Automated Property Synthesis of ODEs Based Bio-pathways Models

Jun Zhou[1], R.Ramanathan[1], Weng-Fai Wong[1], and P.S.Thiagarajan[2]

[1] Department of Computer Science, National University of Singapore,
{zhoujun, ramanathan, wongwf}@comp.nus.edu.sg
[2] Laboratory of Systems Pharmacology, Harvard Medical School, Boston, USA
thiagu@hms.harvard.edu

Abstract. Identifying non-trivial requirements for large complex dynamical systems is a challenging but fruitful task. Once identified such requirements can be used to validate updated versions of the system and verify functionally similar systems. Here we present a technique for discovering behavioural properties of bio-pathway models whose dynamics is modelled as a system of ordinary differential equations (ODEs). These models are usually accompanied at best by high level functional requirements while undergoing many revisions as new experimental data becomes available. In this setting we first specify a set of property templates using bounded linear-time temporal logic (BLTL). A template will have the skeletal structure of a BLTL formula but the time bounds associated with the temporal operator as well as the value bounds associated with the system variables encoded as atomic propositions will be unknown parameters. We classify a given model's behavior as corresponding to one of these templates using a convolutional neural network. We then synthesize a concrete property from this template by estimating its parameters via a standard search procedure combined with statistical model checking (SMC). We have synthesized and validated properties of a number of pathway models of varying complexity using our method.

Keywords: property synthesis, statistical model checking, bounded linear-time temporal logic, ODEs models of bio-pathways

1 Introduction

Synthesizing specifications of system models is a useful but challenging task. This is especially so for bio-pathway models. These models are rarely come with concrete temporal specifications. Instead, they are accompanied by functionalities such as "EGF-NGF stimulation of PC12 cells discriminates between proliferation and differentiation". Synthesizing more concrete temporal specifications from these models is appealing for at least two reasons. First, the synthesized specifications can point to mechanistic chains of events that determine the overall functionality such as "transient activation of Erk1/2 leads to proliferation while its sustained activation results in differentiation". (We hasten to add that

this is merely an illustration using the functional specification and the concrete mechanistic property presented in [5]). Second, the construction of a model is rarely complete. Instead, it is repeatedly updated as fresh experimental data becomes available. In such settings, the temporal specifications synthesized from a previous version of the model can be used to assess whether the new model is qualitatively different from the older one.

As is well known there are two major classes of models to describe the dynamics of bio-pathways, namely deterministic ones based on ODEs [2] and stochastic ones [13] based on continuous-time Markov chains (CTMCs). In this paper, we shall focus on ODEs based models. In both types of models many rate constants of the reactions as well as the initial concentrations will be unknown. Here we consider this to be an important but orthogonal issue. Hence for evaluating our proposed method, we consider curated models with known parameter values taken from the Biomodels database [18].

We first build a set of property templates that capture parametrized families of pathway dynamical properties. Each template is built out of a BLTL (bounded linear time temporal logic) formula but whose time bounds associated with the temporal operators are integer-valued parameters. Furthermore, the atomic propositions appearing in the formula will be of the form $(\ell \leq x \leq u)$ where x is a system variable and ℓ and u are parameters that take values from the value domain of x. (These template parameters are not to be confused with the (model) parameters associated with the ODEs model). The choice of BLTL as the specification logic -and the accompanying atomic propositions- is guided by the nature of the experimental data that is usually available for our models of interest, namely signaling pathways. Here the experimental data (ie. observations of the system states) will typically consist of finite precision and noisy measurements regarding a small subset of system variables at a finite number of discrete time points. Further, only qualitative temporal properties will be applicable.

To focus on the main issues, we restrict our attention to four templates that capture key behavioural patterns of interest such as: "the concentration of a species x starts from an initial level in the interval $[c_1, c_2]$, rises to a level $[d_1, d_2]$ within k time units and remains in this interval until t_{max}". Based on these templates we develop a synthesis framework as shown in Figure 1. First, by assuming an initial probability (usually uniform) distribution over the initial states of the system variables, a set of trajectories is generated through numerical simulations. Next, the trajectories are presented to a pre-trained convolutional neural network to identify the template ψ that best corresponds to the trajectories. We then employ a simulated-annealing [21] based global optimization procedure to estimate the parameters of the template. Specifically, in each step of the procedure, the *value generator* instantiates from ψ a concrete property $\widehat{\psi}$. We then use statistical model checking to evaluate the quality of satisfaction of $\widehat{\psi}$. Subsequently, the *loss function* computes the loss, and reports it to the simulated-annealing procedure which then terminates, or generates a new set of values for the parameters.

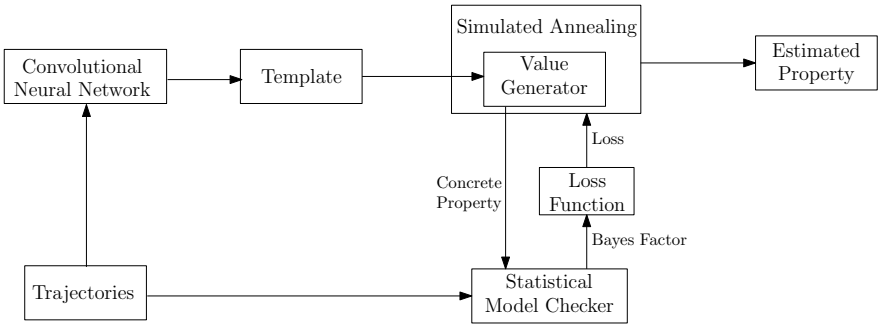


Fig. 1: Overview of the property synthesis framework

1.1 Related Work

Learning temporal logic formulas from data (or a generative model) is becoming a well explored field. The applications come from both cyber physical systems [8, 15, 17] and biological domains [3, 7, 12]. In the latter domain —which is our interest— the line of work reported in [7] is particularly relevant. The authors first learn a stochastic hybrid system from data and then use the model to generate data for learning the temporal logic formulas of interest in two steps. First, using an evolutionary algorithm, the structure (template as we call it) of the formula is learned. Then the parameters in the template are calibrated using a previously developed stochastic optimization method called the Gaussian Process Upper Confidence Bound (GP-UCB) algorithm [4]. The specification logic is bounded metric temporal logic. In our setting the model is available as a system of ODEs. We fix a set of templates in advance and train a convolution network using synthetic data not generated by the model in order to avoid bias. Then using this network and trajectories randomly generated by the model we match a template to the model. We then learn its parameters using simulated annealing combined with statistical model checking. Finally we use BLTL as our specification logic since it is a good fit for the class of models we wish to study.

