Parameter Scanning by Parallel Model Checking with Applications in Systems Biology

J. Barnat, L. Brim, D. Šafránek, and M. Vejnár
Faculty of Informatics
Masaryk University
Brno, Czech Republic
Email: safranek@fi.muni.cz

***Abstract*—In this paper, a novel scalable method for scanning of kinetic parameter values in continuous (ODE) models of biological networks is provided. The presented method is property-driven, in particular, parameter values are scanned in order to satisfy a given dynamic property. The key result – the parameter scanning method – is based on an innovative adaptation of parallel LTL model checking for the framework of parameterized Kripke structures (PKS). First, we introduce the notion of PKS and we identify the parameter scanning and robustness analysis problems in this framework. Second, we present the algorithms for parallel LTL model checking on PKSs. Finally, the evaluation is provided on case studies of mammalian cell-cycle genetic regulatory network model and *E. Coli* ammonium transport model.**

***Keywords*-biological networks; parallel model checking; dynamic systems; parameter scanning; systems biology**

I. Introduction

Efficient analysis of biological models under parameter uncertainty is an important topic in current computational systems biology [1] and it provides a fundamental cornerstone for synthetic biology [2]. Automated methods of searching for optimal and robust sets of kinetic parameters with respect to required dynamic properties form an important precursor of model reconstruction [3], [4]. Given the complexity of the problem and the need for comprehensive large-scale models, there is a natural effort to develop techniques prepared to perform efficiently on high-performance computing platforms [5].

A widely used modeling framework for biological systems is provided by continuous deterministic approach based on ordinary differential equations (ODE models). Model checking [6] has been frequently employed for dynamic analysis of ODE models performed on numerical simulations [4], [7], [8] or on finite discrete abstractions [9]–[11]. Recent work in this area aims at application of model checking to scanning of model parameters. In particular, the common goal of *parameter scanning methods* [3], [7], [12], [13] is to determine the maximal set of parameter values for which a parameterized biological model satisfies a given property. To emphasize this context, we refer to the problem as *property-driven parameter scanning*. Dynamic properties can be succinctly expressed in a temporal logic [9], [14]. The property-driven parameter synthesis is complementary to traditional parameter estimation methods [15] that synthesize parameters from experimentally measured data.

We revisit the parameter synthesis approach of Batt et al. [11] which is performed over discrete abstractions of ODE models. Such an approach is unique in the employed view on the system dynamics. In particular, the dynamic properties with respect to which the parameters are scanned are viewed as global properties independent of particular setting of initial conditions (initial values of the state variables). An example of such a property is presence of multiple steady states. The global view provides biologists a tool which for a given model and a property computes maximal sets of parameter values for which the model dynamics entirely fulfills the phenomenon stated by the universally interpreted temporal property. Such a globally oriented approach is complementary to simulation-based approaches [7], [12] that are exposed to exponential explosion of possible initial conditions which increases with number of state variables. However, the advantage of the global view provided by the discrete abstraction is always traded for the extent of approximation imposed.

Owing to the need for a tractable discrete abstraction, the approach is restricted to a specific class of non-linear ODEs. In particular, we consider models where the non-linear function controlling the dynamics is approximated in terms of a piece-wise multi-affine function. Several studies [14], [16], [17] showed that such approximations give good predictions on biological models. The multi-affine restriction enables abstraction of the model phase space into a finite discrete quotient [3], [18] represented in terms of a finite automaton. The automaton can be directly turned into a Kripke structure and analyzed by model checking. Since model checking-based parameter synthesis performed over discrete abstractions of ODEs has proved to be computationally demanding [11], we develop a new algorithm based on traditional parallel enumerative on-the-fly LTL model checking techniques. That way we enable possibility for high-performance computing whilst using certain properties of parameterized biological models in order to employ heuristics that reduce computational times in expected cases.

We consider the problem of parameter scanning of unknown kinetic parameters for the given (finite) set of possible parameter values (*valuations*). The naive approach is to generate an automaton for each valuation and to perform model checking to ensure the automaton satisfies the property. During our previous experiments we have identified an interesting observation showing that in the case of discrete finite quotients of ODE models, small perturbations in parameter values lead to small locally distributed variations in the transition relation. In consequence, the respective automata exhibit significant similarity. Motivated by this, we propose a general scalable method that performs model checking on many similar Kripke structures simultaneously. If we denote each parameter valuation by a distinct color and assume that the respective (valuation-specific) Kripke structure has all its transitions colored by this color, we can construct a global Kripke structure as a union of all the valuation-specific Kripke structures. The parameter scanning problem can be then viewed as analysis of infinite mono-colored paths in a graph with multi-colored edges.

The fundamental contribution of this paper is formulation of the notion of a *parameterized Kripke structure* (PKS) that unifies a family of Kripke structures sharing the same set of states but differing in some transitions (Section II). PKS parameterizes the transition relation and includes every Kripke structure that instantiates the transition relation for a particular valuation. In Section III we present the main result – the general algorithm for LTL model checking of PKSs. Next, we present a generalized version of the piece-wise multi-affine modeling framework [3] to incorporate a wider set of ODE models (Section IV) and we apply the proposed approach to two different models (Section V). Finally, we evaluate the parallel shared-memory implementation of our algorithm (Section VI).

A. Related Work

The first algorithm for parameter scanning of ODE models based on model checking performed over discrete abstractions has been introduced by Batt et al. in [3], [19]. The abstraction is set up to fit the models of genetic regulatory networks. In this paper we adapt this abstraction method for a generalized set of models. In particular, our framework directly considers also models where regulatory kinetics (e.g., Michaelis-Menten or Hill kinetics) is arbitrarily mixed with mass action kinetics.

Batt et al. defined a general technique for finite partitioning of the parameter space into classes of behaviorally equivalent valuations. We employ this technique in a restricted form – we consider rectangular partitioning provided that only linearly independent parameters can be considered unknown. This restriction has been employed purely due to implementation efficiency – we avoided the use of computationally expensive algorithms which would be otherwise necessary for more complicated parameter spaces.

The algorithm of [3] is sequential and relies on execution of two (symbolic) model checking procedures per each class of valuations. In the expected case, the number of analyzed parameter classes can be reduced by suitable BDD representation of the parameter space. On the contrary, our algorithm is based on a principally different idea. Enumerative LTL model checking procedure (reduced to detection of accepting cycles in the model-property product automaton) is employed directly on a graph that compactly represents the dynamics of all valuations. The computational effort can be significantly reduced in the expected case due to small variations in subgraphs corresponding to different valuations. Moreover, we take advantage of the choice of enumerative model checking and develop a parallel algorithm that accelerates the computation with increasing number of CPUs.

A traditional approach based on numerical simulation is employed in [4]. Numerical simulation produces approximation of trajectories generated with respect to a given starting point in the model phase space – the initial condition. In contrast to that work, the abstraction approach employed in this paper allows model checking of global dynamic properties – the initial condition is generalized to any compact subset of the phase space.

Batt et al. [13] targets the piece-wise affine framework specialized for regulatory networks. The notion of uncertainty is lifted to a higher level of abstraction provided that qualitative (symbolic) valuation is considered. On the other hand, we provide a quantitative parameter scanning method where kinetic parameters are numerically valuated.

II. Problem Definition

In this section we define the notion of a parameterized Kripke structure that allows us to manipulate families of Kripke structures, in which all member Kripke structures have the same set of states (the set of reachable states can differ among individual Kripke structures). In consequence, we formally specify the parameter scanning problem.

We define a *parameterized (fair) Kripke structure* (PKS) to be a tuple $\mathcal{K} = (\mathcal{P}, S, I, F, \rightarrow, L)$, where $\mathcal{P}$ denotes the finite *set of parameter values* (*valuations*), S denotes the finite set of states, $I \subseteq S$ the set of initial states, $F \subseteq S$ is the set of fair states, $L\colon S \to 2^{\mathrm{AP}}$ is the labelling of states by atomic propositions, and $\rightarrow \subseteq S \times \mathcal{P} \times S$ is a transition relation labelled by parameter valuations. We denote the states $\alpha, \beta, \ldots$. We write $\alpha \xrightarrow{p} \beta$ instead of $(\alpha, p, \beta) \in \rightarrow$. We write $\alpha \rightarrow \beta$ if $\alpha \xrightarrow{p} \beta$ for some $p \in \mathcal{P}$. Fixing a valuation $p \in \mathcal{P}$ reduces the parametrized Kripke structure $\mathcal{K}$ to the (non-parameterized) Kripke structure $\mathcal{K}(p) = (S, I, F, \xrightarrow{p}, L)$.

