R, the rate for the third is 1. The assignments in the first two modules decrease the protein level by 1, it is increased by 1 in the third module. This corresponds to the fact that the rate of the reaction is determined by the concentrations of the reactants, and the reactants are *consumed* in the reaction to *produce* Raf-1*/RKIP. But, we must not forget that there is a fourth module, $Constants$, which simply defines constants for the reaction kinetics. In this case the module contains a "dummy" state variable called x, and one (always) enabled transition named *bind* which defines the constant rate (i.e. k/R) for the transition *bind*.

Model 1 Raf-1 binding with RKIP

```
stochastic

const int N = 3;
const double R = 1/N;

module RAF1Process
    RAF1: [0..N] init N;
    [bind] (RAF1>0) -> RAF1*R: (RAF1' = RAF1 - 1);
endmodule

module RKIPProcess
    RKIP: [0..N] init N;
    [bind] (RKIP>0) -> RKIP*R: (RKIP' = RKIP - 1);
endmodule

module RAF1RKIPProcess
    RAF1RKIP: [0..N] init 0;
    [bind] (RAF1RKIP < N) -> 1: (RAF1RKIP' = RAF1RKIP + 1);
endmodule

module Constants
    x: bool init true;
    [bind] (x=true) -> 0.8/R: (x'=true);
endmodule

system
    RAF1Process || RKIPProcess ||  RAF1RKIPProcess || Constants
endsystem
```

Since all four transitions have the same name, they will all have to *synchronise*, and when they do, the resulting transition corresponds directly to the dynamics of reaction described by the differential equation. For example, the first transition will occur with rate $\frac{N \cdot R \cdot N \cdot R \cdot 0.8}{R} = 2.4$ ($RAF1$ and $RKIP$ are initialised to N, $RAF1RKIP$ is initialised to 0).

In this simple PRISM model, all the proteins are involved in only one reaction. But in the RKIP inhibited ERK pathway, each protein is involved in several reactions. We model this quite easily by introducing different names ($r1, r2, \ldots$) for each reaction (and the corresponding transitions). Notice also that we can describe all the transitions of the processes independently of the number of symbolic levels: we simply make the appropriate comparison (in the precondition). The size of complete model depends on number of levels used to model concentrations. For the RKIP inhibited ERK pathway model the size of continuous time Markov chain is the following:

- for 4 levels of concentration: 273 states and 1316 transitions;
- for 6 levels of concentration: 1974 states and 12236 transitions;
- for 10 levels of concentration: 28171 states and 216282 transitions.

The full stochastic model for the RKIP inhibited pathway can be found at the following web page: `http://www.dcs.gla.ac.uk/~vvv/rkip.sm`. And MATLAB® implementation of this model is accessible as `http://www.dcs.gla.ac.uk/~vvv/rkip.m`.

4 Analysis

Simulation is the exploration of a *single* behaviour over a given time interval. It provides a good means of validating a model, and of exploring particular scenarios, but it has drawbacks.

In order to reason rigorously about unbounded time intervals and sets of behaviours, we use use a temporal logic. Temporal logics are powerful tools for expressing temporal queries which may be generic (e.g. state reachability, deadlock) or application specific (e.g. referring to variables representing application characteristics). For example, we can express queries such as *if a concentration reaches a certain level, will it remain at that level thereafter?*, or *if we vary the rate of a particular reaction, how does the network behave?*

Since we have a stochastic model, we employ the logic CSL (Continuous Stochastic Logic) [3], and the symbolic probabilistic model checker PRISM [4] to compute validity. We can not only check validity of logical properties, but using PRISM we can analyse open formulae, i.e. we can perform *experiments* as we vary instances of variables in a formula expressing a property.

CSL is a continuous time logic that allows one to express a probability measure that a temporal property is satisfied, in either transient behaviours or in steady state behaviours. We assume a basic familiarity with the logic. A short description of CSL are given in [4]. The $P_{\bowtie p}[\phi]$ properties are *transient*, that is, they depend on time; $S_{\bowtie p}[\phi]$ properties are *steady state*, that is they hold in the long run. To check the latter properties, we use a linear algebra package in PRISM to generate the steady state solution. Note that in this context steady state solutions are not (generally) single states, rather a network of states (with cycles) which define the probability distributions in the long run.