An important aspect of parameter learning is to determine how well the formula instantiated by a particular choice of parameters matches the training data. Here again there is a good deal of literature on "robustness of satisfaction" [3, 9, 10, 28]. Specifically, [28] is illustrative for ODEs based models in which a continuous notion of satisfaction is combined with an evolutionary search procedure to estimate kinetic parameters meeting temporal logic specifications. On the other hand the work reported in [3] formulates a robustness of satisfaction notion for stochastic systems and then uses this notion to optimize chosen control parameters of a stochastic system in order to maximize the robustness of satisfaction.

The paper is organized as follows. Section 2 presents the preliminaries and property templates. In Sections 3 and 4, we explain our search procedure. Section 5 presents the experimental results and we conclude in Section 6.

2 Preliminaries

We introduce the basic notations we will be using in connection with ODEs, BLTL, statistical model checking. We conclude with the introduction of property templates.

2.1 Trajectories of a System of ODEs

Suppose there are n molecular species $\{x_1, x_2, \ldots, x_n\}$ in the pathway. For each species x_i, an equation of the form $\frac{dx_i}{dt} = f_i(\mathbf{x}, \Theta_i)$ describes the kinetics of the reactions that produce and consume x_i where $\mathbf{x}$ is the concentrations of the molecular species taking part in the reactions. Θ_i consists of the rate constants governing the reactions. Each x_i is a real-valued function of $t \in \mathbb{R}_+$, the set of non-negative reals. We shall realistically assume that $x_i(t)$ takes values in the interval $[L_i, U_i]$, where L_i and U_i are non-negative rationals with $L_i < U_i$. Assuming there are m reactions, we let $\Theta = \{\theta_1, \theta_2, \ldots, \theta_m\}$ be the set of rate constants. We define for each variable x_i an interval $[L_i^{init}, U_i^{init}]$ with $L_i \leq L_i^{init} < U_i^{init} \leq U_i$. We assume the value of the initial concentration of x_i to fall in this interval. We also assume the nominal value of the rate constant θ_j falls in the interval $[L_j^{init}, U_j^{init}]$ for $1 \leq j \leq m$. We set $INIT = (\prod_i [L_i^{init}, U_i^{init}]) \times (\prod_j [L_j^{init}, U_j^{init}])$. Here $INIT$ is meant to capture the variability in the initial concentrations of the variables and the rate constants across a population of cells. Further, we let $\mathbf{v}$ to range over $\prod_i [L_i^{init}, U_i^{init}]$ and $\mathbf{w}$ to range over $\prod_j [L_j^{init}, U_j^{init}]$. We define in the usual way the notion of a trajectory $\sigma_{\mathbf{v},\mathbf{w}}$ starting from $(\mathbf{v}, \mathbf{w}) \in INIT$ at time 0. We let TRJ denote the set of all finite trajectories that start in $INIT$.

As mentioned earlier we assume a probability distribution over $INIT$ and for convenience assume it to be the uniform one. The ODEs systems arising in our setting will induce vector fields that satisfy a natural continuity property. Hence one can define the probability that a trajectory starting from a randomly chosen state in $INIT$ will satisfy a given BLTL formula. Consequently one can develop a statistical model checking procedure to verify whether the system of ODEs meets the given BLTL specification with required probability [27].

2.2 Bounded Linear-time Temporal Logic

An atomic proposition for our setting will be of the form $(L \leq x_i \leq U)$ with $L_i \leq L < U \leq U_i$ where L, U are rationals. The proposition $(L \leq x_i \leq U)$ says "the current concentration level of x_i lies in the interval $[L, U]$" and we fix a finite set of atomic propositions. BLTL formulas are then defined in the usual way.

We fix a finite set of time points $T = \{t_0 < t_1, \ldots, t_K\}$ and interpret a BLTL formulas over a trajectory σ in TRJ observed at the time points in T as usual. We say that σ is a *model* of ψ if $\sigma, t_0 \models \psi$.

For a formula ψ the statement $P_{\geq r}(\psi)$ where r $\in$ [0,1) will mean "the probability that a trajectory in TRJ is a *model* of ψ is at least r". To verify this,

we consider the sequential hypothesis testing problem where the null hypothesis is $\mathcal{H}_0 : P_{\geq r}(\psi)$ and the alternate hypothesis $\mathcal{H}_1 : P_{<r}(\psi)$. A convenient termination criterion here is is the Bayes factor [16, 19].

$$\mathcal{B} = \frac{Pr(d|\mathcal{H}_0)}{Pr(d|\mathcal{H}_1)} \tag{1}$$

where d is the collection of Bernoulli random variables denoting the outcome whether a random trajectory generated by the ODE system satisfies ψ. Comparing $\mathcal{B}$ against a pre-defined threshold h, the property is accepted if $\mathcal{B}$ is larger than h and is rejected if it is less than $1/h$. Unlike the SPRT ratio test one doesn't have to specify an indifference region.

2.3 Templates

A template is a BLTL formula in which the bounds on system variable values in the atomic propositions and the integer bounds associated with the temporal operators are replaced by symbolic variables. These variables will be called *propositional variables* and *temporal variables* respectively in what follows. In addition, the template is augmented by a set of constraints. These constraints will be of the form $[u_j \leq \ell_k]$ or $[u_k \leq \ell_j]$ given two atomic propositions of the form $(\ell_j \leq x_j \leq u_j)$ and $(\ell_k \leq x_k \leq u_k)$.

Here is an example of a template:

$$p_1 \wedge F^{\leq t_1} G^{\leq t_2} p_2 \mid [u_1 \leq \ell_2]$$
$$\text{where } p_1 = (\ell_1 \leq x_1 \leq u_1) \text{ and } p_2 = (\ell_2 \leq x_1 \leq u_2).$$

This template represents the statement "value of x_1, starting from a low level (p_1) reaches within t_1 time units a high level (p_2) and stays at p_2 for at least t_2 units". $[u_1 \leq \ell_2]$ captures the constraint the level (ℓ_1, u_1) is lower than (ℓ_2, u_2).

The main idea is to search over the temporal and atomic proposition variables and use Bayes factor to measure of how well a synthesized property characterizes the observed behavior. A property with a Bayes factor larger than a given Bayes factor threshold is accepted while one with a small Bayes factor is rejected. In this initial study we consider the templates listed in Table 1.