We use the notation $\mathcal{P}(\alpha, \beta) = \{p \in \mathcal{P} \mid \alpha \xrightarrow{p} \beta\}$ to denote the set of parameter values for which a transition from $\alpha \in S$ to $\beta \in S$ is defined.

We define a p-path in the PKS $\mathcal{K}$ as a (possibly infinite) sequence of states $\pi = \alpha_0 \alpha_1 \cdots$ such that $\alpha_0 \in I$, $\alpha_i \xrightarrow{p}$

α_{i+1} for all $i \geq 0$. We refer to p-paths of length n such that $\alpha_0 = \alpha_{n-1}$ as p-cycles. A p-run is defined as a fair infinite-length p-path, i.e., a p-path such that there is a state $\gamma \in F$ which appears in the sequence infinitely often.

We say that a p-path (p-cycle, p-run) is a P-path (P-cycle, P-run) for $P \subseteq \mathcal{P}$, if $p \in P$. We refer to $\mathcal{P}$-paths ($\mathcal{P}$-cycles, $\mathcal{P}$-runs) simply as paths (cycles, runs).

The run $\pi = \alpha_0\alpha_1 \cdots$ satisfies the LTL formula φ over the set of atomic propositions AP, written $\pi \models \varphi$, if the infinite word $L(\alpha_0)L(\alpha_1) \ldots \in (2^{\mathrm{AP}})^\omega$ satisfies φ:

We wish to be able to solve the following problems, given the parameterized Kripke structure $\mathcal{K}$ and formula φ:

1) *Robustness*. Decide whether all runs in $\mathcal{K}$ satisfy φ.
2) *Parameter scan*. Find the maximal $P \subseteq \mathcal{P}$ such that all P-runs in $\mathcal{K}$ satisfy φ.

The robustness problem can be reduced to the parameter scan problem in a straightforward manner. The parameter scan is effectively the problem of finding all $p \in \mathcal{P}$ such that all runs on $\mathcal{K}(p)$ satisfy φ and can therefore be solved by performing any of the well-known model checking procedures [6] on Kripke structures $\mathcal{K}(p)$ for all $p \in \mathcal{P}$. Such approach takes time linear in $|\mathcal{P}|$.

We conjecture that such an approach is optimal for the general case and therefore limit ourselves to the study of well-behaved systems – PKSs such that the parameter value sets $\mathcal{P}(\alpha, \beta)$ can be represented compactly, and in which a small change in the parameter value corresponds to a small change in the respective Kripke structure. Our goal is to provide an algorithm which would in practice perform parameter scans on well-behaved systems reasonably fast.

III. Parameter Scan Algorithm

The traditional approach to model checking [6] a Kripke structure $\mathcal{K}$ against an LTL formula φ is to

1) construct a Büchi automaton over 2^{AP} such that it accepts all words satisfying $\neg\varphi$,
2) convert $\mathcal{K}$ to a Büchi automaton which accepts words of the form $L(\alpha_1)L(\alpha_2)\cdots$ for each run $\alpha_1\alpha_2\cdots$ in $\mathcal{K}$,
3) compute the synchronous product of the two automata, and
4) decide whether the resulting automaton accepts an empty language.

Our parameter scan algorithm is based on this approach. However, we define the synchronous product directly between the PKS and the Büchi automaton in order to avoid defining a parameterized version of Büchi automata.

A (non-deterministic) *Büchi automaton* is a tuple $\mathcal{B} = (\Sigma, S, s_0, \delta, F)$, where Σ is the input alphabet, S is a finite non-empty set of states, $s_0 \in S$ is a distinguished initial state, $\delta \subseteq S \times \Sigma \times S$ is a transition relation, and $F \subseteq S$ is the set of accepting states.

The automaton $\mathcal{B}$ accepts the infinite word $w \in \Sigma^\omega$, if there is a sequence of states $s_0 s_1 \ldots$ such that $(s_i, w_i, s_{i+1}) \in \delta$ and there is a state $f \in F$, which appears in the sequence infinitely often. The language of $\mathcal{B}$, written $L(\mathcal{B})$, is the set of words $w \in \Sigma^\omega$ that are accepted by $\mathcal{B}$.

There are well-known algorithms [6], which given an LTL formula φ over the set of atomic propositions AP construct a Büchi automaton $\mathcal{B}_\varphi$ over the alphabet 2^{AP} accepting exactly the words satisfying φ, i.e., $L(\mathcal{B}_\varphi) = \{w \in (2^{\mathrm{AP}})^\omega \mid w \models \varphi\}$.

Define the *synchronous product of the PKS* $\mathcal{K} = (\mathcal{P}, S_1, I, F_1, \to, L)$ *and the Büchi automaton* $\mathcal{B} = (2^{\mathrm{AP}}, S_2, s_0, \delta, F_2)$ as a PKS $\mathcal{K} \times \mathcal{B} = (\mathcal{P}, S', I', F', \to', L')$ where

- $S' = S_1 \times S_2 \times \{1, 2\}$,
- $I' = I \times \{s_0\} \times \{1\}$,
- $F' = F_1 \times S_2 \times \{1\}$,
- $L'(\alpha, s, i) = L(\alpha)$, and
- $(\alpha, s, i) \xrightarrow{p}{}' (\beta, t, j)$ if $\alpha \xrightarrow{p} \beta$, $(s, L(\alpha), t) \in \delta$, and

$$j = \begin{cases} 1, & \text{if } i = 1 \wedge \alpha \notin F_1 \text{ or } i = 2 \wedge s \in F_2 \\ 2, & \text{otherwise.} \end{cases}$$

The synchronous product $\mathcal{K} \times \mathcal{B}$ contains exactly those runs of $\mathcal{K}$ on which the automaton $\mathcal{B}$ accepts. Hence every p-run in $\mathcal{K} \times \mathcal{B}_{\neg\varphi}$ corresponds to a p-run in $\mathcal{K}$ that violates the formula φ. As a result the parameter scan problem – the problem of finding the maximal $P \subseteq \mathcal{P}$ such that all P-runs in $\mathcal{K}$ satisfy φ – can be efficiently reduced to the problem of finding the maximal set $P \subseteq \mathcal{P}$ such that for every $p \in P$, there is no p-run in the product $\mathcal{K} \times \mathcal{B}_{\neg\varphi}$.

The PKS $\mathcal{K}$ contains a p-run if and only if there exist states $\alpha \in I$, $\gamma \in F$ such that $\alpha \xrightarrow{p}{}^* \gamma \xrightarrow{p}{}^+ \gamma$. We rewrite this condition equivalently as $\alpha \xrightarrow{p}{}^+ \gamma \xrightarrow{p}{}^+ \gamma$. Next, we split the problem into two subtasks:

1) For each $\gamma \in F$ compute the maximal set of parameter values $P \subseteq \mathcal{P}$ such that for all $p \in P$ there is an initial state $\alpha \in I$ such that $\alpha \xrightarrow{p}{}^+ \gamma$.
2) For each $\gamma \in F$ and the corresponding parameter value set $P \subseteq \mathcal{P}$, determine if there is a P-cycle through γ of length at least 1.

We call a mapping of parameter value sets to states a *coloring*, and define it formally as a function $f \colon S \to 2^{\mathcal{P}}$. The coloring that we compute in the first step is called the *initial coloring*.

To detect a P-cycle through the state $\gamma \in F$, we compute the coloring $\mathrm{Succ}(\gamma, P)$ which assigns to each state $\alpha \in S$ the maximal set of parameter values $P' \subseteq P$ such that $\gamma \xrightarrow{p}{}^+ \alpha$ for all $p \in P'$. Formally, we define $\mathrm{Succ}(\gamma, P)(\alpha) = \{p \in P \mid \gamma \xrightarrow{p}{}^+ \alpha\}$. The set of parameter values $\mathrm{Succ}(\gamma, P)(\gamma)$ includes exactly all parameter values p such that there is a non-empty p-cycle through γ.

