On a Continuous Degree of Satisfaction of Temporal Logic Formulae with Applications to Systems Biology

Aurélien Rizk, Grégory Batt, François Fages, and Sylvain Soliman

Projet Contraintes, INRIA Rocquencourt,
BP105, 78153 Le Chesnay Cedex, France.
`Firstname.Lastname@inria.fr`
`http://contraintes.inria.fr`

Abstract. Finding mathematical models satisfying a specification built from the formalization of biological experiments, is a common task of the modeller that techniques like model-checking help solving, in the qualitative but also in the quantitative case. In this article we propose to go one step further by defining a continuous degree of satisfaction of a temporal logic formula with constraints. We show how such a satisfaction measure can be used as a fitness function with state-of-the-art search methods in order to find biochemical kinetic parameter values satisfying a set of biological properties formalized in temporal logic. We also show how it can be used to define a measure of robustness of a biological model with respect to some specification. These methods are evaluated on models of the cell cycle and of the MAPK signalling cascade.

1 Introduction

Temporal logics [1, 2] have proven useful as specification languages for describing the behavior of a broad variety of systems ranging from electronic circuits to software programs, and more recently biological systems in either boolean [3–5], discrete [6], stochastic [7, 8] or continuous [9, 10, 4, 11] settings.

Because temporal logics allow us to express both qualitative (e.g. some protein is eventually produced) and quantitative (e.g. a concentration exceeds 10) information about time and systems variables, they provide a powerful specification language in comparison with the essentially qualitative properties considered in dynamical systems theory (e.g. multistability, existence of oscillations) or with the exact quantitative properties considered in optimization theory (e.g. curve fitting). In particular, these logics are well suited to the increasingly quantitative, yet incomplete, uncertain and imprecise information now accumulated in the field of quantitative systems biology.

This use of temporal logics relies on a logical paradigm for systems biology [12] which consists in making the following identifications:

biological model = *transition system*
biological properties = *temporal logic formulae*
biological validation = *model-checking*

In this paradigm, temporal logics have been used in many applications, either as query languages of large interaction maps such as Kohn's map of the cell cycle [5, 13] or gene regulatory networks [11], or as specification languages of biological properties known or inferred [14] from experiments, and used for validating models, discriminating between models and proposing new biological experiments [6], finding parameter values [9], or estimating robustness [15]. An important limitation of this approach is however due to the logical nature of temporal logic specifications and their boolean interpretation. A yes/no answer to a temporal logic query does not provide indeed any information on how far we are from satisfaction, nor how to guide the search to satisfy a formula. A measure of how close a model is to satisfy a property is needed.

In this paper, we define a continuous violation degree that quantifies how far from satisfaction an LTL formula is in a given model. In order to accommodate the various kinds of quantitative models defined by either ordinary or stochastic differential equation systems [16, 17], rule-based languages like SBML [18] or BIOCHAM [19, 20], hybrid Petri nets [21, 22], stochastic process calculi [23, 24], etc..., we represent the behavior of the system simply by numerical traces [14, 25, 9, 10], so our method is rather general. This notion of violation degree is then used for two applications in systems biology: the search of kinetic parameter values in a model, and the quantitative estimation of the robustness of a model by adapting the general framework of Kitano [26] to our temporal logic setting.

Section 2 presents the quantifier free fragment of first-order linear time logic with constraints over the reals, QFLTL($\mathbb{R}$), studied in [14] and used in this paper. Section 3 defines a real-valued degree of satisfaction of an LTL formula using a variable abstraction mechanism which replaces real valued constants in LTL formulae by QFLTL($\mathbb{R}$) variables, and using an aggregation function which composes the distances between the validity domain of these variables and the corresponding constants.

Section 4 shows how such a continuous degree of satisfaction of an LTL formula can be used as a fitness function in local search methods for searching kinetic parameter values in order to satisfy a temporal logic specification. We describe a gradient based method and use the state-of-the-art Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [27] to evaluate the method on models of the budding yeast cell cycle with 8 parameters and of the MAPK signaling cascade with 30 parameters and 7 unknown initial conditions.

In section 5 we propose a definition of a robustness degree of a property w.r.t. a set of model perturbations weighted by probabilities. This definition is inspired by the abstract definition of robustness proposed by Kitano for systems biology [26]. We develop it here in our temporal logic setting and illustrate its relevance by applying it to the previous model of the cell cycle.

2 Preliminaries on Linear Time Logic with Constraints over the Reals

2.1 LTL($\mathbb{R}$)

The *Linear Time Logic* LTL is a temporal logic [2] that extends classical logic with modal operators for qualifying when a formula is true in an infinite sequence of timed states, named a *trace*. The temporal operators are X ("next", for at the next time point), F ("finally", for at some time point in the future), G ("globally", for at all time points in the future), U ("until"), and W (" weak until"). These operators enjoy some simple duality properties, $\neg X\phi = X\neg\phi$, $\neg F\phi = G\neg\phi$, $\neg G\phi = F\neg\phi$, $\neg(\psi\ U\ \phi) = (\neg\phi\ W\ \neg\psi)$, $\neg(\psi\ W\ \phi) = (\neg\psi\ U\ \neg\phi)$. We have $F\phi = \mathit{true}\ U\ \phi$, $G\phi = \phi\ W\ \mathit{false}$.

A version of LTL with constraints over the reals, named LTL($\mathbb{R}$), has been proposed in [10, 9] to express temporal properties about molecular concentrations. The atomic formulae of LTL($\mathbb{R}$) are formed with inequality relations and arithmetic operators over the real values of molecular concentrations and of their derivatives. The precise syntax of LTL($\mathbb{R}$) is given in Table 1. As negations and implications can be eliminated by propagating the negations down to the atomic constraints in the formula, we will assume in the following that all LTL($\mathbb{R}$) formulae are in negation free normal form.

Formula ::=	Atom $\mid$ Formula $\wedge$ Formula $\mid$ Formula $\vee$ Formula
	$\mid$ Formula $\Rightarrow$ Formula $\mid$ $\neg$ Formula
	$\mid$ X Formula $\mid$ F Formula $\mid$ G Formula
	$\mid$ Formula U Formula $\mid$ Formula W Formula
Atom ::=	Value Op Value
Op ::=	$<$ $\mid$ $>$ $\mid$ $\leq$ $\mid$ $\geq$
Value ::=	float $\mid$ [molecule] $\mid$ d[molecule]/dt $\mid$ Time
	$\mid$ Value + Value $\mid$ Value - Value $\mid$ - Value $\mid$ Value $\times$ Value
	$\mid$ Value / Value $\mid$ Value ˆ Value

Table 1. Syntax of LTL($\mathbb{R}$) formulae.