3 Classifying Templates using a Convolutional Neural Network

Our workflow first trains a convolution network to recognize trajectories presented to it as belonging to one of the set of templates we have fixed. It then classifies a set of random trajectories generated by a model as belonging to one of the templates and then proceeds to synthesize a concrete property using the template.

No.	Template	Description
1	$p_1 \wedge F^{\leq t_1} G^{\leq t_2} p_2$ where $p_1 : (\ell_1 \leq x \leq u_1)$ $p_2 : (\ell_2 \leq x \leq u_2)$	Starting from the level p_1, within t_1 steps, the value of x reaches the level p_2 and stays there for at least t_2 steps. Typically describes sustained activations or deactivations. Constraints can be used to specify whether x decreases or increases from the initial level.
2	$p_1 \wedge F^{\leq t_1}(p_2 \wedge F^{\leq t_2} p_3)$ where $p_1 : (\ell_1 \leq x \leq u_1)$ $p_2 : (\ell_2 \leq x \leq u_2)$ $p_3 : (\ell_3 \leq x \leq u_3)$	Starting from an initial level p_1, the value of x reaches the level p_2 within t_1 steps. Then, from p_2, x reaches a level p_3 within t_2 steps. Formulates evolution of species concentration from an initial level to a new level and then further to another new level or back to the initial level.
3	$p_1 \wedge F^{\leq t_1}(p_2 \wedge F^{\leq t_2} G^{\leq t_3} p_3)$ where $p_1 : (\ell_1 \leq x \leq u_1)$ $p_2 : (\ell_2 \leq x \leq u_2)$ $p_3 : (\ell_3 \leq x \leq u_3)$	Similar to Template 2, the value of x starts from the level p_1, reaches the level p_2 within t_1 steps. Then within the next t_2 steps, reaches a level p_3 and stays in p_3 for at least t_3 steps. Characterizes transient or sustained activations, can be extended to formulate bistability.
4	$p_1 \wedge F^{\leq t_1}(p_2 \wedge F^{\leq t_2}(p_3 \wedge F^{\leq t_1}(p_4)))$ where $p_1 : (\ell_1 \leq x \leq u_1)$ $p_2 : (\ell_2 \leq x \leq u_2)$ $p_3 : (\ell_3 \leq x \leq u_3)$ $p_4 : (\ell_4 \leq x \leq u_4)$	Starting from an initial level p_1, the value of x reaches the level p_2 where $(u_1 < \ell_2)$ within t_1 steps. Then, from p_2, x reaches a level p_3, $(u_3 < \ell_2)$ within t_2 steps. Further from p_3, x reaches a level p_4 where $(u_3 < \ell_4)$. Imposing constraints $[u_1 < \ell_2] \wedge [u_1 < \ell_4] \wedge [u_3 < \ell_2] \wedge [u_3 < \ell_4]$ characterizes oscillations.

Table 1: Basic Templates

3.1 Data Preprocessing

The evolution of a variable x is mainly reflected by changes in its value over time. We first transform the trajectories by evaluating the change in x at each time point, and computing the normalized $\Delta x(t)$ data over time as indicated by the formula below. This transformed data is then fed to the convolutional neural network for classification.

$$\Delta x(t) = \frac{x(t) - x(t-1)}{max(x) - min(x)}, \tag{2}$$

where $max(x)$ and $min(x)$ are the maximum and minimum values of x across all the time points in the simulation.

3.2 Training and Deploying the Convolutional Neural Network

A convolutional neural network (CNN) is a type of feed forward neural network proposed in [24]. It has been successfully used to classify time series data and other features [30]. In this paper, we have adopted a standard convolutional

neural network and implemented it using Tensorflow, a deep learning framework by Google [1]. There is a vast literature available including [24] on CNNs.

The CNN receives the pre-processed inputs described in Section 3.1 and feeds it to two convolutional and pooling layers, connected to two fully connected layers. Then it outputs to four output neurons, corresponding to the four templates. Due to space limitations we present the architecture and other details of the this CNN in the full report [31].

Our CNN is trained for the templates listed in Table 1. The training set is generated from mathematical functions found in [29]. Specifically, we selected 25 functions that conform to the four templates. For each of these functions, we generated 68 'seed' curves using different random initial parameters. Next, we transformed these into the frequency domain using Fast Fourier Transform (FFT). In the frequency domain, we perform further randomization before transforming them back into curves in time domain using the inverse FFT. We obtained $2,000$ randomized curves from each seed curve. In total, $136,000$ curves were used to train the CNN.

After training, the CNN is deployed to identify a template that best matches a set of trajectories randomly generated by a a given model. Since neural networks take as inputs fixed-length data, the trajectories need to be re-scaled using a different sampling rate of simulation as follows. We first generate 20 trajectories. The same simulation time as given in the literature for the respective model is divided up into 200 equally spaced time-points, and sampled. The trajectories are then transformed into $\Delta x(t)$ as mentioned before in Section 3.1, and fed to the neural network. A simple majority across the results of classifying these 20 trajectories is used to determine the final template.

4 The Search Procedure

Given a template ψ identified from the convolutional neural network with time variables Var_T, and propositional variables Var_{AP}, we automatically mine the values of Var_T and Var_{AP} such that the concretized formula is optimal in a certain sense.

In order to reduce the search complexity, we assume the BLTL based template is given as a conjunction of component formula skeletons. We consequently optimize each conjunct in the template.

We adopt a simulated annealing based procedure presented in Algorithm 1 to estimate the parameters.

We generate values for the propositional variables using the constraints specified in the propositional variables and the template constraints. Though the constraint satisfaction problem is NP-complete the constraints in our framework are simple inequalities which enables us to adopt a tree-based solution. The value intervals of a variable are parsed as a tree structure where the values of the child nodes are larger than the parent nodes.