We extend $\mathrm{Succ}(\gamma, P)$ homomorphically to $\mathrm{Succ}(S', P)$ for $S' \subseteq S$, $P \subseteq \mathcal{P}$, where $\mathrm{Succ}(S', P)(\alpha) = \bigcup_{\gamma \in S'} \mathrm{Succ}(\gamma, P)(\alpha)$. Note that the initial coloring is exactly $\mathrm{Succ}(I, \mathcal{P})$.

Algorithm 1 performs the parameter scan in the manner described above. The algorithm iteratively accumulates the parameter value set P and can terminate as soon as $P = \mathcal{P}$, which allows for a fast discovery of runs violating a given property. It is also worth noting that once the existence of a p-run in the PKS $\mathcal{K}$ is detected for some $p \in \mathcal{P}$, it is not necessary to search for p-cycles any longer.

Algorithm 1 Parameter scan

Require: $\mathcal{K} = (\mathcal{P}, S, I, F, \to, L)$
Ensure: $p \in P$ iff $\alpha \xrightarrow{p}{}^{*} \gamma \xrightarrow{p}{}^{+} \gamma$ for some $\alpha \in I$, $\gamma \in F$
1: $P \leftarrow \emptyset$
2: $R \leftarrow \mathrm{Succ}(I, \mathcal{P})$
3: **for all** $\gamma \in F$, $R[\gamma] \setminus P \neq \emptyset$ **do**
4: $\quad P \leftarrow P \cup \mathrm{Succ}(\gamma, R[\gamma] \setminus P)(\gamma)$

Next we give an algorithm for computing $\mathrm{Succ}(S', P)$ for arbitrary $S' \subseteq S$, $P \subseteq \mathcal{P}$. Recall the earlier assumption that the system changes slightly with respect to changes in parameter values, from which we conclude that it is likely that any particular path will be present in the system for several parameter values. Our goal is to take advantage of this and detect paths for all parameter values at once.

Algorithm 2 computes $\mathrm{Succ}(S', P)$. It starts with an empty coloring and incrementally updates it. An update is a tuple (α, P), meaning that the set of parameter values $P \subseteq \mathcal{P}$ should be added to the coloring for the state α. The set of pending updates is stored in Q. The algorithm terminates as soon as there are no more pending updates.

Denote $Q(\alpha) = \{p \in \mathcal{P} \mid \exists P \subseteq \mathcal{P}. p \in P \wedge (\alpha, P) \in Q\}$ the set of parameter values that are currently scheduled to be added to the coloring for state α. In order to prevent Q from containing multiple updates for the same state, we use the merge operation $Q \oplus Q'$ defined as $Q \oplus Q' = \{(\alpha, P) \mid P = Q(\alpha) \cup Q'(\alpha) \wedge P \neq \emptyset\}$ to update Q (line 8). The update of Q can be performed in constant time with respect to the size of Q (and linear in the size of Q').

Algorithm 2 Compute $\mathrm{Succ}(S', P)$ over the PKS $\mathcal{K}$

Require: $\mathcal{K} = (\mathcal{P}, S, I, F, \to, L)$, $P \subseteq \mathcal{P}$, $S' \subseteq S$
Ensure: $R[\alpha] = \mathrm{Succ}(S', P)(\alpha)$
1: **for all** $\alpha \in S$ **do**
2: $\quad R[\alpha] \leftarrow \emptyset$
3: $Q \leftarrow \{(\beta, \mathcal{P} \cap P(\alpha, \beta)) \mid \alpha \to \beta, \alpha \in S'\}$
4: **while** $Q \neq \emptyset$ **do**
5: $\quad$ remove (α, P) from Q
6: $\quad$ **if** $P \not\subseteq R[\alpha]$ **then**
7: $\quad\quad R[\alpha] \leftarrow R[\alpha] \cup P$
8: $\quad\quad Q \leftarrow Q \oplus \{(\beta, P \cap \mathcal{P}(\alpha, \beta)) \mid \alpha \to \beta, \beta \in S\}$

Theorem 1: Given the parameterized Kripke structure $K = (\mathcal{P}, S, I, F, \to, L)$, the value of $R[\alpha]$ for any $\alpha \in S$ at any time during the computation can be constructed from the sets $\mathcal{P}(\alpha, \beta)$ with $\alpha, \beta \in S$ using $\cup$ and $\cap$ operators.

In particular, the set of values that can be assigned to $R[\alpha]$ during the computation is finite. Therefore, there are only finitely many colorings that can be assigned to R. Since in each iteration either the set Q becomes strictly smaller or R is assigned a strictly larger coloring (if f, g are colorings, we define $f \leq g$ if for all $\alpha \in S$, $f(\alpha) \subseteq g(\alpha)$), the algorithm terminates in finitely many steps.

Theorem 2: At the beginning of each iteration,

- $p \in R[\alpha]$ only if there is a p-path from some $\alpha_0 \in S'$ to α, and
- if $p \notin R[\alpha]$ and there is a p-path $\alpha_0 \alpha_1 \dots \alpha_n$ with $\alpha_0 \in S'$ and $\alpha_n = \alpha$, then there exists i, $1 \leq i \leq n$ such that $(\alpha_i, P) \in Q$ for some P with $p \in P$.

Correctness of Algorithm 2 is concluded by noting that the algorithm terminates with $Q = \emptyset$, which implies $p \in R[\alpha]$ if and only if there is a p-path from some $\alpha_0 \in S'$ to α.

To provide complexity bounds, we assume that values are removed from Q in fifo manner. We can then consider the computation to be performed in phases corresponding to BFS levels.

Theorem 3: After k-th phase of the computation, if there is a p-path of length at most k from $\alpha \in S'$ to $\beta \in S$ then $p \in R[\beta]$.

Since for each p-path of length $n \geq |S|$ there is a p-path of length $m \leq |S|$ with the same initial and terminal state, the algorithm terminates after at most $|S|$ phases. For each phase, the size of Q is at most $|S|$, however updates to the queue on line 8 may take up to $O(|E|)$ time per phase, where $E = \{(\alpha, \beta) \mid \alpha \to \beta\}$. Therefore, the algorithm terminates after $O(|S||E|)$ iterations.

Note, that for expected models, the computation will terminate after much less than $|S|$ phases. The lower bound on the number of iterations is $\Omega(|E|)$, which is achieved for models for which there is some $P \subseteq \mathcal{P}$ such that for all $\alpha, \beta \in S$, $\mathcal{P}(\alpha, \beta) = P$. In the worst case, one operation on parameter value sets may take $O(|\mathcal{P}|)$ time. Algorithm 2 therefore has the time complexity of $O(|S||E||\mathcal{P}|)$.

Given the time bounds on Algorithm 2, we conclude that Algorithm 1 requires $O(|S||E||F||\mathcal{P}|)$ time to complete. We will later demonstrate that while the worst-case time complexity is rather high, the algorithm performs well on real biological models.

Since the algorithm traverses the graph in a breath-first manner, it easily offers itself for parallelization. We demonstrate the scalability of a parallel version of our approach in Section VI.

IV. Representation of Biological Models

A. Multi-affine ODE models

By employing the law of mass action [20], dynamics of any biochemical reaction network can be modeled in

terms of a system of coupled non-linear ordinary differential equations (ODE system) [21]. We consider multi-affine systems in the form $\dot{x} = f(x)$ where $x = (x_1, \ldots, x_n)$ is a vector of variables and $f = (f_1, ..., f_n) : \mathbb{R}^n \to \mathbb{R}^n$ is a vector of multi-affine functions. A multi-affine function is a polynomial in the variables $x_1, ..., x_n$ where the degree of any variable is at most 1. Every variable x_i represents concentration of the respective chemical species and is interpreted in $\mathbb{R}_+ = \{x \in \mathbb{R} \mid x \geq 0\}$. An example of a multi-affine system is given in Figure 2.

Since the variables are interpreted as non-negative reals, the continuous state space of the ODE system is considered in the positive orthant $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \mid x \geq 0\}$. With respect to the fact that the variables represent chemically unstable species which degrade in time, we consider a bounded continuous state space $\mathcal{D}$ given by the n-dimensional rectangle $\mathcal{D} = \prod_{i=1}^{n}[0, max_i] \subset \mathbb{R}^n$ where max_i is the upper concentration bound assumed for x_i.