In the next section we use CSL and PRISM to formulate and check a number of biological queries about the RKIP inhibited ERK pathway.

We consider three different kinds of temporal property:

1. steady state analysis of stability of a protein i.e. a protein reaches and then remains within certain bounds,
2. steady state analysis of protein stability when varying reaction rates i.e. a protein is more likely to be stable for certain reaction rates,
3. transient analysis of protein activation sequence i.e. concentration peak ordering.

4.1 Stability of protein in steady state

We illustrate this type of property by considering the concentration of Raf-1*, as represented by the variable RAF1. Stability for this protein (within bounds $C-1, C+1$) is expressed by the formula:$(RAF1 \geq C-1) \wedge (RAF1 \leq C+1)$. where C is the level of interest. In other words, the level of Raf-1* is at most 1 increment/decrement step away from C.

In the steady state, we performed experiments to evaluate the probability of this condition holding as we varied the parameter C. The CSL formula is:

$$S_{=?}[(RAF1 \geq C-1) \wedge (RAF1 \leq C+1)].$$

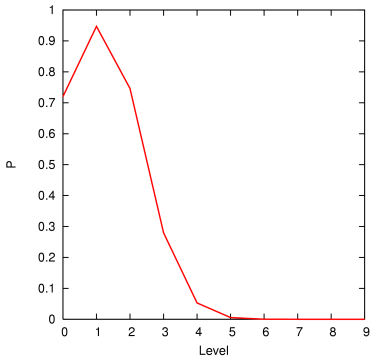

Fig. 2. Stability of Raf-1* in steady state

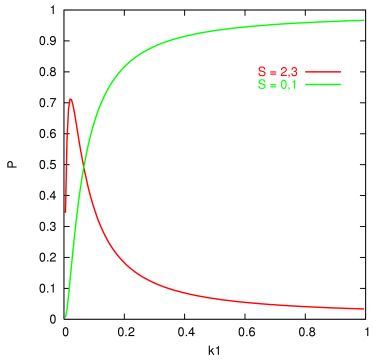

Fig. 3. Probability of Raf-1* stable state while varying the rate of binding to RKIP

The results are given Figure 2, with C ranging over ten (0..9) levels. The results illustrate that Raf-1* is most likely stable at level 1, with relatively high probability of stability at level 0 and level 2. It is unlikely to be stable around levels 3 and higher.

4.2 Protein stability in steady state while varying coefficients

This type of property is particularly useful during model fitting, i.e. fitting the model to experimental data. As an example, consider evaluating the probability of Raf-1* to be stable at level 2 or level 3 in steady state, whilst varying the performance of reaction which binds Raf-1* and RKIP. This reaction is denoted by $r1$. In this experiment we varied the rate of $r1$ (named k_1) over the interval $[0 \ldots 1]$. The stability property is expressed by:$S_{=?}[(RAF1 \geq 2) \wedge (RAF1 \leq 3)]$. Additionally we evaluated the probability for Raf-1* to be stable at levels 0 and 1. The formulae for this property is:$S_{=?}[(RAF1 \geq 0) \wedge (RAF1 \leq 1)]$.

Both experiments were run with six levels of concentration, the results are plotted in Figure 3. We conclude that Raf-1* is more unlikely to be stable at level 2 or level 3, when the binding rate for Raf-1* and RKIP increases, on the other hand, the probability of stability at level 0 or level 1 increases significantly with the binding rate increase.

It is quite important to emphasise the peak of red plot in Figure 3. This peak corresponds to the best fitting of the $k1$ rate (0.03) to keep the Raf-1* protein stable on levels 2 or 3.

4.3 Activation sequence analysis

This last example illustrates queries over several proteins, in particular it concerns sequences of protein activations. We draw our motivation from an examination of this pathway simulation. RAF-1*/RKIP complex reaches its peak at

Level 2 a little bit earlier than RAF-1*/RKIP/ERK-PP complex reaches its maximal value at Level 6. Moreover, RAF-1*/RKIP complex reaches Level 2 earlier than RAF-1*/RKIP/ERK-PP complex.