For example, for the template $p_1 \wedge F^{\leq t_1}(p_2 \wedge F^{\leq t_2} p_3)$ suppose we have the constraints $[u_1 < \ell_2]$ and $[\ell_2 < u_3]$, together with the implicit constraints $[\ell_1 <$

Algorithm 1: optimizeProperty

Input : Template ψ
Output: Synthesized property ψ_{syn}

$\widehat{\psi} \leftarrow$ Initialize Var_T and Var_{AP} using random values;
while ***Simulated Annealing*** *decides to continue* **do**
  Compute Bayes Factor $\mathcal{B}_{\widehat{\psi}} \leftarrow$ **SMC**$(\widehat{\psi})$;
  Compute $Loss_{\widehat{\psi}} \leftarrow$ **Loss Function**$(\widehat{\psi}, \mathcal{B}_{\widehat{\psi}})$;
  Simulated Annealing $\leftarrow Loss_{\widehat{\psi}}$;
  Update Var_T and Var_{AP} ;
return $\psi_{syn} \leftarrow \widehat{\psi}$ with minimum loss if exists;

u_1], $[\ell_2 < u_2]$ and $[\ell_3 < u_3]$ the tree is constructed as shown in Figure 2. We generate values for the leaf variables(u_2 and u_3) first and then use them to bound the value range of parents(ℓ_2 and ℓ_3), recursively till the root (ℓ_1) is reached.

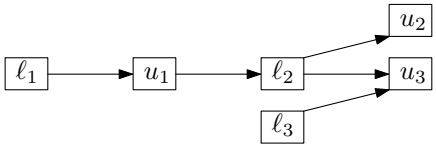


Fig. 2: Generating values for propositional variables using a tree

4.1 Loss Function

Each instantiated property $\widehat{\psi}$ is scored using a *loss function* and the score will guide the direction of the search. The score is composed out of the "loss" suffered by three factors: temporal variables, atomic propositions and the quality of satisfaction. For the temporal variables we use the intuition that if ψ_1 and ψ_2 are two instantiations such that ψ_1 implies ψ_2 then ψ_1 is to be preferred. This suggests that if $\psi_1 = F^{\leq t_1}\varphi$ and $\psi_2 = F^{\leq t_2}\varphi$ are two instantiations and $t_1 \leq t_2$ then t_1 is preferred to t_2. Similarly t_2 is preferred to t_1 if $G^{\leq t_1}\varphi$ and $G^{\leq t_2}\varphi$ are two instantiations with $t_1 \leq t_2$.

The loss component L_T of the temporal variables is given by:

$$L_T = \prod_{t_i \in \text{Var}_T} \left(t_i\right)^{sgn(t_i)}$$

$$sgn(t) = \begin{cases} -1, & \text{if temporal operator of } t_i \text{ is } G \text{ or } U \\ 1, & \text{if temporal operator of } t_i \text{ is } F \end{cases}.$$

Next, we define L_{AP}, the loss function component contributed by the propositional variables. For each atomic proposition, we consider both the tightness of the value range, and how precisely it describes the behaviour of the trajectories.

For each atomic proposition ap_i, we define the tightness as $(u_i - \ell_i)/(max_i - min_i)$, the range normalized to the maximum value range of the variable in trajectories. The idea is to keep the value range as small as possible.

Besides the tightness, we also measure the fitness of the atomic propositions in $\widehat{\psi}$ to the trajectories based on the constraints. Essentially for each constraint of the form $u_j < \ell_k$ attached to the atomic propositions ap_j and ap_k, the estimated levels of ap_j is expected to be lower than ap_k. This information is also used to optimize $\widehat{\psi}$. To this end, we compute the mean value of ap_i as $(\frac{\ell_i+u_i}{2})$. The weight w_i associated with each ap_i is evaluated as follows. We first initialize the set of weights W_{AP} for all the atomic propositions in $\widehat{\psi}$ to 0. Then for each constraint $u_j < \ell_k$, we decrease w_j by 1 and increase w_k by 1. The fitness of an atomic proposition is thus $\left(\frac{\ell_i+u_i}{2}\right)^{w_i}$. Intuitively, the level of ap_j tends to be in the lower range of value space while ap_k to be in higher range.

Combining these two factors, we define the loss function component due to the propositional variables as

$$L_{AP} = \prod_{ap_i \in \text{Var}_{AP}} \left(\Big(\frac{u_i - \ell_i}{max_i - min_i} \Big) \Big(\frac{\ell_i + u_i}{2 \cdot max_i} \Big)^{w_i} \right).$$

Finally, in each iteration of the simulated-annealing procedure, if the Bayes factor $\mathcal{B}_{\widehat{\psi}}$ is larger than a pre-defined threshold h (in our case it is set to 100), we apply the loss function and continue with the iterations according to the search procedure. Otherwise, the loss is set as ∞ and the current combination of parameters is rejected. The search then continues with another combination of parameters.

Thus

$$Loss_{\widehat{\psi}} = \begin{cases} L_{AP} \cdot L_T, & \mathcal{B}_{\widehat{\psi}} > h \\ \infty, & \text{otherwise} \end{cases}.$$

We use the multiplicative form of the loss function since we found that the additive form performs badly. For instance, if two temporal variables and one propositional variable appear in a formula the search gets biased towards optimizing just one the three variables while fixing a trivial value for the other two variables. Admittedly the current formulation of the loss function is just a first and preliminary step. A systematic study of the various possibilities -including other notions of quality of satisfaction- needs to be carried out in the future.

5 Experimental Evaluation

We applied our method to six bio-pathway models taken from the Biomodels database [23]. For the purposes of experimentation we fixed $\pm 5\%$ range around the nominal values as the initial interval of values of each species and we assumed a uniform distribution over the resulting set of initial states. Using the convolutional neural network and randomly generated trajectories using the model, the

most suitable BLTL template was then identified followed by a concrete instantiation for this template to a high satisfaction probability, namely, $r \geq 0.9$.

Table 2 shows $|\mathbf{x}|$, the number of system variables and $|\Theta|$, the number of rate constants of the ODEs systems associated with the six models. The time unit for the F and G operators is 'minutes'. Furthermore, the number of time points to simulate (i.e. t_K) for each of the models was fixed using the literature of the respective models [5, 6, 11, 14, 20, 25]. We next present the synthesized properties for the important species in each of the bio-pathway models. Across all the six case studies, there is a total of 13 such species.

Bio-pathway models	EGF-NGF	Segmentation Clock	MAPK Cascade	Atorvastatin	Va Factor	CD95 Signalling
$\|x\|$	32	16	8	18	30	23
$\|\Theta\|$	48	71	22	30	9	17

Table 2: Characteristics of the models

Validation In the six case studies we present here, we compared the synthesized properties against the observed qualitative trends of species documented in [5, 6, 11, 14, 20, 25]. For one of the models we provide further validation by using the synthesized properties in the context of rate constants estimation problem as explained in Section 5.3.

