Formal Analysis of Discrete-Time Piecewise Affine Systems

Boyan Yordanov and Calin Belta

Abstract—**In this technical note, we study temporal logic properties of trajectories of discrete-time piecewise affine (PWA) systems. Specifically, given a PWA system and a linear temporal logic formula over regions in its state space, we attempt to find the largest region of initial states from which all trajectories of the system satisfy the formula. Our method is based on the iterative computation and model checking of finite quotients. We illustrate our method by analyzing PWA models of two synthetic gene networks.**

Index Terms—**Abstraction, formal analysis, genetic networks, model checking, piecewise affine (PWA) systems, uncertain systems.**

I. Introduction

Temporal logics and model checking [3] are customarily used for specifying and verifying the correctness of digital circuits and computer programs. However, due to their resemblance to natural language, expressivity, and existence of off-the-shelf algorithms for model checking, temporal logics have the potential to impact several other areas. Examples include analysis of systems with continuous dynamics [4], control of linear systems from temporal logic specifications [5], [6], task specification and controller synthesis in mobile robotics [7], [8] and specification and analysis of qualitative behavior of genetic circuits [9], [10].

In this technical note, we focus on piecewise affine systems (PWA) that evolve along different discrete-time affine dynamics in different polytopic regions of the (continuous) state space. PWA systems are widely used as models in many areas. They can approximate nonlinear dynamics with arbitrary accuracy and are equivalent with other classes of hybrid systems [11]. In addition, there exist techniques for the identification of such models from experimental data, which include Bayesian methods, bounded-error procedures, clustering-based methods, mixed-integer programming, and algebraic geometric methods (see [12] for a review). We allow the parameters of the PWA systems to have polytopic uncertainty ranges, and consider specifications given as linear temporal logic (LTL) formulas over the polytopic regions in the state space of the system.

Unlike approaches that attempt to synthesize parameters for the system from a temporal logic specification [13], [14], in this work we assume that uncertainty is inherent in the system, and therefore the parameter ranges cannot be restricted further. We attempt to guarantee the satisfaction of a property by selecting appropriate initial states for the system. Specifically, given a PWA system, where parameters are possibly uncertain but known to belong to polytopic sets, and an LTL formula over regions in the state space of the system, we attempt to find the largest region, from which all trajectories of the system satisfy the property expressed by the formula, regardless of the parameter values. Our approach is based on the iterative construction and model checking of discrete abstractions in the form of finite transition systems [3].

This work can be seen in the context of literature focused on the construction of finite quotients of infinite systems (see [15] for an earlier review), and is closely related to [5], [16], [17]. The embedding of discrete-time systems into transition systems is inspired from [5], [16]. However, while the focus there is on characterizing the existence of bisimulation quotients or developing control strategies using such quotients for linear systems, in this work we focus on the computation and refinement of simulation quotients of PWA systems and consider an analysis problem. Unlike counterexample guided refinement [17], our approach can target the refinement to specific states. The analysis of PWA systems for properties such as invariance and reachability has been previously considered in literature focused on controlling PWA systems [18], [19]. In this technical note, we significantly expand this class of properties by allowing for arbitrary LTL specifications (invariance and reachability are particular examples of LTL properties).

The method developed in this technical note has been implemented as the MATLAB software tool Formal Analysis of Piecewise Affine Systems (FAPAS) [1], [2] and is freely available for download at http://hyness.bu.edu/software. To illustrate our method, we computed the basins of attraction for the equilibria of a PWA model of a two-gene network inspired by the Genetic Toggle Switch system [20]. In addition, we computed initial regions guaranteeing oscillations for a PWA model of a three-gene network inspired by the Repressilator system [21]. From this perspective, this technical note relates to [22], [23], where temporal logics are used to specify properties of biomolecular networks. These works aim at checking whether a system satisfies dynamical properties for given (sets of) initial conditions. In contrast, we search for the largest set of initial conditions for which the given properties are satisfied.

II. Definitions and Preliminaries

Given a set Q, we use $|Q|$ and 2^Q to denote its cardinality and powerset, respectively.