The logical formula to check this property is:

$$P_{=?}[(RAF1RKIPERKPP < M)\mathbf{U}(RAF1RKIP = C)]. \quad (1)$$

This property expresses "What is the probability that the concentration of Raf-1*/RKIP/ERK-PP complex will be less than Level M until Raf-1*/RKIP complex reaches concentration Level C?" The results of this query for the values of C within interval $[1 \dots 2]$ and the values of M within interval $[1 \dots 5]$ are plotted on the Figure 4; the line representing Raf-1*/RKIP/ERK-PP complex concentration at Level 5 is emphasised with crosses. The point we are especially interested in is $(C = 2, M = 5)$. The probability at this point is 0.9986, which means that the RAF-1*/RKIP complex reaches concentration Level 2 before RAF-1*/RKIP/ERK-PP complex reaches concentration Level 5, with the probability 99.86%.

Further analysis of this plot shows that it is possible, with the probability almost 96%, that the RAF-1*/RKIP complex will reach concentration Level 2 before RAF-1*/RKIP/ERK-PP complex reaches concentration Level 2.

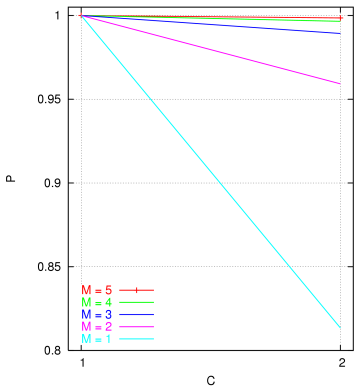

Fig. 4. Activation sequence analysis

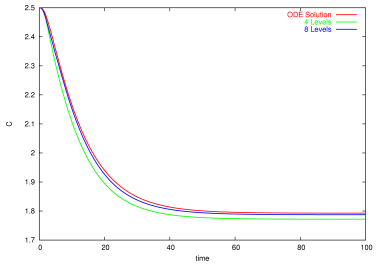

Fig. 5. MEKPP behaviour modelled with MATLAB® and PRISM.

This concludes our analysis, we now consider the correlation between simulations of our stochastic model and the ordinary differential equations model.

5 Comparison with ODE simulations

We can also use PRISM to carry out numerical calculations for simulation, using the concept of state rewards [5]. For comparison, we have implemented the

ODE model in the MATLAB® toolset and in Figure 5 we plot the behaviour of phosphorylated MEK, MEK-PP, over a time interval, using both the ODE model, and two instances of our stochastic model, with $N = 3$ and $N = 7$ (i.e. 4 and 8 discrete levels). We observe that as N increases, the closer the plots. Indeed, with $N = 7$ we do not distinguish the two plots by visual inspection. We have many more simulation results, but for brevity, these are excluded.

Levels	ϵ_a	ϵ_r	$C\epsilon_a$	$C\epsilon_a^2$
4	0.126 mM	0.280	21.557 mM	2.58
5	0.103 mM	0.217	17.569 mM	1.727
6	0.086 mM	0.176	14.582 mM	1.191
8	0.061 mM	0.122	10.402 mM	0.605
12	0.036 mM	0.071	6.042 mM	0.204

Fig. 6. Error measurements for PRISM model

To decide which number of levels is sufficient to make the two models indistinguishable, for any practical purposes, we define the following error metrics: maximal absolute error of simulation ϵ_a; maximal relative error of simulation ϵ_r; cumulative absolute error of simulation $C\epsilon_a$; cumulative square error of simulation $C\epsilon_a^2$. We have measured these errors with 200 data points in time interval [0..100]. Cumulative errors are measured for the protein complex Raf-1*/RKIP which has maximal absolute error. The results are shown in the Figure 6.

Of course experimental measurements also have associated error bars, thus we conclude that in this network, 7 or 8 levels are sufficient to make the two models indistinguishable, for all practical purposes. We conjecture that this is the case for any network of biochemical reactions without modifiers.