5.1 Template recognition

We first generate 20 trajectories from the model and use these as inputs to the CNN. For each trajectory and for each species (variable) of interest the CNN returns the confidence level in classifying the trajectory to each of the four templates and the template with the highest confidence is chosen. Finally, the template with most votes from all the trajectories is chosen as the template to be the candidate for synthesizing a concrete formula.

For each of the case studies in Section 5.2, we observed that the CNN returns the same template overwhelmingly for all the 20 trajectories with high confidence (above 98%). This data is reported in [31].

5.2 Case Studies

EGF-NGF Pathway The EGF-NGF signalling pathway [5] captures the differential response of PC12 cells to two growth hormones, EGF and NGF. EGF induces cell proliferation while NGF stimulates cell differentiation. It has been reported that the signal specificity is correlated with different Erk dynamics. A transient activation of Erk has been associated with cell proliferation, while a sustained activation has been linked to differentiation. The model has 32 ODEs and 48 kinetic rate parameters. We simulated this model for 60 minutes.

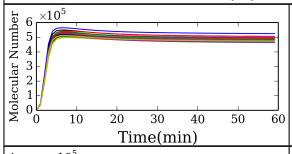
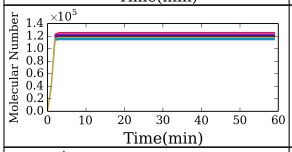
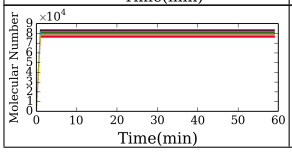
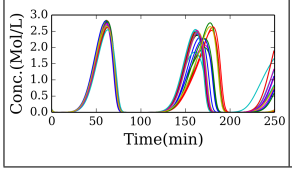
Table 3(a) shows three properties that describe the sustained activation of Erk*, bound-EGFR and C3G*, rising rapidly (within 10 minutes) to a high

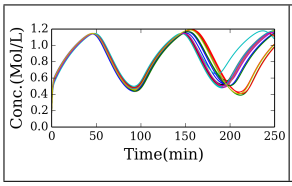
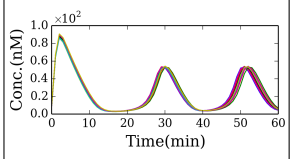
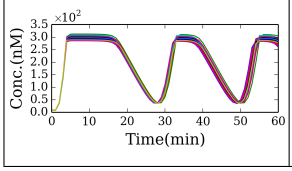
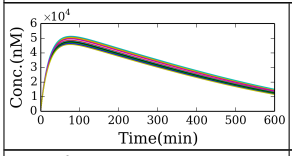
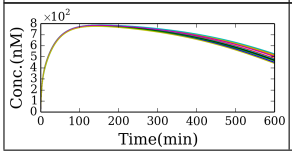
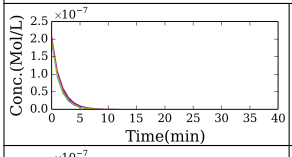
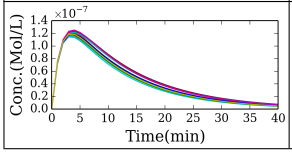
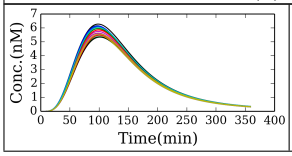
level. It has been verified from experimental data that under NGF stimulation, sustained activation of Erk* is induced by the phosphorylation of C3G. The synthesized property captures this behaviour: ($[0 \leq \text{Erk}^* \leq 0] \wedge F^{\leq 5} G^{\leq 55}([477401 \leq \text{Erk}^* \leq 571121])$) returned that the concentration level of Erk* rises from an initial level $[0 \leq \text{Erk}^* \leq 0]$ to a peak level $[477401 \leq \text{Erk}^* \leq 571121]$ and stays at that level for 50 minutes.

Segmentation Clock Network Formation of segments in vertebrate embryos is controlled by coupled oscillations in the Notch, Wnt and FGF signalling pathways governed by a segmentation clock network that periodically activates the segmentation genes [11]. The model consists of 16 ODEs and 71 kinetic rate parameters. We simulated this model for 250 minutes.

From Table 3(b), one can find that both properties characterize the oscillation of Lunatic fringe-mRNA and cytosolic NicD, capturing the peak values. Although the search space of 11 parameters is large, the mined properties are closed to the nominal ones from literature. For example, the Lunatic fringe-mRNA property is close to the one observed in [27]:

$(([\text{Lunatic fringe mRNA} \leq 0.4]) \wedge (F^{\leq 40}([\text{Lunatic fringe mRNA} \geq 2.2] \wedge F^{\leq 40}([\text{Lunatic fringe mRNA} \leq 0.4] \wedge F^{\leq 40}([\text{Lunatic fringe mRNA} \geq 2.2] \wedge F^{\leq 40}([\text{Lunatic fringe mRNA} \leq 0.4])))))$.

Simulation profile	Synthesized property
(a) EGF-NGF Pathway Model	
	$p_1 \wedge F^{\leq 5} G^{\leq 55} p_2$ $p_1 : 0 \leq \text{Erk}^* \leq 0$ $p_2 : 477401 \leq \text{Erk}^* \leq 571121$
	$p_1 \wedge F^{\leq 3} G^{\leq 57} p_2$ $p_1 : 0 \leq \text{C3G}^* \leq 0$ $p_2 : 111035 \leq \text{C3G}^* \leq 138166$
	$p_1 \wedge F^{\leq 2} G^{\leq 58} p_2$ $p_1 : 0 \leq \text{bound-EGFR} \leq 0$ $p_2 : 81639.9 \leq \text{bound-EGFR} \leq 86368.9$
(b) Segmentation Clock Network Model	
	$p_1 \wedge F^{\leq 58}(p_2 \wedge F^{\leq 23}(p_3 \wedge F^{\leq 75}(p_4)))$ $p_1 : 0.096 \leq \text{Lunatic fringe mRNA} \leq 0.102$ $p_2 : 2.42 \leq \text{Lunatic fringe mRNA} \leq 2.68$ $p_3 : 0.000 \leq \text{Lunatic fringe mRNA} \leq 0.008$ $p_4 : 1.83 \leq \text{Lunatic fringe mRNA} \leq 2.65$