For instance, `F([A]>10)` expresses that the concentration of `A` eventually gets above the threshold value 10. `G([A]+[B]<[C])` expresses that the concentration of C is always greater than the sum of the concentrations of A and B. Oscillation properties, abbreviated as `oscil(M,K)`, are defined as a change of sign of the derivative of M at least K times:
`F((d[M]/dt > 0)` $\wedge$ `F((d[M]/dt < 0)` $\wedge$ `F((d[M]/dt > 0)...)))`

LTL($\mathbb{R}$) formulae are interpreted over infinite traces of the form

$$(< t_0, \boldsymbol{x}_0, d\boldsymbol{x}_0/dt >, < t_1, \boldsymbol{x}_1, d\boldsymbol{x}_1/dt >, ...)$$

which give at discrete time points t_i, the concentration values $\boldsymbol{x}_i$ of the molecules, and the values of their first derivatives $d\boldsymbol{x}_i/dt$. Whereas LTL($\mathbb{R}$) formulae are interpreted over infinite traces, the ones we consider are always finite. For instance, in a model described by a system of ordinary differential equations (ODE), and under the hypothesis that the initial state is completely defined, numerical integration methods (such as Runge-Kutta or Rosenbrock method for stiff systems) provide a finite simulation trace. To extend it to an infinite trace, we adopted the solution of adding a loop on the last state, with the assumption that the finite time horizon considered for the numerical integration is sufficiently large to check the properties at hand.

It is worth noticing that the semantics of the "next" operator refers to the next time point on the trace and that in adaptive step size integration methods of ODE systems, the step size $t_{i+1} - t_i$ is not constant but determined through an estimation of the error made by the discretization.

Formally, the truth value of an LTL($\mathbb{R}$) formula in a trace π is given in Table 2. These truth values can be computed on traces by model-checking [9].

$s \models \alpha$	iff α is a propositional formula and α is true in the state s,
$\pi \models \phi$	iff $s \models \phi$ where s is the first state of π,
$\pi \models X\psi$	iff $\pi^1 \models \psi$,
$\pi \models \psi\ U\ \psi'$	iff there exists $k \geq 0$ s.t. $\pi^k \models \psi'$ and $\pi^j \models \psi$ for all $0 \leq j < k$.
$\pi \models \psi\ W\ \psi'$	iff either for all $k \geq 0$, $\pi^k \models \psi$. or there exists $k \geq 0$ s.t. $\pi^k \models \psi \wedge \psi'$ and for all $0 \leq j < k$, $\pi^j \models \psi$.
$\pi \models \neg\psi$	iff $\pi \not\models \psi$,
$\pi \models \psi\ \wedge\ \psi'$	iff $\pi \models \psi$ and $\pi \models \psi'$,
$\pi \models \psi\ \vee \psi'$	iff $\pi \models \psi$ or $\pi \models \psi'$,
$\pi \models \psi \Rightarrow \psi'$	iff $\pi \models \psi'$ or $\pi \not\models \psi$,

Table 2. Inductive definition of the truth value of an LTL($\mathbb{R}$) formula in a trace π.

2.2 QFLTL($\mathbb{R}$)

In [14], the quantifier free fragment of the first-order extension of LTL($\mathbb{R}$), named QFLTL($\mathbb{R}$), has been considered for the purpose of analyzing numerical data time series in temporal logic and computing automatically LTL($\mathbb{R}$) specifications from experimental traces. Syntactically, QFLTL($\mathbb{R}$) adds variables to atomic expressions with the following grammar:

$$\text{Atom} ::= \text{ Value Op Value} \mid \text{Value Op Variable}$$

For instance, the QFLTL($\mathbb{R}$) formula $G([A] < v)$ expresses the constraint that v is greater than the maximum concentration of A. The restriction that a variable can only appear in the right-hand side of a comparison is motivated by computability results.

As usual, the semantics of a QFLTL($\mathbb{R}$) formula containing variables is defined by its ground instances which are LTL($\mathbb{R}$) formulae. Given a trace π and a QFLTL($\mathbb{R}$) formula $\phi(\boldsymbol{x})$ over a vector $\boldsymbol{x}$ of v real-valued variables, the *constraint satisfaction problem*, $\exists \boldsymbol{x} \in \mathbb{R}^v\ (\phi(\boldsymbol{x}))$, is the problem of determining the valuations $\boldsymbol{v}$ of the variables for which the formula ϕ is true. In other words, we look for the domain of validity $\mathcal{D}_\phi \subset \mathbb{R}^v$ such that $\pi \models \forall \boldsymbol{v} \in \mathcal{D}_\phi\ (\phi(\boldsymbol{v}))$.

In [14], an LTL($\mathbb{R}$) *model-checking algorithm* has been generalized to a QFLTL($\mathbb{R}$) *constraint solving algorithm* which computes the exact domain of validity $\mathcal{D}_\phi$ for any QFLTL($\mathbb{R}$) formula ϕ, in time $O((nf)^{2v})$ where v is the number of variables in ϕ, f the size of the formula and n the length of the trace. This algorithm is at the heart of the methods presented in the following sections.

3 Continuous Satisfaction Degree of LTL($\mathbb{R}$) Formulae

In order to evaluate numerically the adequateness of a model w.r.t. a temporal logic specification, we introduce a continuous violation degree relating a trace of the model to the given constraint LTL formula. When the model satisfies its specification the degree will be null, and the farther the traces from the expected behavior, the biggest the violation degree.

3.1 Variable Abstraction

Our definition of the violation degree of an LTL($\mathbb{R}$) formula relies on an abstraction of the constants occurring in the formula by variables. Starting from an LTL specification ϕ of the expected behavior of a system, we transform it into a QFLTL formula ϕ^* by mapping the constants (i.e. real numbers corresponding to concentration thresholds, amplitudes, etc.) $c_1, \ldots, c_n$ appearing in ϕ, to distinct variables $x_1, \ldots, x_n$. It is worth noting that ϕ^* is a QFLTL formula that can also be seen as a function over $\mathbb{R}^n$ associating a closed LTL formula to an instantiation of its variables.

Definition 1. *Given an LTL($\mathbb{R}$) formula ϕ and a QFLTL abstraction ϕ^*, the* objective*, noted $var(\phi)$, is the single point in the* variable space $\mathbb{R}^n$ *of ϕ^*, with x_i equal to c_i for all $1 \leq i \leq n$.*

Example 1. Consider the LTL formula $\phi = F([A] > 20)$ indicating that from experiments it was observed that after some time the concentration of compound A becomes greater than 20. We get $\phi^* = F([A] > x)$ as a QFLTL formula and $\mathbb{R}$ as variable space. We have $var(\phi) = 20$.