B. Piece-wise multi-affine generalization

The multi-affine format of ODEs entirely fits the mass action kinetics with the only exception of homodimerization reactions. Any biochemical system can be expressed in terms of mass action kinetics assuming that all low-level reactions are specified. In practice, such a detailed approach would lead to models of enormous dimensions. To this end, simplifying mechanisms based on quasi-steady state approximation have been proposed to reduce the model dimensionality (e.g., Michaelis-Menten or Hill kinetics). Right-hand sides of reduced ODEs have the form of rational polynomials obtained as linear combinations of *regulatory functions* [22]. In general, the resulting non-linear ODE systems are not multi-affine. However, rational polynomial ODE systems can be approximated in terms of piece-wise multi-affine systems [19] by replacing each regulatory function with a suitable composition of piece-wise linear *ramp-functions* defined in the following way:

$$r^+(x_i, \theta_i, \theta'_i) = \begin{cases} 0, & \text{if } x_i \leq \theta_i, \\ \frac{x_i - \theta_i}{\theta'_i - \theta_i}, & \text{if } \theta_i < x_i < \theta'_i, \\ 1, & \text{if } x_i \geq \theta'_i. \end{cases}$$

where $\theta_i, \theta'_i \in \mathbb{R}^+$, $\theta_i < \theta'_i \leq max_i$. The negative ramp-function is defined $r^-(x_i, \theta_i, \theta'_i) = 1 - r^+(x_i, \theta_i, \theta'_i)$.

The *piece-wise multi-affine ODE model (PMA model)* $\mathcal{M}$ is given by an ODE system in the form $\dot{x} = f(x)$ where $x = (x_1, ..., x_n)$ is a vector of variables and $f = (f_1, ..., f_n) : \mathbb{R}^n \to \mathbb{R}^n$ is a vector of piece-wise multi-affine functions, and a *set of thresholds* $\theta^i_j \in \mathbb{R}$ satisfying $min_i = \theta^i_1 < \theta^i_2 < \ldots < \theta^i_{\zeta_i} = max_i$ for each variable x_i. The number of thresholds on x_i is denoted ζ_i, $\zeta_i \geq 2$. We denote Ω the *partition* of $\mathcal{M}$, $\Omega = \prod_{i=1}^{n}\{1, \ldots, \zeta_i - 1\}$. A function $g : \mathbb{R}^n \to \mathbb{R}^+$ is piece-wise multi-affine if it is multi-affine on each n-dimensional interval $(\theta^1_{j_1}, \theta^1_{j_1+1}) \times ... \times (\theta^n_{j_n}, \theta^n_{j_n+1})$ where $(j_1, ..., j_n) \in \Omega$ and $\forall i, 1 \leq i \leq n. j_i < max_i$.

We consider n-dimensional PMA models in which $f = (f_1, ..., f_n)$ has the form

$$\forall i \in \{1, ..., n\}. f_i(x) = \sum_{j \in I^+} \kappa^j_i \varrho^j_i(x) - \sum_{j \in I^-} \gamma^j_i \varrho^j_i(x)$$

where I^+ and I^- are finite index sets such that $I^+ \cap I^- = \emptyset$, $0 \neq \kappa^j_i, \gamma^j_i \in \mathbb{R}^+$ are kinetic parameters, and ϱ^j_i is an arbitrary regulatory function.

Regulatory function $\varrho(x)$ is defined inductively as follows:

- $\varrho(x) = 1$, $\varrho(x) = x_k$ or $\varrho(x) = r^*(x_k, \theta^k_l, \theta^k_{l+1})$ for any $k \in \{1, ..., n\}$, $* \in \{+, -\}$, $l \in \{1, ..., \zeta_k - 1\}$ are regulatory functions.
- If ϱ_1, ϱ_2 regulatory functions such that $dep(\varrho_1) \cap dep(\varrho_2) = \emptyset$ then $\varrho = 1 - \varrho_1$ and $\varrho = \varrho_1\varrho_2$ are regulatory functions, where $dep(\varrho)$ denotes the set of variables on which ϱ depends.

An example of a PMA model is given in Figure 3.

Note that in addition to definition in [19], we extend regulatory functions with the direct option to include constant and linear components. Since each term $\varrho(x) = x_k$ can be equivalently rewritten as $r^+(x_k, 0, max_k)$, this extension fits the piece-wise multi-affine framework.

The restriction imposed on the product of regulatory functions ensures f to be piece-wise multi-affine. Moreover, the considered format of equations makes the class of multi-affine models a proper subclass of PMA models. That way we obtain a sufficiently expressive formalism practically suitable for a significant set of biological systems.

C. Rectangular abstraction

For $\alpha \in \Omega$, $Re(\alpha)$ denotes a closed rectangular subset of $\mathbb{R}^n$ – the so-called *rectangle*, $Re(\alpha) \stackrel{\text{df}}{=} \prod_{i=1}^{n}\left[\theta^i_{\alpha_i}, \theta^i_{\alpha_i+1}\right]$. $Re_{\mathcal{V}}(\alpha)$ then denotes the set of vertices of $Re(\alpha)$, $Re_{\mathcal{V}}(\alpha) \stackrel{\text{df}}{=} \prod_{i=1}^{n}\{\theta^i_{\alpha_i}, \theta^i_{\alpha_i+1}\}$.

The *set of initial regions* of $\mathcal{M}$, denoted $\mathrm{Inset}(\mathcal{M})$, is defined as an arbitrary n-dimensional continuous subspace of $\mathcal{D}$ which corresponds to some subset of Ω.

Note that every pair $\alpha, \beta \in \Omega$ such that $\exists 1 \leq j \leq n. \beta_j = \alpha_j \pm 1$ and $\forall 1 \leq i \leq n, i \neq j. \beta_i = \alpha_i$ satisfies $||\alpha - \beta|| = 1$ where $||x||$ denotes the Euclid norm of $x \in \mathbb{R}^n$. For such α, β we say that $Re(\alpha)$ and $Re(\beta)$ are *neighboring rectangles*. Neighboring rectangles satisfy $Re(\alpha) \cap Re(\beta) \neq \emptyset$. The intersection of their realization is a hyperrectangular facet of dimension $(n-1)$. Recall that $||\alpha - \beta|| \leq 1$ means that either $\alpha = \beta$ or α, β are neighboring rectangles.

Let $\alpha, \beta \in \Omega$ be such that $||\alpha - \beta|| = 1$ with $1 \leq i \leq n$, $j \in \{-1, 1\}$ satisfying $\beta_i = \alpha_i + j$. We denote $of(\alpha, i, j)$ the *outgoing facet of* α *along* x_i *in direction* j, $of(\alpha, i, j) \stackrel{\text{df}}{=} Re(\alpha) \cap Re(\beta)$. The notation $of_{\mathcal{V}}(\alpha, i, j)$ then denotes the set of vertices of the respective outgoing facet, $of_{\mathcal{V}}(\alpha, i, j) \stackrel{\text{df}}{=} Re_{\mathcal{V}}(\alpha) \cap Re_{\mathcal{V}}(\beta)$.

Rectangle $Re(\alpha)$ is called *transient* if for every solution of $\dot{x}(t) = f(x)$ satisfying $x(t_0) \in Re(\alpha)$ for some $t_0 \in \mathbb{R}$ there exists $t_1 > t_0$ such that $x(t_1) \notin Re(\alpha)$.

The *rectangular abstraction transition system* for $\mathcal{M}$, written $RATS(\mathcal{M})$, is a triple (S, T, I) where $S \subseteq \Omega$ is the *set of states*, $I \subseteq S$ the *set of initial states* corresponding to $\mathrm{Inset}(\mathcal{M})$, and $T \subseteq S \times S$ the *transition relation*.

The relation T contains only those pairs $\langle \alpha, \beta \rangle$ for which $||\alpha - \beta|| \leq 1$ and either of the following conditions holds:

1) $||\alpha - \beta|| = 1$ with $1 \leq i \leq n$, $j \in \{-1, 1\}$ such that $\beta_i = \alpha_i + j$ and there exists $v \in of_{\mathcal{V}}(\alpha, i, j)$ satisfying $f_i(v) \cdot j > 0$.
2) $||\alpha - \beta|| = 0$ and $\vec{0} \in hull\{f(v) \mid v \in Re_{\mathcal{V}}(\alpha)\}$ where $hull$ denotes the convex hull of the given set.