	$p_1 \wedge F^{\leq 40}(p_2 \wedge F^{\leq 56}(p_3 \wedge F^{\leq 26}(p_4)))$ $p_1 : 0.199 \leq$ cytosolic NicD ≤ 0.207 $p_2 : 1.11 \leq$ cytosolic NicD ≤ 1.23 $p_3 : 0.25 \leq$ cytosolic NicD ≤ 0.46 $p_4 : 0.86 \leq$ cytosolic NicD ≤ 1.03
(c) MAPK Pathway Model	
	$p_1 \wedge F^{\leq 4}(p_2 \wedge F^{\leq 13}(p_3 \wedge F^{\leq 15}(p_4)))$ $p_1 : 9.50 \leq$ Mos-P ≤ 10.50 $p_2 : 81.38 \leq$ Mos-P ≤ 88.97 $p_3 : 0.00 \leq$ Mos-P ≤ 5.13 $p_4 : 44.24 \leq$ Mos-P ≤ 68.94
	$p_1 \wedge F^{\leq 6}(p_2 \wedge F^{\leq 24}(p_3 \wedge F^{\leq 11}(p_4)))$ $p_1 : 9.50 \leq$ Erk2-PP ≤ 10.50 $p_2 : 275.68 \leq$ Erk2-PP ≤ 328.35 $p_3 : 1.77 \leq$ Erk2-PP ≤ 40.22 $p_4 : 263.68 \leq$ Erk2-PP ≤ 299.09
(d) Atorvastatin Pharmacokinetics Model	
	$p_1 \wedge F^{\leq 160}(p_2 \wedge F^{\leq 434} p_3)$ $p_1 : 0 \leq \mathrm{AS}_c \leq 0$ $p_2 : 42419.6 \leq \mathrm{AS}_c \leq 45998.8$ $p_3 : 15109.7 \leq \mathrm{AS}_c \leq 15314.1$
	$p_1 \wedge F^{\leq 245}(p_2 \wedge F^{\leq 352} p_3)$ $p_1 : 0 \leq \mathrm{ASL}_c \leq 0$ $p_2 : 739.05 \leq \mathrm{ASL}_c \leq 773.13$ $p_3 : 520.33 \leq \mathrm{ASL}_c \leq 526.39$
(e) Va Factor Inactivation Model	
	$p_1 \wedge F^{\leq 7} G^{\leq 33} p_2$ $p_1 : 1.9 \times 10^{-7} \leq \mathrm{Va} \leq 2.1 \times 10^{-7}$ $p_2 : 0 \leq \mathrm{Va} \leq 6.72^{-9}$
	$p_1 \wedge F^{\leq 8}(p_2 \wedge F^{\leq 32} p_3)$ $p_1 : 0 \leq \mathrm{Va5} \leq 0$ $p_2 : 1.02 \times 10^{-7} \leq \mathrm{Va5} \leq 1.11 \times 10^{-7}$ $p_3 : 0 \leq \mathrm{Va5} \leq 7.96 \times 10^{-9}$
(f) CD95 Signalling Model	
	$p_1 \wedge F^{\leq 143}(p_2 \wedge F^{\leq 139} p_3)$ $p_1 : 0.00 \leq \mathrm{C8}^* \leq 0.00$ $p_2 : 4.53 \leq \mathrm{C8}^* \leq 4.63$ $p_3 : 0.789 \leq \mathrm{C8}^* \leq 0.832$

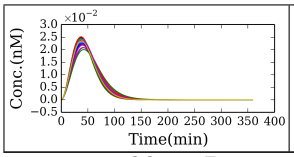
	$p_1 \wedge F^{\leq 27}(p_2 \wedge F^{\leq 132} G^{\leq 204} p_3)$ $p_1 : 0.000 \leq$ NF-κB-IκB-P ≤ 0.000 $p_2 : 0.021 \leq$ NF-κB-IκB-P ≤ 0.024 $p_3 : 0.000 \leq$ NF-κB-IκB-P ≤ 0.000

Table 3: Properties synthesized for the six case studies.

MAPK Cascade From yeast to mammals, mitogen activated protein kinase (MAPK) cascades are bio-molecular networks widely involved in signal transduction of extracellular stimulus from the plasma membrane to the cytoplasm and nucleus. They play a major role in processes involving cell growth, mitogenesis, differentiation and stress responses in mammalian cells. The MAPK pathway [20] consists of three levels where the activated kinase at each level phosphorylates the kinase at the subsequent level down the cascade. It has been shown that a negative feedback loop of MAPK cascade results in sustained oscillations in MAPK phosphorylation [20]. This ODEs model of this MAPK cascade consists of 8 species and 22 rate parameters. We simulated the model for 60 minutes.

Table 3(c) illustrates the properties for the two species, namely, phosphorylated Mos (Mos-P) at the initial level of the cascade, and biphosphorylated kinase Erk (Erk-PP) at the terminal level of the cascade. With the increased production of Erk-PP, the negative-feedback due to Erk-PP affects the phosphorylation of the initial level kinase, Mos. This in turn affects downstream phosphorylation of intermediate kinases, and ultimately the concentration of Erk-PP is decreased. Thus an oscillation cycle is triggered. The two properties synthesized by our method reflect this behaviour.

Experimental findings [22] indicate that "dual serine/threonine phosphorylation of SOS by Erk has been found to cooperatively inhibit MKKK phosphorylation". When the ODEs model is updated to reflect this change in the network, our method synthesized the following property:

$$[9.5 \leq \text{Erk2-PP} \leq 10.5] \wedge F^{\leq 17}([251.30 \leq \text{Erk2-PP} \leq 262.66]$$
$$\wedge F^{\leq 15}([0 \leq \text{Erk2-PP} \leq 42.74] \wedge (F^{\leq 14}[173.20 \leq \text{Erk2-PP} \leq 192.45]))).$$

From this synthesized property, one can infer that the amplitude of the oscillations has decreased compared to the nominal model presented in Table 3(c).

Atorvastatin pharmacokinetics Drug metabolism of statins inside the liver cells plays an important role in reducing cholesterol synthesis, and the stimulation of the uptake of LDL-cholesterol from the blood [6]. This ODEs model describes the pharmacokinetics of transport processes and metabolic enzymes in the biotransformation of atorvastatin. It consists of 18 ODEs and 30 rate parameters and the model was simulated for 600 minutes.