Because of the syntactical restriction imposed on the occurrences of variables in the right-hand sides of the inequalities in QFLTL($\mathbb{R}$) formulae, the transformation from ϕ to ϕ^* cannot always be done automatically. However for polynomial expressions over concentrations and derivatives, one can apply the following transformation on atomic expressions:

$$(e_1\ Op\ e_2)^* \ = \ e\ Op\ x$$

where Op is an inequality operator, $e_1 - e_2$ is a polynomial in the concentrations and derivatives with term c of degree 0, $e = e_1 - e_2 - c$ and x is a new variable introduced for the term $-c$.

More generally, ϕ^* will be a QFLTL formula given with a variable space $\mathbb{R}^n$ that may include variables defined from other ϕ^* variables with linear inequalities, allowing some rescaling between variables if necessary. The objective $var(\phi)$ will be defined explicitly through an instantiation of those variables, i.e. a point in the variable space.

Example 2. Consider the QFLTL formula $\phi^* = F([A] \geq v) \wedge F([A] \leq w)$, let us define the amplitude variable $amp = v - w$ and use it as the only variable for our variable space $\mathbb{R}$. We can set as objective that the amplitude of variation of the compound A is at least 10 with $var(\phi) = 10$.

3.2 Quantitative Satisfaction

Given a QFLTL formula ϕ^* and a numerical trace T, the QFLTL($\mathbb{R}$) constraint solving algorithm of [14] computes the exact *domain of validity* for ϕ^* on T, as the domain of the variables $D_{\phi^*}(T) \subset \mathbb{R}^n$.

Definition 2. *The* violation degree *of a numerical trace T to an LTL formula ϕ, noted $vd(T, \phi)$ is the Euclidean distance between $D_{\phi^*}(T)$ and $var(\phi)$, i.e. $min_{v \in D_{\phi^*}(T)} d(v, var(\phi))$.*

Example 3. In the example 1 and given a mathematical model of our system, let us suppose that the QFLTL constraint solving algorithm applied to ϕ^* on simulation trace T computes $D_{\phi^*}(T) =]-\infty, 15]$ as domain for variable x. Since $var(\phi) = 20$ we get $vd(T, \phi) = 5$, i.e. the violation degree is 5 since the compound reaches a maximum of 15 whereas the formula expresses that the threshold 20 be reached.

For the specification of example 2, suppose that the constraint solving computes the domains of v and w: $D(v) =]-\infty, 15]$ and $D(w) = [10, +\infty[$. For this formula ϕ, the maximum value of $D(v)$ represents the maximum value of $[A]$ and the minimum value of $D(w)$ its minimum value in the trace. The domain for variable amp is $D_{\phi^*}(T) =]-\infty, 5]$ since we know that $amp = v - w$, and thus, since $var(\phi) = 10$, we obtain $vd(T, \phi) = 5$, i.e. the amplitude of the curve is 5 whereas we wanted it to be 10.

Note that if T is such that ϕ is satisfied then $vd(T, \phi) = 0$ since $var(\phi) \in D_{\phi^*}(T)$. However when ϕ is not valid on T, the violation degree vd provides a quantitative measurement of its degree of non satisfaction. The use of this measure is illustrated in the following sections to improve parameter search for biological models and to define a quantitative notion of robustness of a system w.r.t. a temporal logic formula.

4 Kinetic Parameter Search using Violation Degree

The violation degree provides a measure of how far a given numerical trace is from an LTL specification. It is thus quite natural to use this measure to guide the search when trying to satisfy such a formula by replacing the scanning of parameter values described in [9] by a much more efficient local search method which makes evolve parameter sets by exploring a neighborhood of the current parameter set and by choosing the one which minimizes the violation measure.

4.1 Principle

Let us consider an LTL formula ϕ, an SBML/BIOCHAM reaction model with initial conditions and known parameter values, a set of unknown parameters to explore and for each of those an interval of search. We consider the problem of finding a set of values of the unknown parameters such that the violation degree of the corresponding trace T obtained by numerical simulation is $vd(T, \phi) = 0$.

A generic optimization algorithm for parameter search can be described as follows:

Algorithm 1 (generic parameter search method)

1. *Set the current point in the parameter space to a random point belonging to the provided search box, compute a numerical simulation with trace T and the corresponding violation degree $vd(T, \phi)$;*
2. *if $vd = 0$ jump to 5.*
3. *for each point in a defined neighborhood of the current point, compute a trace and its violation degree;*
4. *based on the violation degrees of the neighbors, determine the next point of the iteration, set the current point to this point, update current vd and go to 2.*
5. *Return the current point in the parameter space.*

This procedure can be interrupted after a given number of steps, returning the best parameter set (minimizing the violation degree). It can also be restarted with a new initial point (step 1) several times in order to diversify the search.

A naive method would be to define as neighborhood of the current parameter state the parameter sets obtained by modifying one parameter by values $\pm\delta$; and to choose as next parameter set the best neighbor.

More efficient instances of this algorithm can be obtained however, by combining state-of-the-art nonlinear optimization methods with the computation of our violation degree used as a blackbox fitness function. In the following sections, we use the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) of Hansen and Ostermeier [27]. This method uses a probabilistic definition of the neighborhood, and stores information in a covariance matrix in order to replace the approximate gradient and Hessian of a quasi-Newton method by an evolutionary algorithm.

4.2 Evaluation on Cell Cycle Models

In this section we present the application of the parameter search method outlined above to the budding yeast cell cycle model of [28]. This model displays how proteins `cdc2` and `cyclin` interact to form the heterodimer `Cdc2-Cyclin~{p1,p2}` known as maturation promoting factor (MPF) and playing a key role in the control of mitotic cycles. The reaction rules of the model are the following:

```
MA(k1)       for                      _ => Cyclin.
MA(k3)       for  Cyclin + Cdc2~{p1} => Cdc2-Cyclin~{p1,p2}
MA(k4p)      for Cdc2-Cyclin~{p1,p2} => Cdc2-Cyclin~{p1}
AUTOCAT(k4)  for Cdc2-Cyclin~{p1,p2} => Cdc2-Cyclin~{p1}
MA(k6)       for Cdc2-Cyclin~{p1}    => Cyclin~{p1} + Cdc2
MA(k7)       for         Cyclin~{p1} => _
MA(k8)       for                Cdc2 => Cdc2~{p1}
MA(k9)       for           Cdc2~{p1} => Cdc2
```

`MA(k)` denotes Mass Action law kinetics with parameter `k` while `~{p1}` and `~{p1,p2}` denote phosphorylated forms of a molecule. The rate of reaction 4 is described by: `AUTOCAT(k4)= k4*[Cdc2-Cyclin~{p1,p2}]*[Cdc2-Cyclin~{p1}]^2`.

We use as reference point $\boldsymbol{k}_{Tyson}$ the values of the kinetic parameters determined in [28]. The simulation for $\boldsymbol{k}_{Tyson}$ of the system of ODEs extracted from these rules, given in appendix, is displayed in Figure 1. The total amount of cyclin presents oscillations of period 35 while MPF exhibits activity peaks with same period.