We denote the fact $\langle \alpha, \beta \rangle \in T$ as $\alpha \to \beta$.

In [19] it is shown that $RATS(\mathcal{M})$ represents an over-approximation with respect to almost all trajectories in $\mathcal{M}$. Example of a rectangular abstraction is given in Figure 1.

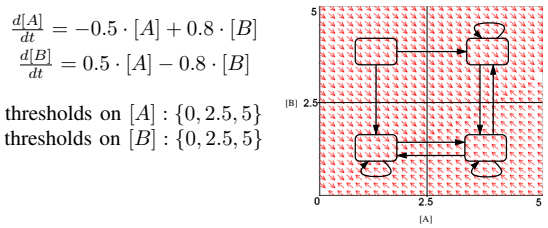


Figure 1. Example of a Rectangular Abstraction Transition System

D. Model Checking of RATS

To express dynamical properties of paths in rectangular abstraction of an n-dimensional model $\mathcal{M}$ we employ traditional Linear Temporal Logic (LTL) built over atomic propositions AP:

$$AP = \left\{ x_i \odot \theta_j^i \mid 1 \leq i \leq n, 1 \leq j \leq \zeta_i \}, \odot \in \{<, >\} \right\}.$$

$RATS(\mathcal{M}) \equiv (S, T, I)$ can be directly turned into a Kripke structure $\mathcal{K}_{\mathcal{M}}$ labeled by AP, $\mathcal{K}_{\mathcal{M}} = (S, T, I, L)$, $L : S \to 2^{AP}$. That way, LTL is interpreted over infinite paths of $RATS$. For $\mathcal{K}_{\mathcal{M}}$ we denote $\pi \in \mathcal{K}_{\mathcal{M}}$ any infinite path $\pi = \alpha_0 \alpha_1 \cdots$ such that $\alpha_0 \in I$.

Given a PMA model $\mathcal{M}$ we can say that $\mathcal{M}$ satisfies a formula φ, written $\mathcal{M} \models \varphi$, only if all solution trajectories starting in $\mathrm{Inset}(\mathcal{M})$ satisfy φ. In the context of RATS, a formula φ is satisfied in $\mathcal{K}_{\mathcal{M}}$, written $\mathcal{K}_{\mathcal{M}} \models \varphi$, only if each infinite path $\pi \in \mathcal{K}_{\mathcal{M}}$ satisfies φ. To decide $\mathcal{K}_{\mathcal{M}} \models \varphi$, the traditional LTL model checking method is used [6]. The following theorem [19] characterizes the relation between validity of φ in the RATS and in the original PMA model.

Theorem 4: Consider a PMA model $\mathcal{M}$ and the respective Kripke structure $\mathcal{K}_{\mathcal{M}}$. If $\mathcal{K}_{\mathcal{M}} \models \varphi$ then $\mathcal{M} \models \varphi$.

The theorem above guarantees conservativeness of the abstraction. However, the converse (completeness) does not hold. In the case when there exists a path $\pi \in \mathcal{K}_{\mathcal{M}}$ such that $\pi \not\models \varphi$, π may not include any trajectory of $\mathcal{M}$. In such a case, π is a *false-positive* (*spurious*) path. There are two different and independent reasons for this phenomenon.

The first reason is the false transitivity of a sequence of adjacent rectangles. This phenomenon affects reachability. There were developed methods that decrease the extent of this issue by nullcline driven abstraction refinement [17] or by extending RATS with history [23].

The second reason is the presence of time-convergent paths made by spurious cycles in $\mathcal{K}_{\mathcal{M}}$. This issue affects liveness properties and it has been treated in [19] by identifying transiency of final cycles (lassos) appearing on infinite paths in $\mathcal{K}_{\mathcal{M}}$. Denote $C(\pi) \subseteq S$ the smallest region in which a run $\pi = \alpha_0 \alpha_1 \cdots$ eventually ends up cycling, i.e, $C(\pi)$ is the smallest set satisfying $\exists i \geq 0. \forall j \geq i. \alpha_j \in C(\pi)$. A sufficient condition implying the path is not time-convergent is stated in [19]:

Theorem 5: Let $\pi \in \mathcal{K}_{\mathcal{M}}$. If $\vec{0} \notin hull\{f(v) \mid v \in Re_{\mathcal{V}}(\alpha), \alpha \in C(\pi)\}$ then π is not time-convergent.

E. Model Parameterization

Let $\mathcal{M}$ be an n-dimensional PMA model. Let χ be the set of *unknown kinetic parameters* of $\mathcal{M}$ satisfying for all $\kappa_i^j, \gamma_i^j \in \chi$ that there is no $k \neq j$ such that $\kappa_i^k \in \chi$ or $\gamma_i^k \in \chi$. In other words, for each equation of the underlying PMA system there is at most one unknown parameter. Thus, $|\chi| \leq n$. Denote μ_i the unknown parameter in ith equation, $\mu_i = \kappa_i^j$ or $\mu_i = \gamma_i^j$, for some j. Further denote $I(\chi) = \{i \mid \mu_i \in \chi\}$ the ordered set of indices of unknown parameters. For each $i \in I(\chi)$ we consider the minimal and maximal parameter value min_{μ_i} and max_{μ_i}. For a given χ we define the *parameter space* $\mathcal{P} = \prod_{i \in I(\chi)} [min_{\mu_i}, max_{\mu_i}]$. Finally, we define a *valuation* of χ as any $p \in \prod_{i \in I(\chi)} [min_{\mu_i}, max_{\mu_i}]$.

Now denote $\mathcal{M}_\chi$ the *parameterized PMA model* where for any $i \in I(\chi)$ the respective multi-affine function $f_i(x)$ of $\mathcal{M}$ is parameterized to $f_i(x, \mu_i)$. The multi-dimensional function f is respectively parameterized to $f(x, \mu)$ where $\mu = \prod_{i \in I(\chi)} \mu_i$. From the format of each f_i it directly follows f_i is piece-wise affine in μ_i and thus $f(x, \mu)$ is piece-wise affine in μ.

It is known [19] that when considering the rectangular abstraction $RATS(\mathcal{M}) = (S, T, I)$, the parameter space $\mathcal{P}$ of $\mathcal{M}_\chi$ can be partitioned by a finite *parameter grid* $\mathcal{G}$ defined in the following way:

$$\mathcal{G} = \prod_{i \in I(\chi)} \{u \in \mathcal{P} \mid f_i(v, u) = 0, v \in Re_{\mathcal{V}}(\alpha), \alpha \in S\}.$$

Let $p \in \mathcal{P}$ be a *valuation* of χ. We denote $\mathcal{M}_p$ the PMA model obtained from $\mathcal{M}_\chi$ by substituting $f(x, p)$ for $f(x, \mu)$. We consider values on each axis of $\mathcal{G}$ ordered. The following theorem [19] characterizes classes of behaviorally equivalent parameter valuations.

Theorem 6: Let $\mathcal{M}_\chi$ be a parameterized PMA model. Let $u, u' \in \mathcal{G}$ be such that for each $i \in I(\chi)$, $u_i < u'_i$ and there is no $k \in \mathcal{G}_i$ satisfying $u_i < k < u'_i$. For any $p, q \in \mathcal{P}$ satisfying $u_i < p_i < u'_i$ and $u_i < q_i < u'_i$ for each $i \in I(\chi)$, it holds $RATS(\mathcal{M}_p) = RATS(\mathcal{M}_q)$, and hence $\mathcal{K}_p = \mathcal{K}_q$.

F. Translation to Parameterized Kripke Structure

A parameterized model $\mathcal{M}_\chi$ can be turned into a PKS $\mathcal{K} = (\mathcal{P}, S, I, F, \rightarrow, L)$, with $F = S$, and $\alpha \xrightarrow{p} \beta$ for $\alpha, \beta \in S$, $p \in \mathcal{P}$ if and only if $\langle \alpha, \beta \rangle \in T$ in $RATS(\mathcal{M}_p)$.

Let us denote $e(v, i, j)$ for $1 \leq i < n$, $j \in \{-1, 1\}$ the set of parameter values p such that $f_i(v, p) \cdot j > 0$. Due to piece-wise affinity of $f(x, p)$ in p, the set $e(v, i, j)$ is an intersection of $|\chi|$ half-spaces in the parameter space.