Definition 1: A transition system is a tuple $T = (Q, \to, O, o)$, where Q is a (possibly infinite) set of states, $\to \subseteq Q \times Q$ is a transition relation, O is a finite set of observations, and $o : Q \to O$ is an observation map.

A transition $(x, x') \in \to$ is also denoted by $x \to x'$. Transition system T is *finite* if its set of states Q is finite and *infinite* otherwise, *deterministic* if, for all $x \in Q$, there exists at most one $x' \in Q$ such that $(x, x') \in \to$, and *non-blocking* if, for every state $x \in Q$, there exists $x' \in Q$ such that $(x, x') \in \to$. In this technical note only non-blocking transition systems are considered.

A *trajectory* of T starting from state $x_0 \in Q$ is an infinite sequence $x_0 x_1 x_2 \ldots$ with the property that $x_i \in Q$, and $(x_i, x_{i+1}) \in \to$, for all $i \geq 0$. A trajectory $x_0 x_1 x_2 \ldots$ defines a *word* $w_0 w_1 w_2 \ldots$, where $w_i = o(x_i)$. The set of all words generated by the set of all trajectories starting at $x \in Q$ is called the *language* of T originating at x and is denoted by $\mathcal{L}_T(x)$. A subset $X \subseteq Q$ is called a *region* of T and the language of T originating at X is $\mathcal{L}_T(X) = \bigcup_{x \in X} \mathcal{L}_T(x)$. The language of T is defined as $\mathcal{L}_T(Q)$, which for simplicity is denoted as $\mathcal{L}_T$. For an arbitrary region X, we define the set of states $Pre_T(X)$ that reach X in one step as

$$Pre_T(X) = \{x \in Q | \exists x' \in X, x \to x'\}. \tag{1}$$

Manuscript received August 06, 2008; revised October 08, 2009; accepted August 10, 2010. Date of publication September 02, 2010; date of current version December 02, 2010. This paper was presented in part at the European Control Conference (ECC), 2001 and at the American Control Conference (ACC), 2008. This work was supported in part by NSF 0834260, NSF CAREER 0447721, and AFOSR FA9550-09-1-020 at Boston University. Recommended by Associate Editor J. Lygeros.

B. Yordanov is with the Department of Biomedical Engineering, Boston University, Boston, MA 02446 USA (e-mail: yordanov@bu.edu).

C. Belta is with the Division of Systems Engineering, Department of Mechanical Engineering, and the Bioinformatics Graduate Program, Boston University, Boston, MA 02446 USA (e-mail: cbelta@bu.edu).

Color versions of one or more of the figures in this technical note are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2010.2072530

Similarly, we define the set of states $Post_T(X)$ that can be reached from X in one step as

$$Post_T(X) = \{x' \in Q | \exists x \in X, x \to x'\}. \tag{2}$$

The observation map o of a transition system T induces an equivalence relation $\sim$ over the set of states Q. We say that states $x_1, x_2 \in Q$ are equivalent (written as $x_1 \sim x_2$) if and only if $o(x_1) = o(x_2)$. The equivalence relation naturally induces a *quotient transition system* $T/_\sim = (Q/_\sim, \to_\sim, O, o_\sim)$. $Q/_\sim$ is the quotient space (the set of all equivalence classes). Given an equivalence class $S \in Q/_\sim$, we denote the set of all equivalent states in that class by $con(S) \subseteq Q$ (con stands for concretization map). If $\mathbb{S} \in 2^{Q/_\sim}$ is a region of $T/_\sim$, then $con(\mathbb{S}) = \bigcup_{S \in \mathbb{S}} con(S)$ is a region of T. Since all states $x \in Q$ in an equivalence class $S \in Q/_\sim$ have the same observation, $o_\sim(S)$ is well defined and given by $o_\sim(S) = o(x), x \in con(S)$. The transition relation $\to_\sim$ is defined as follows: for $S_1, S_2 \in Q/_\sim$, $S_1 \to_\sim S_2$ if and only if there exist $x_1 \in con(S_1)$ and $x_2 \in con(S_2)$ such that $x_1 \to x_2$. It is easy to see that for all $S \in Q/_\sim$

$$\mathcal{L}_T(con(S)) \subseteq \mathcal{L}_{T/_\sim}(S). \tag{3}$$

The quotient transition system $T/_\sim$ is said to *simulate* the original system T.