Using the optimization method CMA-ES together with our violation degree as a parameter search method we wonder whether it is possible to find values of the kinetic parameters corresponding to higher MPF peaks or oscillations with higher amplitudes or shorter periods.

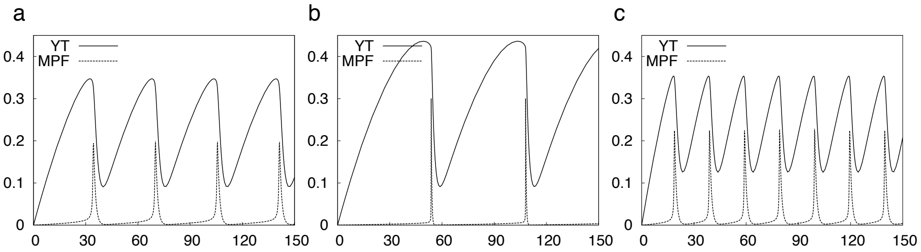


Fig. 1. Dynamical behavior of the cell cycle model. The plots represent total cyclin (YT) and maturation promoting factor (MPF). (a) Oscillatory behavior obtained with parameter values $\boldsymbol{k}_{Tyson}$. (b) Higher MPF peaks obtained with $\boldsymbol{k}^{*}_{Tyson}$ (solution of problem S1). (c) Shorter oscillations period obtained with $\boldsymbol{k}^{*}_{4}$ (solution of problem S4).

Search problem S1 : higher MPF peaks (2 parameters unknown)

Two parameters, k4 and k6, have been found in [28] to play a particular role for the existence of oscillations. Depending on their values the system exhibits either a steady state behavior or limit cycle oscillations. We wonder whether it is possible to obtain higher MPF peaks by changing values of k4 and k6 only, all other parameters remaining at the value $\boldsymbol{k}_{Tyson}$ chosen in [28]. More precisely, we want to reach at least MPF peaks of 0.3, the maximum amount of MPF for $\boldsymbol{k}_{Tyson}$ being 0.19.

Therefore we define the LTL specification : $\phi_1 = F([MPF] > 0.3)$ with the corresponding QFLTL formula being :

$$\phi_1^* = F([MPF] > max)$$

The variable space associated to ϕ_1^* is $\mathbb{R}$ and corresponds to the sole variable max. The objective is $var(\phi) = 0.3$, i.e the target peak value of MPF is 0.3. We have been able to find valid parameter values, denoted $\boldsymbol{k}_{Tyson}^*$, satisfying $vd(T, \phi_1^*) = 0$ where T is the corresponding simulated trace (see Figure 1b). $\boldsymbol{k}_{Tyson}^*$ is given in Table 3.

As the plot shows, for these parameter values essential features of the curve, especially repeated MPF peaks, are conserved although it was not enforced by the specification. In particular, a constantly growing amount of MPF would have also resulted in a null violation degree of this formula.

All computations have been performed on an Intel Core 2 Duo 2Ghz with 2Go RAM. Note that as the optimization method CMA-ES uses a probabilistic neighborhood two consecutive runs can yield different results. In this example answers are typically obtained in less than 1 minute after around 250 numerical simulations and violation degree computations.

Search problem S2 : amplitude of MPF oscillation (2 parameters unknown)

In this example we refine the previous query by constraining the minimum level of MPF. We search for k4 and k6 values that preserve at least two periods of MPF oscillations having same amplitudes as those observed for $\boldsymbol{k}_{Tyson}$.

$$\begin{aligned}\phi_2^* = F(\quad & [MPF] > max \wedge F([MPF] < min \\ & \wedge F([MPF] > max \wedge F([MPF] < min))))\end{aligned}$$

In order to specify that the amplitude is at least 0.19, we use the variable space $\mathbb{R}$ corresponding to only one variable, $amp = max - min$, and set $var(\phi) = 0.19$, i.e the target amplitude is 0.19. This value corresponds to the amplitude obtained for $\boldsymbol{k}_{Tyson}$. Starting from a different value $\boldsymbol{k}_2$ for k4 and k6, we try to recover the behavior of $\boldsymbol{k}_{Tyson}$. We found such parameters (in 11 s), given in Table 3 and referred to as $\boldsymbol{k}_2^*$.

To illustrate the path followed during the search from $\boldsymbol{k}_2$ to $\boldsymbol{k}_2^*$ we computed the violation degree landscape in the k4, k6 parameter space. The resulting landscape is displayed in Figure 2. Note that as all constants of the formula have been abstracted by variables, the violation degree can only be finite. In particular when no oscillations are present in the trace amp will be equal to 0, thus leading to a violation degree of 0.19. Regions where the violation degree is 0.19 correspond to regions of steady state behavior whereas regions with a violation degree between 0 and 0.19 correspond to regions of oscillations.

Under mild assumptions Tyson determined linear equations defining a region in the k4, k6 plane where oscillations occurs, also represented in Figure 2. Our results are fully consistent with his analytical analysis, and provide more information on the amplitude of oscillation w.r;t. parameters k4 and k6.

Search problem S3 and S4 : amplitude and period of oscillations (all 8 parameters unknown)

To illustrate the scalability of the method we carry out two parameter searches on all 8 parameters of the model. The first one (problem S3) is the same query as above with formula ϕ_2^* but with all parameters unknown. The second one is a more complex query used to find shorter oscillation periods of `Cdc2` :

$$\begin{aligned}\phi_3^* = F(\quad & d([Cdc2])/dt < 0 \wedge X(d([Cdc2])/dt > 0 \wedge Time = t1 \\ & \wedge X(F(d([Cdc2])/dt > 0 \wedge X(d([Cdc2])/dt < 0 \wedge Time = t2))\end{aligned}$$

To specify that the target period is 20, we use the variable space $\mathbb{R}$ corresponding to the variable $per = t2 - t1$ with target $var(\phi) = 20$. Search problem S3 starts from parameter values $\boldsymbol{k_3}$ satisfying the constraints on their order of magnitude given in [28]. $\boldsymbol{k_3}$ does not give rise to oscillations. Search problem S4 starts from $\boldsymbol{k_4} = \boldsymbol{k_{Tyson}}$. In both cases parameter values are found satisfying the query (in 30 s for S3 and 350 s for S4). Results are given in Table 3.