Table 3(d) shows two requirements synthesized for the atorvastatin pathway. AS (a hydrophilic hydroxyl-acid) and ASL (a very lipophilic lactone), the two

forms of atorvastatin are transported into the cell and converted into different metabolites. The properties: $[0 \leq \mathrm{AS}_c \leq 0] \wedge F^{\leq 160}([42419.6 \leq \mathrm{AS}_c \leq 45998.8] \wedge F^{\leq 434}([15109.7 \leq \mathrm{AS}_c \leq 15314.1]))$ and $[0 \leq \mathrm{ASL}_c \leq 0] \wedge F^{\leq 245}([739.05 \leq \mathrm{ASL}_c \leq 773.13] \wedge F^{\leq 352}([520.33 \leq \mathrm{ASL}_c \leq 526.39]))$ describe this behaviour. The estimated value bounds $[42419.6 \leq \mathrm{AS}_c \leq 45998.8]$ and $[739.05 \leq \mathrm{ASL}_c \leq 773.13]$ are close to the peak observed in the system. The subsequent fall in the concentration due to the conversion of AS_c and ASL_c to their corresponding para- and ortho-hydroxy metabolites is also captured accurately by the value bounds $[15109.7 \leq \mathrm{AS}_c \leq 15314.1]$ and $[520.33 \leq \mathrm{ASL}_c \leq 526.39]$.

Va factor pathway The regulation of Va factor plays a crucial role in hemostasis. As studied in [14], activated-protein-C (APC) causes inactivation of bovine factor Va and this model involves bond cleavage and dissociation of Va and its associated intermediate complexes produced in the process. The model consists of 30 ODEs and 9 kinetic rate parameters and was simulated for 20 minutes.

The two properties synthesized by our method characterizes the behaviour of the three species, namely Va and Va_5 are shown in Table 3(e). In particular, the properties synthesized using our method captures the rapid dissociation of Va by APC within 7 minutes.

CD-95 Signalling Activation of CD-95 [25] in some situations results in cell death, and, in some other situations, induces activation of the NF-κB pathway. This has been found to be due to the cleavage of an anti-apoptotic protein, $\mathrm{cFLIP}_\mathrm{L}$ and Procaspase-8. This model has 23 variables and 17 parameters and was simulated for 360 minutes.

The properties synthesized in Table 3(f) show the activation of Caspase-8 and the NF-κB-IκB-P by CD-95. Our method was able to mine the properties which characterize the rise and fall of the two proteins. More specifically, the third property mined for $NF\text{-}\kappa B\text{-}I\kappa B\text{-}P$ reflects transient activation within 150 minutes. It has been reported that CD-95 results in parallel – and not mutually-exclusive – transient activation of NF-κB and the Death Inducing Signalling Complex (DISC). This is in agreement with our findings.

5.3 Rate Constants Estimation Based on the Synthesized Properties

To further demonstrate the efficacy of the property synthesis procedure, we used the synthesized properties to estimate the unknown rate constants $\mathbf{w}$ of a pathway model in the context of the method developed in [27]. In this method both time course experimental data and known qualitative trends are encoded as BLTL formulas and the rate constant estimation problem is solved through evolutionary search combined with statistical model checking. In the present setting, we use the synthesized properties ψ_{syn} as sole inputs (ie. no experimental data) to this estimation procedure. We then compared the quality of the rate constants obtained using our synthesized properties with the rate constants reported in the literature [25].

We applied our method to the CD-95 signalling pathway. We assumed 10 (k_2, k_3, k_5, k_6, k_7, k_{11}, k_{12}, k_{14}, k_{15}, k_{17}) out of 17 rate constants to be unknown. The inputs to the estimation procedure of [27] consists of 7 BLTL properties synthesized by our method. Figure 3 shows the simulation profiles generated using the predicted rate constant values. More precisely, $1,000$ trajectories were generated using the rate constants estimated by our method and plotted against trends observed using the constants reported in [25].

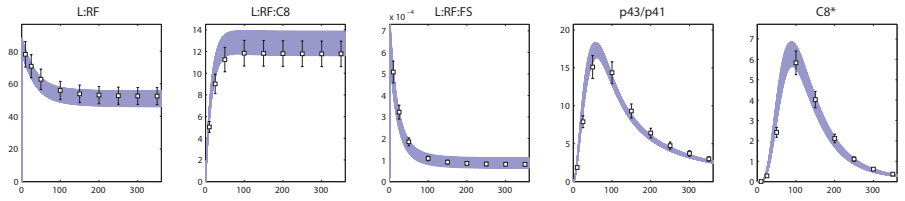


Fig. 3: Parameter estimation results for the CD-95 pathway using the synthesized properties

6 Conclusion

We have proposed an automated method to mine dynamic properties from ODEs based models of bio-pathways. Using simulated trajectories, our method first identifies a BLTL template matching their behaviour with the help of a convolutional neural network. A simulated-annealing based procedure combined with statistical model checking is then applied to this template to mine a concrete property. By checking the synthesized properties against the ones given in the literature as well as using them to do rate constants estimation of biopathways we have provided strong evidence that the mined BLTL formulas faithfully describe the behaviour of various species in our case studies.

In this preliminary study we have started with four templates. It will be useful to expand this templates library. Equally important, we have considered here only templates involving a single system variable. It will be challenging but very fruitful to learn properties that involve (at least) two system variables. This will enable for instance, to learn regulatory trends; for instance how an upstream variable representing a perturbation generates a pathway response in terms of a downstream variable.

Here we have focused on synthesizing properties for biological pathways modelled as a system of ODEs. However, our technique can be applied to ODEs systems arising in other settings as well.

To improve computational scalability, it will be important to port our current implementation to a GPU platform and exploit parallel search strategies such as parallel simulated annealing [26]. Finally it will be interesting to extend our method to the setting partial differential equations based models that capture spatial aspects of biopathways dynamics.