Given neighboring $\alpha, \beta \in S$ with $j \in \{-1, 1\}$ and i such that $\alpha_i = \beta_i + j$, the set $\mathcal{P}(\alpha, \beta)$ of parameter values p for which there is a transition from α to β in $RATS(\mathcal{M}_p)$ can be computed as $\mathcal{P}(\alpha, \beta) = \bigcup\{e(v, i, j) \mid v \in of_\mathcal{V}(\alpha, i, j)\}$.

To deal with self-transitions trivially implying time-convergency, we first define the set of parameter values $T(V)$, for which a region bounded by a set of vertices V is transient. Ideally, we would define the region as $\{p \in P \mid \vec{0} \notin hull\{f(v, p) \mid v \in Re_\mathcal{V}(\alpha)\}\}$. To simplify computation, we approximate the set by including only those parameter values, for which there is a dimension i and a direction $j \in \{-1, 1\}$ such that $\forall v \in V. f_i(v, p) \cdot j > 0$. We define $T(V) = \bigcup_{i=1}^{n} \bigcap_{v \in V} e(v, i, 1) \cup \bigcup_{i=1}^{n} \bigcap_{v \in V} e(v, i, -1)$ and we set $\mathcal{P}(\alpha, \alpha) = \mathcal{P} \setminus T(Re_\mathcal{V}(\alpha))$.

Note that the approximation above respects the conservativeness of the abstraction in the sense of Theorem 4. Observe that if $p \in T(V)$, then there is $1 \leq i \leq n$ and $j \in \{-1, 1\}$ such that $p \in \bigcap_{v \in V} e(v, i, j)$. Therefore for all $v \in V$, $f_i(v, p) \cdot j > 0$ and hence $\vec{0} \notin hull\{f(v, p) \mid v \in V\}$.

Note that the sets $\mathcal{P}(\alpha, \beta)$ can be represented as a bounded set of rectangular regions in $\mathcal{P}$, which makes the PKS $\mathcal{K}$ suitable for model checking.

V. Case Study

In this section we provide case studies for safety and liveness properties. The former demonstrates the power of parallel algorithm on a large model, whereas the latter, due to reasons mentioned below, is exemplified on a smaller model.

Note that both safety and liveness properties can be expressed in LTL and the particular type of a property affects neither the correctness nor the complexity of the parametrized model checking algorithm described in Section III. However, due to the nature of the particular discretization of biological models that we have chosen (see Section IV), many more spurious paths violating a property usually appear when checking a model against a liveness property. The results of liveness checking therefore tend to high extents of spurious paths than those of safety checking, often to such a degree that the resulting set of parameter values that satisfy the property is empty.

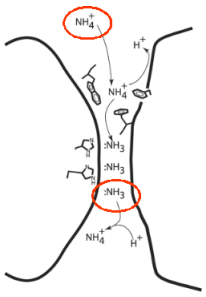
The approach employed by Batt et al. [19] to remove some of the spurious paths during liveness checking involves merging of states of the Kripke structure and is therefore not applicable in the framework of parametrized Kripke structures. We therefore utilize an alternative approach to reduce the number of spurious paths. The solution we propose in Section V-B is tuned for a particular two-variable model and it serves as a proof of concept.

A. Analysis Based on Safety Model Checking

We consider a model specifying ammonium transport from external environment into the cells of *E. Coli* [24]. The model describes the ammonium transport process that takes effect at very low external ammonium concentration. In such conditions, the transport process complements the deficient ammonium diffusion. The process is driven by a membrane-located ammonium transport protein $AmtB$ that binds external ammonium cations NH_4ex and uses their electrical potential to conduct NH_3 into the cytoplasm. In Figure 2, the scheme of the transport channel is shown (left) as well as the corresponding biochemical and mathematical model (right). For a sake of simplicity, pH conditions and external ammonium concentration are considered constant.

Owing to the membrane location of $AmtB$, *in vitro* measuring of concentration of $AmtB$-based species is impossible and therefore the estimation of kinetic parameters of this model is very difficult. To this end, we use the model to investigate the effect of different parameter settings to the production of the model output – the internal ammonium forms NH_3in and NH_4in. In particular, we look for perturbations in individual kinetic parameters that lead to an increase of internal ammonia concentration above the standard values. In terms of LTL model checking, we formulate negation of this requirement – we check whether the standard value is never exceeded. We formalize the discussed requirement by safety LTL properties $\varphi_1 = \mathbf{G}[NH_3in] < 1.1 \cdot 10^6$ and $\varphi_2 = \mathbf{G}[NH_4in] < 2.1 \cdot 10^6$.

We performed two groups of parameter scanning experiments. In the first group, each single parameter was unknown (remaining parameters were set w.r.t. literature [14], [24]). In the second group, we considered a set of three unknown parameters. In all experiments, the interval of uncertainty has been set to $(1 \cdot 10^{-12}, 1 \cdot 10^{12})$. In Table I, the most interesting results are summarized. The presented data show the scanned parameter set, the analyzed property with the computed valid parameter valuations, states of the PKS reached, and the best computation times. Note that each parameter which is not mentioned led trivially to validity on entire $(1 \cdot 10^{-12}, 1 \cdot 10^{12})$. Results for three parameters show good practicability also for complex parameter scanning problems – in the case of $\{k_1, k_6, k_9\}$, a large number of parameter valuation classes $(7.2 \cdot 10^4)$ were tested. This implies the significant increase of time. All experiments have been performed on an 8 core 2.2GHz CPU with 24GB RAM.



$$AmtB + NH_4ex \overset{k_2}{\underset{k_1}{\leftrightarrows}} AmtB:NH_4$$
$$AmtB:NH_4 \overset{k_3}{\to} AmtB:NH_3 + H_{ex}$$
$$AmtB:NH_3 \overset{k_4}{\to} AmtB + NH_3in$$
$$NH_4in \overset{k_5}{\to}$$
$$NH_3in + H_{in} \overset{k_7 k_6}{\leftrightarrows} NH_4in$$
$$NH_3in \overset{k_9 k_8}{\leftrightarrows} NH_3ex$$

$$\frac{d[AmtB]}{dt} = -k_1[AmtB][NH_4ex] + k_2[AmtB:NH_4] + k_4[AmtB:NH_3]$$
$$\frac{d[AmtB:NH_3]}{dt} = k_3[AmtB:NH_4] - k_4[AmtB:NH_3]$$
$$\frac{d[AmtB:NH_4]}{dt} = k_1[AmtB][NH_4ex] - k_2[AmtB:NH_4] - k_3[AmtB:NH_4]$$
$$\frac{d[NH_3in]}{dt} = k_4[AmtB:NH_3] - k_6[NH_3in][H_{in}] + k_7[NH_4in] + k_9[NH_3ex]$$
$$\frac{d[NH_4in]}{dt} = k_6[NH_3in][H_{in}] - k_5[NH_4in] - k_7[NH_4in]$$

constant species: $NH_3ex, NH_4ex, H_{in}, H_{ex}$

initial conditions and threshold numbers:

$AmtB \in \langle 0, 1\cdot 10^{-5}\rangle$ [7], $AmtB:NH_3 \in \langle 0, 1\cdot 10^{-5}\rangle$ [3], $AmtB:NH_4 \in \langle 0, 1\cdot 10^{-5}\rangle$ [3],
$NH_3in \in \langle 1\cdot 1^{-6}, 1.1\cdot 10^{-6}\rangle$ [6], $NH_4in \in \langle 2\cdot 10^{-6}, 2.1\cdot 10^{-6}\rangle$ [4],
$NH_3ex \in \langle 0, 1\cdot 10^{-5}\rangle$ [4], $NH_4ex \in \langle 0, 1\cdot 10^{-5}\rangle$ [4]