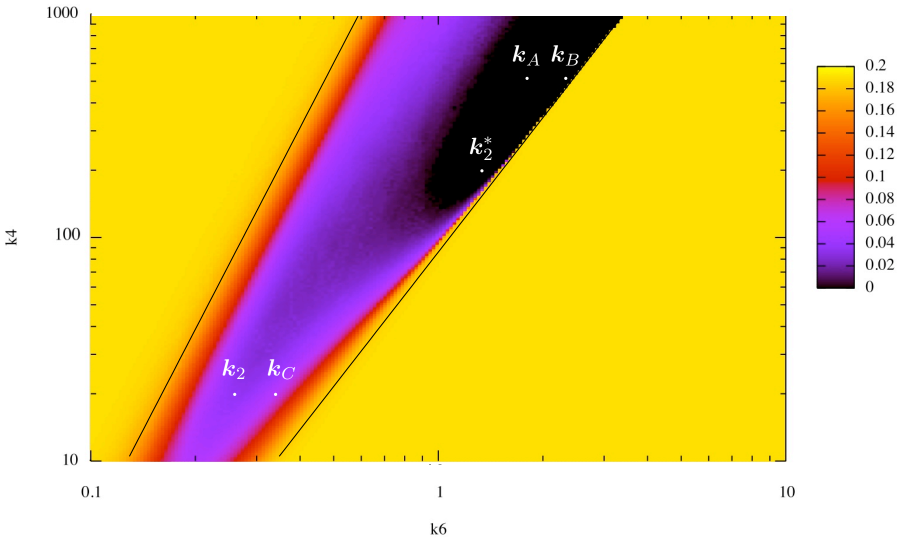


Fig. 2. Violation degree landscape of problem S2. This violation degree measures amplitude of oscillations. Non oscillating regions have highest violation degree.

	S1		S2		S3		S4	
	Initial values	Result	Initial values	Result	Initial values	Result	Initial values	Result
$vd(T, \phi)$	0.11	0	0.04	0	0.19	0	15.1	4.90e-4
Parameters	$\boldsymbol{k}_{tyson}$	$\boldsymbol{k}^*_{tyson}$	$\boldsymbol{k}_2$	$\boldsymbol{k}^*_2$	$\boldsymbol{k}_3$	$\boldsymbol{k}^*_3$	$\boldsymbol{k}_4$	$\boldsymbol{k}^*_4$
k1	1.50e-2	1.50e-2	1.50e-2	1.50e-2	1.00e-2	**1.14e-2**	1.50e-2	**2.41e2**
k3	2.00e2	2.00e2	2.00e2	2.00e2	1.00e2	**1.13e2**	2.00e2	**2.83e2**
k4p	1.80e-2	1.80e-2	1.80e-2	1.80e-2	1.00e-2	**8.77e-3**	1.80e-2	**2.24e-2**
k4	1.80e2	**8.99e2**	2.00e1	**1.94e2**	1.00e2	**1.82e2**	1.80e2	**2.28e2**
k6	1.00	**3.23**	0.25	**1.41**	1.00	**4.17e-1**	1	**1.13**
k7	0.60	0.60	0.60	0.60	1.00	**1.37**	0.60	**5.99e-1**
k8	1.00e2	1.00e2	1.00e2	1.00e2	1.00e3	**8.99e2**	1.00e2	**1.42e2**
k9	1.00e2	1.00e2	1.00e2	1.00e2	1.00e2	**8.44e1**	1.00e2	**6.94e1**

Table 3. Resulting parameter values for search problems S1, S2, S3 and S4.

4.3 Evaluation on MAPK Signal Transduction Model

The MAPK signal transduction model [29] is used to test the scalability of the parameter search method on a larger model. This model, made of a cascade of phosphorylation reactions, consists of the following rules :

```
(MA(k1), MA(k2)) for RAF + RAFK <=> RAF-RAFK.
(MA(k3),MA(k4)) for RAF~{p1} + RAFPH <=> RAF~{p1}-RAFPH.
(MA(k5),MA(k6)) for MEK~$P + RAF~{p1} <=> MEK~$P-RAF~{p1}
   where p2 not in $P.
(MA(k7),MA(k8)) for MEKPH + MEK~{p1}~$P <=> MEK~{p1}~$P-MEKPH.
(MA(k9),MA(k10)) for MAPK~$P + MEK~{p1,p2} <=> MAPK~$P-MEK~{p1,p2}
   where p2 not in $P.
(MA(k11),MA(k12)) for MAPKPH + MAPK~{p1}~$P <=> MAPK~{p1}~$P-MAPKPH.
MA(k13) for RAF-RAFK => RAFK + RAF~{p1}.
MA(k14) for RAF~{p1}-RAFPH => RAF + RAFPH.
MA(k15) for MEK~{p1}-RAF~{p1} => MEK~{p1,p2} + RAF~{p1}.
MA(k16) for MEK-RAF~{p1} => MEK~{p1} + RAF~{p1}.
MA(k17) for MEK~{p1}-MEKPH => MEK + MEKPH.
MA(k18) for MEK~{p1,p2}-MEKPH => MEK~{p1} + MEKPH.
MA(k19) for MAPK-MEK~{p1,p2} => MAPK~{p1} + MEK~{p1,p2}.
MA(k20) for MAPK~{p1}-MEK~{p1,p2} => MAPK~{p1,p2} + MEK~{p1,p2}.
MA(k21) for MAPK~{p1}-MAPKPH => MAPK + MAPKPH.
MA(k22) for MAPK~{p1,p2}-MAPKPH => MAPK~{p1} + MAPKPH.
```

We denote by k_{MAPK} the set of kinetic parameter values used as reference for this model.

Search problem S5 : curve fitting at specific time points (22 parameters unknown)

In this example, we investigate the use of our parameter search method as a curve fitting tool at specific time points, on 22 parameter values. In order to express the classical distance between two curves at time points 30 and 60 for instance, we use the following pattern of formulae :

$$\begin{aligned}\phi_4^* = G(\quad & Time = 30 \rightarrow [MEK - RAF\tilde{}\{p1\}] = u \\ & \wedge Time = 60 \rightarrow [MEK - RAF\tilde{}\{p1\}] = v)\end{aligned}$$

The parameter space of this formula is $\mathbb{R}^2$ is defined by the two variables u and v. We set target $var(\phi)$ to the target values of $[MEK - RAF\tilde{}\{p1\}]$ at time 30 and 60. Note that this formula can be extended to any number of time points and molecules in order to perform a complete curve fitting, if it is relevant.

This pattern of formulae can be used to search the values of all the 22 parameters of the model to fit the concentration $[MEK - RAF\tilde{}\{p1\}]$ at six time points. The objective values for these time points are the values of the original model, obtained by simulation with the original parameters k_{MAPK}. The initial values for the search are some random altered values $k_{MAPK_{alt}}$. Numerical simulations obtained with k_{MAPK}, $k_{MAPK_{alt}}$ and the resulting parameter values are given in Figure 3. It took 290 s to obtain the result. This shows that the search method scales up well with the dimension of the parameter space, in comparison with the parameter scanning method which has an exponential time complexity in the number of parameters. Here, the computation time is more dependent on the type of problem (formula used and initial values of the parameters) and on the landscape of the violation degrees than on the number of parameters.