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015), `http://tensorflow.org/`, software available from tensorflow.org
2. Aldridge, B.B., Burke, J.M., Lauffenburger, D.A., Sorger, P.K.: Physicochemical modelling of cell signalling pathways. Nature cell biology 8(11), 1195–1203 (2006)
3. Bartocci, E., Bortolussi, L., Nenzi, L., Sanguinetti, G.: System design of stochastic models using robustness of temporal properties. Theoretical Computer Science 587, 3 – 25 (2015), `http://www.sciencedirect.com/science/article/pii/S0304397515002224`, interactions between Computer Science and Biology
4. Bortolussi, L., Sanguinetti, G.: Learning and Designing Stochastic Processes from Logical Constraints, pp. 89–105. Springer Berlin Heidelberg, Berlin, Heidelberg (2013), `http://dx.doi.org/10.1007/978-3-642-40196-1_7`
5. Brown, K.S., Hill, C.C., Calero, G.A., Myers, C.R., Lee, K.H., Sethna, J.P., Cerione, R.A.: The statistical mechanics of complex signaling networks: nerve growth factor signaling. Physical biology 1(3), 184 (2004)
6. Bucher, J., Riedmaier, S., Schnabel, A., Marcus, K., Vacun, G., Weiss, T.S., Thasler, W.E., Nüssler, A.K., Zanger, U.M., Reuss, M.: A systems biology approach to dynamic modeling and inter-subject variability of statin pharmacokinetics in human hepatocytes. BMC systems biology 5(1), 1 (2011)
7. Bufo, S., Bartocci, E., Sanguinetti, G., Borelli, M., Lucangelo, U., Bortolussi, L.: Temporal Logic Based Monitoring of Assisted Ventilation in Intensive Care Patients, pp. 391–403. Springer Berlin Heidelberg, Berlin, Heidelberg (2014), `http://dx.doi.org/10.1007/978-3-662-45231-8_30`
8. Chen, G., Sabato, Z., Kong, Z.: Active learning based requirement mining for cyber-physical systems. In: Decision and Control (CDC), 2016 IEEE 55th Conference on. pp. 4586–4593. IEEE (2016)
9. Donzé, A., Maler, O.: Robust satisfaction of temporal logic over real-valued signals. In: FORMATS. vol. 6246, pp. 92–106. Springer (2010)
10. Fainekos, G.E., Pappas, G.J.: Robustness of temporal logic specifications for continuous-time signals. Theoretical Computer Science 410(42), 4262–4291 (2009)
11. Goldbeter, A., Pourquié, O.: Modeling the segmentation clock as a network of coupled oscillations in the notch, wnt and fgf signaling pathways. Journal of theoretical biology 252(3), 574–585 (2008)
12. Grosu, R., Smolka, S.A., Corradini, F., Wasilewska, A., Entcheva, E., Bartocci, E.: Learning and detecting emergent behavior in networks of cardiac myocytes. Communications of the ACM 52(3), 97–105 (2009)
13. Heath, J., Kwiatkowska, M., Norman, G., Parker, D., Tymchyshyn, O.: Probabilistic model checking of complex biological pathways. Theoretical Computer Science 391(3), 239–257 (2008)
14. Hockin, M.F., Cawthern, K.M., Kalafatis, M., Mann, K.G.: A model describing the inactivation of factor va by apc: bond cleavage, fragment dissociation, and product inhibition. Biochemistry 38(21), 6918–6934 (1999)

15. Hoxha, B., Dokhanchi, A., Fainekos, G.: Mining parametric temporal logic properties in model-based design for cyber-physical systems. International Journal on Software Tools for Technology Transfer (Feb 2017), http://dx.doi.org/10.1007/s10009-017-0447-4
16. Jha, S.K., Clarke, E.M., Langmead, C.J., Legay, A., Platzer, A., Zuliani, P.: A bayesian approach to model checking biological systems. In: Computational Methods in Systems Biology. pp. 218–234. Springer (2009)
17. Jin, X., Donzé, A., Deshmukh, J.V., Seshia, S.A.: Mining requirements from closed-loop control models. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 34(11), 1704–1717 (2015)
18. Juty, N., Ali, R., Glont, M., Keating, S., Rodriguez, N., Swat, M.J., Wimalaratne, S.M., Hermjakob, H., Le Novère, N., Laibe, C., Chelliah, V.: BioModels: Content, Features, Functionality and Use. CPT: Pharmacometrics & Systems Pharmacology (2015)
19. Kass, R.E., Raftery, A.E.: Bayes factors. Journal of the american statistical association 90(430), 773–795 (1995)
20. Kholodenko, B.N.: Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activated protein kinase cascades. European Journal of Biochemistry 267(6), 1583–1588 (2000)
21. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., et al.: Optimization by simulated annealing. science 220(4598), 671–680 (1983)
22. Langlois, W.J., Sasaoka, T., Saltiel, A.R., Olefsky, J.M.: Negative feedback regulation and desensitization of insulin-and epidermal growth factor-stimulated p21ras activation. Journal of Biological Chemistry 270(43), 25320–25323 (1995)
23. Le Novere, N., Bornstein, B., Broicher, A., Courtot, M., Donizelli, M., Dharuri, H., Li, L., Sauro, H., Schilstra, M., Shapiro, B., et al.: Biomodels database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. Nucleic acids research 34(suppl 1), D689–D691 (2006)
24. LeCun, Y., Bengio, Y.: The handbook of brain theory and neural networks. chap. Convolutional Networks for Images, Speech, and Time Series, pp. 255–258. MIT Press, Cambridge, MA, USA (1998), http://dl.acm.org/citation.cfm?id=303568.303704
25. Neumann, L., Pforr, C., Beaudouin, J., Pappa, A., Fricker, N., Krammer, P.H., Lavrik, I.N., Eils, R.: Dynamics within the cd95 death-inducing signaling complex decide life and death of cells. Molecular Systems Biology 6(1), 352 (2010)
26. Onbaşoğlu, E., Özdamar, L.: Parallel simulated annealing algorithms in global optimization. Journal of Global Optimization 19(1), 27–50 (2001), http://dx.doi.org/10.1023/A:1008350810199
27. Palaniappan, S.K., Gyori, B.M., Liu, B., Hsu, D., Thiagarajan, P.: Statistical model checking based calibration and analysis of bio-pathway models. In: Computational Methods in Systems Biology. pp. 120–134. Springer (2013)
28. Rizk, A., Batt, G., Fages, F., Soliman, S.: Continuous valuations of temporal logic specifications with applications to parameter optimization and robustness measures. Theor. Comput. Sci. 412(26), 2827–2839 (Jun 2011), http://dx.doi.org/10.1016/j.tcs.2010.05.008
29. Seggern, D.v.: CRC standard curves and surfaces. CRC Press, 1 edn. (1993)
30. Zheng, Y., Liu, Q., Chen, E., Ge, Y., Zhao, J.L.: Time Series Classification Using Multi-Channels Deep Convolutional Neural Networks, pp. 298–310. Springer International Publishing, Cham (2014), http://dx.doi.org/10.1007/978-3-319-08010-9_33

31. Zhou, J., Ramanathan, R., Wong, W.F., Thiagarajan, P.: Automated property synthesis of odes based bio-pathways models, http://www.comp.nus.edu.sg/~zhoujun/full_report.pdf