Figure 2. Ammonium Transport Model

P	prop.	intervals of validity	# states reached	time
k_4	φ_1	$(1\cdot 10^{-12}, 2.7\cdot 10^6)$	124580	30 s
k_5	φ_2	$(1.5\cdot 10^7, 1\cdot 10^{12})$	3068	0.40 s
k_6	φ_1	$(5.2\cdot 10^6, 1\cdot 10^{12})$	67572	22 s
k_6	φ_2	$\emptyset$	6319	1.8 s
k_7	φ_1	$(1\cdot 10^{-12}, 3.3\cdot 10^6)$	126458	33 s
k_7	φ_2	$(1.6\cdot 10^7, 1\cdot 10^{12})$	12523	3.5 s
k_9	φ_1	$(1\cdot 10^{-12}, 2.7\cdot 10^6)$	97495	20 s
k_9	φ_2	$\emptyset$	5779	1.5 s
$k_{1,6,9}$	φ_1	$k_9 \in (1\cdot 10^{-12}, 2.7\cdot 10^6) \vee$ $[k_9 \in (2.7\cdot 10^6, 3.2\cdot 10^6) \wedge k_6 \in (1\cdot 10^{-12}, 1.07\cdot 10^6)]$	202638	51 min
$k_{1,6,10}$	φ_2	$[k_1 \in (1\cdot 10^{-12}, 1\cdot 10^7) \wedge k_6 \in$ $(1\cdot 10^{-12}, 1.4\cdot 10^5) \wedge k_{10} \in (1.18\cdot 10^6, 1\cdot 10^{12})] \vee$ $[k_6 \in (1.4\cdot 10^5, 1.07\cdot 10^6) \wedge k_{10} \in (1\cdot 10^{-12}, 1.18\cdot 10^5)]$	19473	19 min

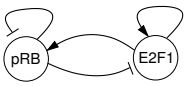
Table I
SAFETY MODEL CHECKING EXPERIMENTS

We have checked the computed validity intervals against numerical simulations. All the results show good estimation. Of special interest are individual scans of k_6 and k_9 for φ_2. In particular, the results show that in the given parameter value range there is no perturbation which would satisfy the property. With respect to both parameters, the model is robust in the negative property $\mathbf{F}[NH_4in] > 2.1\cdot 10^6$. Thus, regardless the setting of k_6, k_9, on each trajectory leading from the range specified by initial conditions, NH_4in must exceed the standard concentration range.

B. Model Checking of Liveness Properties

We investigate a model representing the central module of the genetic regulatory network governing the G_1/S cell cycle transition in mammalian cells [25]. In particular, the model considers a two-gene network describing interaction of the tumor suppressor protein pRB and the central transcription factor $E2F1$ (see Figure 3). We have abstracted the original non-linear model by replacing the non-linear Hill functions with piece-wise affine functions as defined in Section IV. The resulting PMA system is shown in Figure 3. Each function $\varrho_i(x)$ is defined as a sum of several ramp-functions that gradually approximate the respective regulatory S-shaped curve by a suitable polyline.

This system has a single stable state into which it eventually evolves regardless of the initial conditions. However, the precise configuration of the stable state is strongly influenced by the degradation coefficient γ_{pRB}. Figure 4 shows the state space of the above system for two different values of γ_{pRB}.



$$\frac{d[pRB]}{dt} = k_1\varrho_1(pRB, E2F1) - \gamma_{pRB}[pRB]$$
$$\frac{d[E2F1]}{dt} = k_p + k_2\varrho_2(pRB, E2F1) - \gamma_{E2F1}[E2F1]$$

Figure 3. Genetic Regulatory Network Model of G_1/S Transition

Our goal is to determine the set of parameters in the range [0.01, 1] for which the concentration of E2F1 is greater than 8 in the stable state. The LTL formula against which the system is model-checked is $\varphi = \mathbf{F}\,\mathbf{G}[E2F1] > 8$.

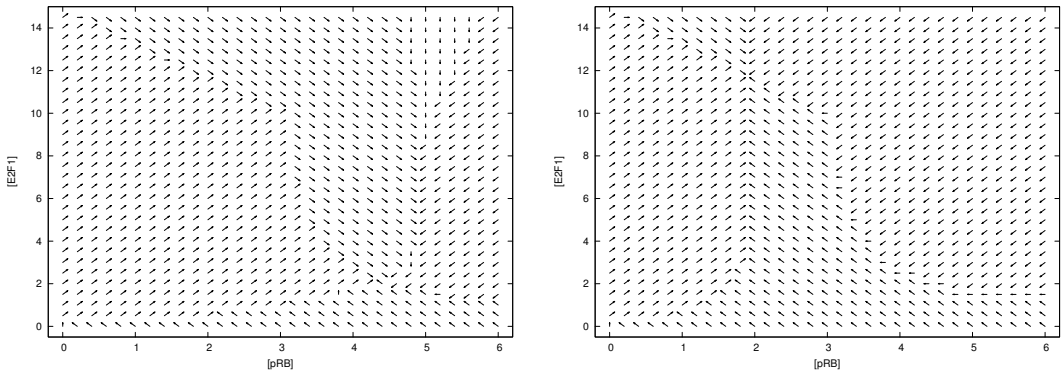
The system can be represented as a PKS $\mathcal{K} = (P, S, I, F, \to, L)$ with $F = S$ as described in Section IV. The liveness property φ is, however, negatively affected by the presence of spurious runs introduced by time-convergent paths of the original system. Our goal is to remove some of these runs from $\mathcal{K}$.

First, let us introduce the notation $\mathcal{T}(p) \subseteq 2^S$ denoting the set of regions that are transient in $\mathcal{K}(p)$, i.e., $\mathcal{T}(p) = \{H \subseteq S \mid p \in T(V), V = \bigcup_{\alpha\in H} of_{\mathcal{V}}(\alpha)\}$.

Consider now the class $\mathcal{S}_c$ of spurious runs such that for a p-run $\pi \in \mathcal{S}_c$, $C(\pi) \in \mathcal{T}(p)$. In other words, $\mathcal{S}_c$ is the set of runs which eventually cycle in a transient region. The set $\mathcal{S}_c$ is exactly the set of runs we want to remove from $\mathcal{K}$.

We therefore construct a PKS $\mathcal{K}' = (P, S', I', F', \to, L')$ such that it contains exactly those runs that correspond to runs in $\mathcal{K}$ with the exception of runs in $\mathcal{S}_c$. We do so by annotating the states with sets $H \subseteq S$, which correspond to the maximal transient regions recently visited. We define the transition relation $\to$ so that for each infinite-length p-path $\pi = \alpha_0\alpha_1\cdots$ in $\mathcal{K}$, which is also a p-run since all states in $\mathcal{K}$ are fair, there is an infinite-length p-path $\sigma = (\alpha_0, H_0)(\alpha_1, H_1)\cdots$ in $\mathcal{K}'$, and σ is a p-run if and only if $\pi \notin \mathcal{S}_c$. In other words, we make σ unfair if $C(\pi) \in \mathcal{T}(p)$. Formally, we set

- $S' = S \times 2^S$,
- $I' = I \times \{\emptyset\}$,
- $F' = S \times \{\emptyset\}$, and
- for each $\alpha \overset{p}{\to} \beta$ and each $H \subseteq S$ we add a transition



(a) $\gamma_{pRB} = 0.01$, the system stabilizes with $E2F1 < 3$ (b) $\gamma_{pRB} = 0.1$, the system stabilizes with $E2F1 > 11$

Figure 4. Vector field of the liveness model.

$(\alpha, H) \xrightarrow{p} (\beta, H')$, where

$$H' = \begin{cases} H \cup \{\alpha\}, & \text{if } H \cup \{\alpha, \beta\} \in \mathcal{T}(p), \text{ and} \\ \emptyset, & \text{otherwise.} \end{cases}$$

Note that there is a one-to-one correspondence between infinite paths in $\mathcal{K}$ and $\mathcal{K}'$. Moreover, for an infinite p-path $\sigma = (\alpha_0, H_0)(\alpha_1, H_1)\cdots$ in $\mathcal{K}'$ corresponding to a run $\pi \in \mathcal{S}_c$ the sequence $H_0 H_1 \cdots$ eventually stabilizes to some $H \supseteq C(\pi) \neq \emptyset$ and thus σ does not form a p-run. Inversely, if $\pi \notin \mathcal{S}_c$, and thus $C(\pi) \notin \mathcal{T}(p)$, the sequence $H_0 H_1 \cdots$ contains $\emptyset$ infinitely often, making σ a p-run.

Recall that only runs in $\mathcal{S}_c$ are removed from the PKS. Thus spurious runs may still be present in the transformed PKS. By applying this technique to the model described above we were able to prove that for $\gamma_{pRB} > 0.053$, the system stabilizes with $E2F1 > 8$. The PKS after the above transformation had 732 states reachable from initial states, the total of 827 states were colored. The computation took approximately 0.8 seconds on a single processor core.