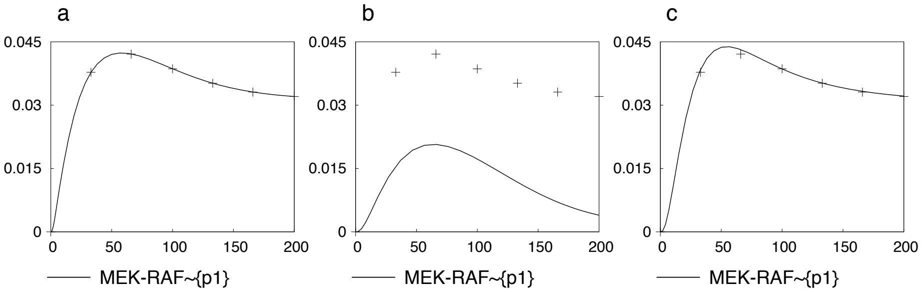


Fig. 3. Dynamical behavior of the MAPK model. The curves display $[MEK - RAF\tilde{}\{p1\}]$. (a) Reference curve obtained with k_{MAPK} (b) Simulated curve obtained with altered parameter values $k_{MAPK_{alt}}$. Points are the reference values taken from curve (a). (c) Simulated curve obtained after curve fitting (solution of problem S5).

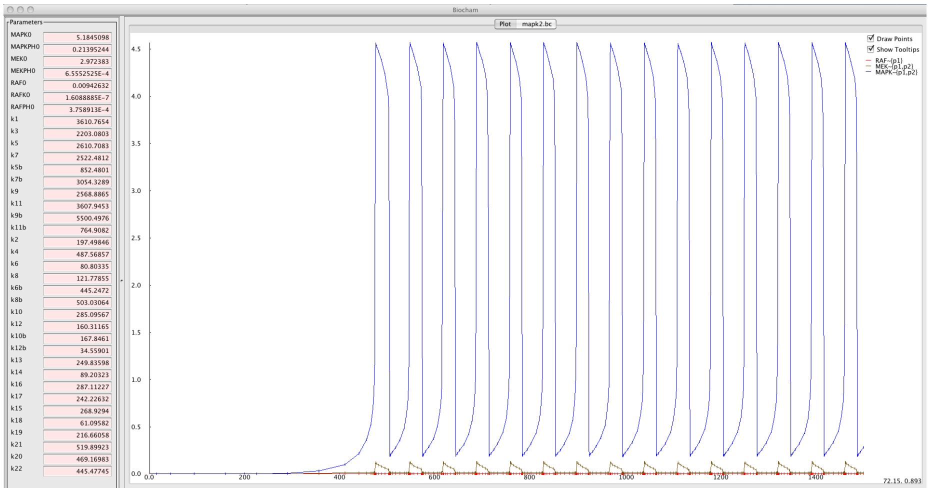
Search problem S6 : find oscillations (30 kinetic parameters and 7 initial conditions unknown)

In [30], oscillations have been found in the MAPK cascade model of [29] although this model does not contain any negative feedback reaction. This does not contradict Thomas' necessary condition for sustained oscillations as such a purely directional cascade does contain negative feedback in its influence graph as shown in [31] and analyzed in [32]. However, to know whether these negative circuits in the influence graph are functional, one needs to search for kinetic parameter values and initial conditions that exhibit sustained oscillations.

Just by defining the following formula:

$$\phi_5^* = F(\ [MAPKp1p2] > max \wedge F([MAPKp1p2] < min))$$

using the variable space $\mathbb{R}$ for the single variable $amp = max - min$, and by asking that the amplitude be at least 0.5, setting $var(\phi) = 0.5$, parameter values leading to sustained oscillations, such as the ones depicted in Figure 4, were found in a few minutes.



Parameters

MAPK0	5.1845098
MAPKPH0	0.21395244
MEK0	2.972383
MEKPH0	6.5552525E-4
RAF0	0.00942632
RAFK0	1.6088885E-7
RAFPH0	3.758913E-4
k1	3610.7654
k3	2203.0803
k5	2610.7083
k7	2522.4812
k5b	852.4801
k7b	3054.3289
k9	2568.8865
k11	3607.9453
k9b	5500.4976
k11b	764.9082
k2	197.49846
k4	487.56857
k6	80.80335
k8	121.77855
k6b	445.2472
k8b	503.03064
k10	285.09567
k12	160.31165
k10b	167.8461
k12b	34.55901
k13	249.83598
k14	89.20323
k16	287.11227
k17	242.22632
k15	268.9294
k18	61.09582
k19	216.66058
k21	519.89923
k20	469.16983
k22	445.47745

☑ Draw Points
☑ Show Tooltips

Fig. 4. Oscillations of MAPK found with CMA-ES in BIOCHAM

5 Quantitative Robustness Analysis

5.1 Principle

We have seen in the previous section that our notion of violation degree allows us to use optimization techniques to efficiently guide parameter search given temporal logic properties. Here, we show that the notion of violation degree also allows us to define in a mathematically precise way a degree of robustness of a systems behavior described in temporal logic w.r.t. a set of perturbations, and estimate it computationally. This robustness degree is defined as the inverse of the average violation degree of the property of interest over all admissible perturbations, possibly weighted by their probabilities. This definition is an adaptation of the general definition given by Kitano [26] to our temporal logic setting. Formally, using the notations introduced in previous sections, we set:

Definition 3. *Let P be a set of perturbations, $prob(p)$ be the probability of perturbation p, $T(p)$ be the timed trace of the system under perturbation $p \in P$. The* robustness degree *$R_{\phi,P}$ of a property ϕ with respect to P is the real value*

$$R_{\phi,P} = \left(\int_{p \in P} vd(T(p), \phi) prob(p) dp \right)^{-1}$$

If the set of perturbations is finite (eg, gene knock outs), the robustness degree is simply the inverse of a finite weighted sum and can be exactly computed. If the set of perturbations is infinite, the robustness degree can be estimated by computing the violation degree between the behavior of the perturbed system $T(p)$ and the specification ϕ for sufficiently many perturbations.

5.2 Evaluation on Cell Cycle Model

Using the same cell cycle model as in section 4.2, we compare the robustness of oscillation properties with regard to perturbations of parameter values k4 and k6 for different points in the parameter space.

We consider that parameter values for k4 and k6 are normally distributed around their reference value with coefficient of variation equal to 0.2. We also enforce that $k4 \geq 0 \wedge k6 \geq 0$. We examine the robustness of the property expressed by ϕ_2^*, that is, MPF oscillations are of amplitude at least 0.19.