6 Discussion

In our stochastic model we have made an assumption that there are equal distributions of absolute concentrations in each of our discrete abstractions. We chose equal distributions because we have no information to the contrary, we are simply choosing abstractions over a continuous range. We note that this distribution gives us simulation results which align with the behaviour of the differential equations, thus there is no evidence to support a different distribution (at least for the example pathway). Our simulation results also show convergence with behaviour defined by the (mass action) differential equations, but it is quite simple to handle other kinds of kinetics, for example Henri-Michaelis-Menten kinetics.

A number of interesting (generic) temporal biological properties were proposed in [6], but we have not repeated that analysis here. Rather, we have concentrated on further properties which are specific to signalling networks models with discretized protein concentrations. Mainly, we have found steady-state analysis most useful, but we have also illustrated the use of transient properties (in 4.3).

PRISM has been a useful tool for model checking, experimentation, and even simulation. All computations have been tractable on a single standard processor. We note that for networks with inhibition (not exhibited in our example network), the computations became intractable, and we required a computational grid of some 90+ machines to carry out the simulations. This complexity is conditioned by the dynamics of an inhibition which includes division, thus very small changes of inhibitor concentration, when this tends to zero, causes quite significant changes in the reaction flux. Furthermore, this error may be amplified by feedback structures. Consequently, an enormous number of discrete levels is needed, in order to obtain a good approximation. At the moment we are looking for an alternative representation of negative feedbacks, in order to find a tractable solution for this problem.

7 Related Work

The most widely used models are systems of ordinary differential equations (ODEs) [1,7], but more recent approaches include using process calculi and algebras [8,9], Petri nets [10,11], and logics [12].

The π-calculus [13] has been used for modelling biochemical systems, with molecules and their domains represented by computational processes, and reactions by communication and channel passing. The π-calculus offers the ability to reconfigure communication, thus it is particularly suitable for systems in which communication evolves. Further developments include the stochastic π-calculus [8], and BioSPi, a hybrid system. The models developed thus far are for simulation only.

An alternative is proposed in [9] where the stochastic process algebra PEPA is used to model a pathway. This model also handles incomplete data. The main advantage is that using the algebra, different formulations of the model can be compared (by bisimulation). For example, one formulation relates clearly to the data, whereas another permits abstraction over sub-pathways. However, while it is possible to show how the algebraic models relate to ordinary differential equations, they cannot be used directly for simulation.

Petri nets are another class of modelling notations widely used to analyse biochemical networks [14]. A number of logical properties can be verified using Petri nets approaches [15,16], such as hybrid function Petri nets (HFPN) [10], time Petri nets [17], and stochastic Petri nets [18]. Hybrid function Petri nets are useful for illustrating system behaviour and quite mature simulation algorithms exist for these models. However, at the moment there are no algorithms for model checking hybrid function Petri nets. Stochastic Petri nets have almost the same expressiveness as our approach. Some of our experiments can be repeated with SPN, but there exist no general model checkers, thus the most of our CSL queries cannot be checked on stochastic Petri nets automatically. There are a number of model checking algorithms for time Petri nets [19], and many of the logical properties we consider in this paper could be verified using a time Petri nets approach. But, in order to do so, the description of our nonlinear system

behaviour would be approximated with linear time constraints, these would also be huge and inconvenient to read. We note that usually Petri nets modellers just neglect nonlinearity in biochemical systems.

The BIOCHAM workbench [6,12] provides an interface to the symbolic model checker NuSMV; the interface is based on a simple language for representing biochemical networks. The workbench provides mechanisms to reason about reachability of certain states, existence of partially described stable states, and some types of temporal behaviour. But this approach does not support quantitative model checking of the biochemical systems, and only qualitative (structural) queries can be verified.

8 Conclusions

We have described a new modelling and analysis approach for signal transduction networks in the presence of incomplete data. We model the dynamics of networks by continuous time Markov chains, making discrete approximations to protein concentrations. We describe the models in a high level language, using the PRISM modelling language: reactions are synchronous processes and concentrations are discrete, abstract quantities. Throughout, we have illustrated our approach with an example, the RKIP inhibited ERK pathway [1].