VI. Scalability

We have developed a prototype implementation of our parameter scanning framework in C++. In order to show that our approach can be used in a high-performance environment, our implementation employs a multi-core data-parallel version of Algorithm 2. The states of the PKS are evenly distributed among threads using a hash function. Each thread then possesses the corresponding partition of the coloring R. The threads only communicate via the update queue Q, which is implemented as a set of lock-free queues, one queue per thread. We also found it necessary to keep the threads synchronized on BFS levels in order to achieve good times.

We have evaluated the scalability of our implementation by running the algorithm on the model described in Section V-A. The model was checked against the property $\varphi_1 = \mathbf{G}[NH_3in] < 1.1 \cdot 10^6$ (see Section V-A). We did not evaluate against property φ_2 since the individual checks were completed in less than 5 seconds.

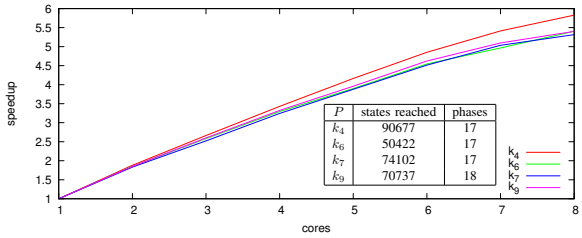


Figure 5. Scalability of the algorithm

Trials consisted of running the parameter scan several times on the model with single parameter from the set $\{k_4, k_6, k_7, k_9\}$ on the number of cores ranging from 1 to 8. The trials were run on an 8 core machine with 2.2GHz CPUs with 24GB RAM. The times of individual runs in each trial were then averaged. Results are shown in Figure 5. Table in the graph shows for each PKS the number of BFS levels and the number of reachable states, i.e., the number of states reached during the computation of initial coloring. Note that most of the running time of Algorithm 1 is spent in the computation of the initial coloring, therefore the numbers are close to the state counts from Table I.

Recall that the complexity of Algorithm 2 is $O(|S||E||\mathcal{P}|)$ and that the factor $|S|$ comes from an upper bound on the number of BFS levels in the PKS. Notice that on the ammonium transport model, the number of BFS levels is much smaller than $|S|$.

VII. Conclusions

We have presented a new approach to scanning of parameter values in parameterized dynamic systems. The approach is based on the model checking method provided that the parameter values are estimated w.r.t. a given temporal property. Our solution reduces parameter scanning to a general problem of efficient model checking of a parameterized family of Kripke structures in which the set of states is

shared but the transition relation is variable. At this level, we provide a model checking algorithm that can be in expected cases more efficient than a series of separate model checking procedures each performed on every Kripke structure in the given parameterized family.

To enable effective usage of high-performance computing, we have provided a parallel implementation of the algorithm tuned for multi-core platforms. The evaluation presented in the paper shows good scalability. We applied the approach to two biological models differing in the form of nonlinearity. The most complex model was 7-dimensional (2 inputs, 5 variables), partitioned to $1.6 \cdot 10^4$ rectangles. Parameter scans realized for a single unknown parameter were computed in seconds on a common multi-core hardware, parameter scans for three unknown parameters were finished in tens of minutes. When comparing with the sequential algorithm of [19] (4 hours for scanning three unknown parameters in a 5-dimensional model partitioned to $1.5 \cdot 10^3$ rectangles), we conclude that our parallel algorithm can significantly accelerate the computation.

Questions for the future work remain mainly in the topic of effective spurious counterexample elimination, especially for the case of liveness checking.

ACKNOWLEDGMENT

This work has been supported in part by the Czech Science Foundation grants No. 201/09/1389 and 201/09/P497.

REFERENCES

[1] H. Kitano, "Systems biology: a brief overview." *Science*, vol. 295, no. 5560, pp. 1662–4, 2002.

[2] Y.N.Kaznessis, "Models for synthetic biology," *BMC Syst. Biol.*, vol. 47, 2007.

[3] G. Batt, B. Yordanov, R. Weiss, and C. Belta, "Robustness analysis and tuning of synthetic gene networks," *Bioinformatics*, vol. 23, no. 18, pp. 2415–2422, 2007.

[4] A. R. et al., "A general computational method for robustness analysis with applications to synthetic gene networks," *Bioinformatics*, vol. 25, no. 12, pp. i169–178, 2009.

[5] P. Ballarini et.al., "Taming the complexity of biological pathways through parallel computing," *Brief. in Bioinformatics*, vol. 10, no. 3, pp. 278–288, 2009.

[6] E. Clarke, O. Grumberg, and D. Peled, *Model Checking*. MIT Press, 2000.

[7] R. Donaldson and D. Gilbert, "A model checking approach to the parameter estimation of biochemical pathways," in *Computational Methods in Systems Biology*, ser. LNBI, vol. 5307. Springer, 2008, pp. 269–287.

[8] S. J. et al., "A bayesian approach to model checking biological systems," in *Computational Methods in Systems Biology*, vol. 5688. Springer, 2009, pp. 218–234.

[9] G. Batt et.al., "Validation of qualitative models of genetic regulatory networks by model checking: analysis of the nutritional stress response in Escherichia coli," *Bioinformatics*, vol. 21, pp. 19–28, 2005.

[10] J. Barnat, L. Brim, and D. Šafránek, "High-performance analysis of biological systems dynamics with the divine model checker," *Brief. in Bioinformatics*, vol. 11, pp. 301–312, 2010.

[11] G. Batt, C. Belta, and R. Weiss, "Model checking liveness properties of genetic regulatory networks," in *TACAS 2007*, ser. LNCS, vol. 4424. Springer, 2007, pp. 323–338.

[12] A. Rizk et.al., "On a continuous degree of satisfaction of temporal logic formulae with applications to systems biology," in *Computational Methods in Systems Biology*, ser. LNBI, vol. 5307. Springer, 2008, pp. 251–268.

[13] G. Batt et.al., "Efficient parameter search for qualitative models of regulatory networks using symbolic model checking," INRIA, Tech. Rep., 2010.

[14] J. B. et al., "BioDiVinE: A Framework for Parallel Analysis of Biological Models," *Electronic Proceedings in Theoretical Computer Science*, vol. 6, pp. 31–45, 2009.

[15] M. Rodriguez-Fernandez, P. Mendes, and J. R. Banga, "A hybrid approach for efficient and robust parameter estimation in biochemical pathways," *Biosystems*, vol. 83, no. 2-3, pp. 248 – 265, 2006.

[16] H. de Jong et.al., "Experimental and computational validation of models of fluorescent and luminescent reporter genes in bacteria," *BMC Systems Biology*, vol. 4, no. 1, p. 55, 2010.

[17] M. Kloetzer and C. Belta, "Reachability analysis of multi-affine systems," *Transactions of the Institute of Measurement and Control*, 2008, in press.

[18] C. Belta and L. Habets, "Controlling a class of nonlinear systems on rectangles," *IEEE Transactions on Automatic Control*, vol. 51, no. 11, pp. 1749–1759, 2006.

[19] G. Batt and C. Belta, "Model Checking Genetic Regulatory Networks with Applications to Synthetic Biology," CISE, Tech. Rep. 2006-IR-0030, 2006.

[20] F. Horn and R. Jackson, "General mass action kinetics," *Archive for Rational Mechanics and Analysis*, vol. 47, pp. 81–116, 1972.

[21] M. Feinberg, "Chemical reaction network structure and the stability of complex isothermal reactors i. the deficiency zero and the deficiency one theorems," *Chemical Engineering Science*, vol. 42, pp. 2229–2268, 1987.

[22] T. Mestl et.al., "A mathematical framework for describing and analysing gene regulatory networks," *Journal of Theoretical Biology*, vol. 176, no. 2, pp. 291–300, 1995.

[23] J. Fabriková, "Computational methods for discrete abstractions of ODE models," 2010, Master's Thesis, Masaryk University.

[24] H. Ma, F. Boogerd, and I. Goryanin, "Modelling nitrogen assimilation of escherichia coli at low ammonium concentration," *Journal of Biotechnology*, vol. 144, pp. 175–83, 2009.

[25] M. Swat, A. Kel, and H. Herzel, "Bifurcation analysis of the regulatory modules of the mammalian G1/S transition," *Bioinformatics*, vol. 20, no. 10, pp. 1506–1511, 2004.