The robustness degree of this property is compared for three different values of k4 and k6. These three points in the parameter space of k4 and k6 are indicated by the three points $\boldsymbol{k}_A$, $\boldsymbol{k}_B$ and $\boldsymbol{k}_C$ in Figure 2. In all cases, the estimation of the robustness degree is done by computing the mean value of the violation degree for 500 samples.

The estimated degree of robustness for parameters $\boldsymbol{k}_A$, $\boldsymbol{k}_B$ and $\boldsymbol{k}_C$ are respectively 133, 12.9 and 13.5. This is consistent with the location of points $\boldsymbol{k}_A$, $\boldsymbol{k}_B$ and $\boldsymbol{k}_C$. Perturbations around point $\boldsymbol{k}_A$ have high probabilities of staying in the region satisfying the specification whereas perturbations around point $\boldsymbol{k}_B$ have high probabilities of moving the system to the region with no oscillation. $\boldsymbol{k}_C$ is more robust than $\boldsymbol{k}_B$ even though, as opposed to $\boldsymbol{k}_B$, its violation degree is non null. This can be explained by the abrupt transition between oscillating and non oscillating regions near $\boldsymbol{k}_B$ compared to the smoother transition near $\boldsymbol{k}_C$.

The robustness degree can be estimated for perturbations on any number of parameters. For instance, by computing a robustness estimate for perturbations on all parameters, with coefficient of variation 0.2 for specification ϕ_2^* and parameter values $\boldsymbol{k}_{Tyson}$ and $\boldsymbol{k}_3$, the estimated robustness degrees for $\boldsymbol{k}_{Tyson}$ and $\boldsymbol{k}_3$ are 20.7 and 27.1 respectively. This indicates that the oscillations are more robust to variations of the parameters values for $\boldsymbol{k}_3$ than for the parameters given in the original model of Tyson.

6 Related Work

Probabilistic temporal logics and probabilistic model checking have been used in systems biology [33], e.g. for an analysis of a probabilistic model of the MAP kinase signaling cascade. However these techniques provide information on the probability that a given property is exactly satisfied. They thus provide no quantitative information on unsatisfied formulae and cannot be compared to the satisfaction degree presented in this paper.

More closely related to our continuous satisfaction degree are the linear metrics for quantitative transition systems defined in [34]. These metrics apply to traces and can be characterized by quantitative LTL formulae. LTL formulae are interpreted on the $[0, 1]$ interval. However, no implementation is proposed, and the applicability of this approach to solving optimization and robustness problems is not discussed.

To the best of our knowledge, the most closely related approach is the one proposed by Fainekos and Pappas [35], where a satisfaction degree for temporal logic specifications is defined. Although the two approaches share many similarities, a significant difference is that in [35] the satisfaction degree corresponds to a distance between a trace and the set of traces satisfying a formula, whereas in our case the violation degree corresponds to a distance between a formula and the set of formulae satisfied by the trace. An advantage of the satisfaction degree is that it offers a rather intuitive interpretation, since it corresponds to the minimal perturbation of the trace that can change the truth value of the specification. However, the dimension of the space of traces is in general considerably higher than the dimension of the space of formulae. In the first case, traces are represented in a space of dimension $X^{|\tau|}$ where X the state space and $|\tau|$ the lenght of the trace. In the second case, formulae are represented in a space whose dimension equals the number of variables appearing in the QFLTL formula, typically corresponding to the number of numerical constants appearing in the original LTL formula. Because the computation of satisfaction or violation degree involves set operations, the dimensionality of the corresponding spaces may strongly affect the practical applicability of these methods. Note however that these approaches, handling sets of traces [36, 37], and our approach, handling sets of formulae, are *a priori* compatible, and that their combination might combine their benefits.

Concerning robustness, in [38], Chaves and colleagues propose a quantitative measure of robustness corresponding to the volume of the set of valid parameters in the parameter space. This measure thus reflects the proportion of parameters that satisfy exactly the property, as opposed to our measure that represents how close to satisfying the property the system is for various parameters. These two measures provide complementary information on robustness. In [15], robustness is similarly defined with respect to temporal logic specifications. However, it has a Boolean interpretation, since a property is defined as robustly satisfied by an ODE system if it is satisfied by the system for all possible perturbations. As stated earlier, obtaining a quantitative measure of robustness is more informative for many practical problems.

7 Conclusion

We have defined a continuous measure of satisfaction of an LTL($\mathbb{R}$) formula in a numerical trace and shown that it can be computed using the QFLTL($\mathbb{R}$) constraint solving algorithm of [14]. This measure is more informative that the Boolean interpretation of the formulae and can be used in many situations in systems biology to reason about numerical traces.

This measure can be used as a fitness function in state-of-the-art optimization tools to efficiently guide the search of kinetic parameter values in biochemical reaction models in order to satisfy a set of properties formalized in LTL($\mathbb{R}$).

It can similarly be used to estimate the robustness of a model w.r.t. temporal logic specifications, in accordance to Kitano's notion of robustness for systems biology.

The generalization of model-checking to temporal logic constraint solving which is at the basis of the computation of this satisfaction measure thus seems to open new research avenues for the use of temporal logics in systems biology.

Acknowledgements

This work is partly supported by EU FP7 STREP project TEMPO and by the INRA Agrobi project INSIGHT.

References

1. Emerson, E.: Temporal and modal logic. In van Leeuwen, J., ed.: Handbook of Theoretical Computer Science. Volume B: Formal Models and Sematics. MIT Press (1990) 995–1072
2. Clarke, E.M., Grumberg, O., Peled, D.A.: Model Checking. MIT Press (1999)
3. Eker, S., Knapp, M., Laderoute, K., Lincoln, P., Meseguer, J., Sönmez, M.K.: Pathway logic: Symbolic analysis of biological signaling. In: Proceedings of the seventh Pacific Symposium on Biocomputing. (2002) 400–412
4. Chabrier, N., Fages, F.: Symbolic model checking of biochemical networks. In Priami, C., ed.: CMSB'03: Proceedings of the first workshop on Computational Methods in Systems Biology. Volume 2602 of Lecture Notes in Computer Science., Rovereto, Italy, Springer-Verlag (2003) 149–162
5. Chabrier-Rivier, N., Chiaverini, M., Danos, V., Fages, F., Schächter, V.: Modeling and querying biochemical interaction networks. Theoretical Computer Science **325** (2004) 25–44
6. Bernot, G., Comet, J.P., Richard, A., Guespin, J.: A fruitful application of formal methods to biological regulatory networks: Extending thomas' asynchronous logical approach with temporal logic. Journal of Theoretical Biology **229** (2004) 339–347
7. Calder, M., Vyshemirsky, V., Gilbert, D., Orton, R.: Analysis of signalling pathways using the continuous time markow chains. In Plotkin, G., ed.: Transactions on Computational Systems Biology VI. Volume 4220 of Lecture Notes in BioInformatics. Springer-Verlag (2006) 44–67 CMSB'05 Special Issue.
8. Heath, J., Kwiatkowska, M., Norman, G., Parker, D., Tymchyshyn, O.: Probabilistic model checking of complex biological pathways. In: Proc. Computational Methods in Systems Biology (CMSB'06). Volume 4210 of Lecture Notes in Computer Science., Springer-Verlag (2006) 32–47
9. Calzone, L., Chabrier-Rivier, N., Fages, F., Soliman, S.: Machine learning biochemical networks from temporal logic properties. In Plotkin, G., ed.: Transactions on Computational Systems Biology VI. Volume 4220 of Lecture Notes in BioInformatics. Springer-Verlag (2006) 68–94 CMSB'05 Special Issue.
10. Antoniotti, M., Policriti, A., Ugel, N., Mishra, B.: Model building and model checking for biochemical processes. Cell Biochemistry and Biophysics **38** (2003) 271–286
11. Batt, G., Ropers, D., de Jong, H., Geiselmann, J., Mateescu, R., Page, M., Schneider, D.: Validation of qualitative models of genetic regulatory networks by model checking : Analysis of the nutritional stress response in *Escherichia coli.* Bioinformatics **21** (2005) i19–i28