The main advantage of our approach is that using a (continuous time) stochastic logic and the PRISM model checker, we can perform quantitative analysis of queries such as *if a concentration reaches a certain level, will it remain at that level thereafter?* This approach offers considerably more expressive power than simulation. We can also perform standard simulations and we have compared our results with traditional ordinary differential equation-based (simulation) methods, as implemented in MATLAB®. An interesting result is that in the example pathway, only a small number of discrete data values are required to render the simulations practically indistinguishable. We have conjectured that in the absence of inhibition, this result will hold for any pathway represented as a stochastic transition system, using our approach.

Future work will be to prove that conjecture and to consider the addition of spatial dimensions (e.g. scaffolds) to our models.

Acknowledgements

This research is supported by the project *A Software Tool for Simulation and Analysis of Biochemical Networks*, funded by the DTI Beacon Bioscience Programme.

References

1. Cho, K.H., Shin, S.Y., Kim, H.W., Wolkenhauer, O., McFerran, B., Kolch, W.: Mathematical modeling of the influence of RKIP on the ERK signaling pathway. Lecture Notes in Computer Science **2602** (2003) 127–141

2. Kwiatkowska, M., Norman, G., Parker, D.: PRISM: Probabilistic Symbolic Model Checker. Lecture Notes in Computer Science **2324** (2002) 200–204
3. Aziz, A., Sanwal, K., Singhal, V., Brayton, R.: Model checking continuous time markov chains. ACM Transactions on Computational Logic **1** (2000) 162–170
4. Parker, D., Norman, G., Kwiatkowska, M.: PRISM 2.1 Users' Guide. The University of Birmingham (2004)
5. The University of Birmingham: PRISM Web page. http://www.cs.bham.ac.uk/∼dxp/prism/ (2005)
6. Chabrier, N., Fages, F.: Symbolic model checking of biochemical networks. Lecture Notes in Computer Science **2602** (2003) 149–162
7. Voit, E.O.: Computational Analysis of Biochemical Systems. Cambridge University Press (2000)
8. Priami, C., Regev, A., Silverman, W., Shapiro, E.: Application of a stochastic name passing calculus to representation and simulation of molecular processes. Information Processing Letters **80** (2001) 25–31
9. Calder, M., Gilmore, S., Hillston, J.: Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA. In Proceedings of Bio-Concur 2004 (2004)
10. Matsuno, H., Doi, A., Nagasaki, M., Miyano, S.: Hybrid petri net representation of gene regulatory network. Pacific Symposium on Biocomputing **5** (2000) 341–352
11. Peleg, M., Yeh, I., Altman, R.B.: Modelling biological processes using workflow and petri net models. Bioinformatics **18** (2002) 825–837
12. Chabrier-Rivier, N., Chiaverini, M., Danos, V., Fages, F., Schächter, V.: Modeling and querying biomolecular interaction networks. Theoretical Computer Science **325** (2004) 25–44
13. Regev, A., Silverman, W., Shapiro, E.: Representation and simulation of biochemical processes using π-calculus process algebra. Pacific Symposium on Biocomputing 2001 (PSB 2001) (2001) 459–470
14. Pinney, J., Westhead, D., McConkey, G.: Petri Net representations in systems biology. Biochem. Soc. Trans. **31** (2003) 1513 – 1515
15. Koch, I., Heiner, M.: Qualitative modelling and analysis of biochemical pathways with petri nets. Tutorial Notes, 5th Int. Conference on Systems Biology - ICSB 2004, Heidelberg/Germany (2004)
16. Heiner, M., Koch, I., Will, J.: Model validation of biological pathways using Petri nets, demonstrated for apoptosis. Biosystems **75** (2004) 15 – 28
17. Popova-Zeugmann, L., Heiner, M., Koch, I.: Modelling and analysis of biochemical networks with time petri nets. Informatik-Berichte der HUB Nr. 170 **1** (2004) 136 – 143
18. Goss, P.J., Peccoud, J.: Quantitative modeling of stochastic systems in molecular biology by using stochastic Petri nets. Proc. Natl. Acad. Sci. USA (Biochemistry) **95** (1998) 6750 – 6755
19. Yoneda, T., Ryuba, H.: CTL model checking of time Petri nets using geometric regions. IEICE Trans. Inf. and Syst. **E99-D, No. 3** (1998) 297–396