12. Fages, F.: Temporal logic constraints in the biochemical abstract machine BIOCHAM (invited talk). In Springer-Verlag, ed.: Proceedings of Logic Based Program Synthesis and Transformation, LOPSTR'05. Number 3901 in Lecture Notes in Computer Science, London, UK (2005)
13. Kohn, K.W.: Molecular interaction map of the mammalian cell cycle control and DNA repair systems. Molecular Biology of the Cell **10** (1999) 2703–2734
14. Fages, F., Rizk, A.: On temporal logic constraint solving for the analysis of numerical data time series. Theoretical Computer Science. CMSB'07 special isssue (2008)
15. Batt, G., Yordanov, B., Weiss, R., Belta, C.: Robustness analysis and tuning of synthetic gene networks. Bioinformatics **23** (2007) 2415–2422
16. Segel, L.A.: Modeling dynamic phenomena in molecular and cellular biology. Cambridge University Press (1984)
17. Szallasi, Z., Stelling, J., Periwal, V., eds.: System Modeling in Cellular Biology: From Concepts to Nuts and Bolts. MIT Press (2006)
18. Hucka, M., et al.: The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. Bioinformatics **19** (2003) 524–531
19. Fages, F., Soliman, S., Chabrier-Rivier, N.: Modelling and querying interaction networks in the biochemical abstract machine BIOCHAM. Journal of Biological Physics and Chemistry **4** (2004) 64–73
20. Calzone, L., Fages, F., Soliman, S.: BIOCHAM: An environment for modeling biological systems and formalizing experimental knowledge. BioInformatics **22** (2006) 1805–1807
21. Gilbert, D., Heiner, M., Lehrack, S.: A unifying framework for modelling and analysing biochemical pathways using petri nets. In: CMSB'07: Proceedings of the fifth international conference on Computational Methods in Systems Biology. Volume 4695 of Lecture Notes in Computer Science., Springer-Verlag (2007)
22. Matsuno, H., Doi, A., Nagasaki, M., Miyano, S.: Hybrid petri net representation of gene regulatory network. In: Proceedings of the 5th Pacific Symposium on Biocomputing. (2000) 338–349
23. 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
24. Phillips, A., Cardelli, L.: A correct abstract machine for the stochastic pi-calculus. Transactions on Computational Systems Biology (to appear) Special issue of Bio-Concur 2004.
25. Nickovic, D., Maler, O.: Amt: a property-based monitoring tool for analog systems. In: Proceedings of 5th International Conference on Formal Modelling and Analysis of Times Systems, FORMATS'07. Lecture Notes in Computer Science, Salzburg, Austria, Springer-Verlag (2007)
26. Kitano, H.: Towards a theory of biological robustness. Molecular Systems Biology **3** (2007) 137
27. Hansen, N., Ostermeier, A.: Completely derandomized self-adaptation in evolution strategies. Evolutionary Computation **9** (2001) 159–195
28. Chen, K.C., Csikász-Nagy, A., Györffy, B., Val, J., Novàk, B., Tyson, J.J.: Kinetic analysis of a molecular model of the budding yeast cell cycle. Molecular Biology of the Cell **11** (2000) 396–391
29. Levchenko, A., Bruck, J., Sternberg, P.W.: Scaffold proteins may biphasically affect the levels of mitogen-activated protein kinase signaling and reduce its threshold properties. PNAS **97** (2000) 5818–5823

30. Qiao, L., Nachbar, R.B., Kevrekidis, I.G., Shvartsman, S.Y.: Bistability and oscillations in the huang-ferrell model of mapk signaling. PLoS Computational Biology **3** (2007) 1819–1826
31. Fages, F., Soliman, S.: From reaction models to influence graphs and back: a theorem. In: Proceedings of Formal Methods in Systems Biology FMSB'08. Number 5054 in Lecture Notes in Computer Science, Springer-Verlag (2008)
32. Ventura, A.C., Sepulchre, J.A., Merajver, S.D.: A hidden feedback in signaling cascades is revealed. PLoS Computational Biology **to appear** (2008)
33. Kwiatkowska, M., Norman, G., Parker, D.: Using probabilistic model checking in systems biology. SIGMETRICS Performance Evaluation Review **35** (2008) 14–21
34. de Alfaro, L., Faella, M., Stoelinga, M.: Linear and branching metrics for quantitative transition systems. In Diaz, J., Karhumäki, J., Lepistö, A., Sannella, D., eds.: Proceedings of the 31st International Colloquium on Automata, Languages and Programming (ICALP'04). Volume 3142 of Lecture Notes in Computer Science. Springer (2004) 97–109
35. Fainekos, G., Pappas, G.: Robustness of temporal logic specifications. In Havelund, K., Núñez, M., Rosu, G., Wolff, B., eds.: Proceedings of the International Workshop on Formal Approaches to Software Testing and Runtime Verification, FATES/RV'06. Volume 4262 of Lecture Notes in Computer Science., Springer (2006) 178–192
36. Fainekos, G., Pappas, G.: Robust sampling for MITL specifications. In Raskin, J.F., Thiagarajan, P., eds.: Proceedings of the Fifth International Conference on Formal Modeling and Analysis of Timed Systems, FORMATS'07. Volume 4763 of Lecture Notes in Computer Science., Springer (2007) 147–162
37. Dang, T., Donze, A., Maler, O., Shalev, N.: Sensitive state space exploration. submitted, available at http://www-verimag.imag.fr/ maler/ (2008)
38. Chaves, M., Sontag, E., Sengupta, A.: Shape, size and robustness: feasible regions in the parameter space of biochemical networks. submitted, available as arXiv:0710.4269v1 (2008)