Definition 2: The equivalence relation $\sim$ induced by the observation map o is a bisimulation of a transition system $T = (Q, \to, O, o)$ if, for all states $x_1, x_2 \in Q$, if $x_1 \sim x_2$ and $x_1 \to x_1'$, then there exist $x_2' \in Q$ such that $x_2 \to x_2'$ and $x_1' \sim x_2'$.

If $\sim$ is a bisimulation, then the quotient transition system $T/_\sim$ is called a *bisimulation quotient* of T, and the transition systems T and $T/_\sim$ are called *bisimilar*. An immediate consequence of bisimulation is language equivalence, i.e., for all $S \in Q/_\sim$tac-shrt4

$$\mathcal{L}_T(con(S)) = \mathcal{L}_{T/_\sim}(S). \tag{4}$$

Using the $Pre_T()$ operator defined in (1), a characterization of bisimulation can be given as follows: the equivalence relation $\sim$ is a bisimulation if and only if for all equivalence classes $S' \in Q/_\sim$, $Pre_T(con(S'))$ is either empty or a finite union of equivalence classes. Equivalently, the bisimulation property (Def. 2) is violated at $S \in Q/_\sim$ if there exists a state $S' \in Q/_\sim$, such that tac-shrt4

$$\emptyset \subset con(S) \cap Pre_T(con(S')) \subset con(S). \tag{5}$$

This leads to an iterative procedure for the construction of the coarsest bisimulation $\sim$, known as the "bisimulation algorithm" [24].

To specify temporal logic properties for system trajectories, in this technical note we use LTL formulas [3]. We use the standard notation for the Boolean operators (i.e., $\neg$ (negation), $\vee$ (disjunction), $\wedge$ (conjunction)) and the graphical notation for the temporal operators, e.g., $\bigcirc$ ("next"), $\mathcal{U}$ ("until"), $\Box$ ("always"), $\Diamond$ ("eventually"). Given a finite transition system $T = (Q, \to, O, o)$ and an LTL formula ϕ over O, an off-the-shelf model checker, such as NuSMV [25], can be used to check whether the language $\mathcal{L}_T(x)$ satisfies ϕ, for all $x \in Q$. For a region $X \subseteq Q$, we write $T(X) \models \phi$ if all the words from $\mathcal{L}_T(X)$ satisfy ϕ. In this technical note, we use our in-house implementation of LTL model checking [6] (denoted by $\mathrm{MODEL\text{-}CHECK}()$) in order to separate the translation of a formula ϕ to the accepting Büchi automaton [26] from the rest of the computation involved in model checking. This allows for a more efficient implementation of the iterative procedure described in Algorithm 1.

Algorithm 1 Given an infinite transition system T and a LTL formula ϕ, find $con\left(X^{\phi}_{\hat{T}/_\sim}\right) \subseteq X^{\phi}_T$

```
Construct $T/_\sim$
Initialize $\hat{T}/_\sim := T/_\sim$
Initialize $X^{\phi}_{\hat{T}/_\sim} := \mathrm{MODEL\text{-}CHECK}\ (\hat{T}/_\sim, \hat{Q}/_\sim, \phi)$
Initialize $X^{\neg\phi}_{\hat{T}/_\sim} := \mathrm{MODEL\text{-}CHECK}\ (\hat{T}/_\sim, \hat{Q}/_\sim, \neg\phi)$
repeat
$\mathbb{S}_r := \left\{S \in \hat{Q}/_\sim | S \text{ is large enough}, S \notin X^{\phi}_{\hat{T}/_\sim}, S \notin X^{\neg\phi}_{\hat{T}/_\sim}\right\}$
for each $S \in \mathbb{S}_r$ do
$\hat{T}/_\sim := \mathrm{REFINE}(\hat{T}/_\sim, S)$
end if
$\hat{\mathbb{S}}_r := \hat{Q}/_\sim \setminus \left(X^{\phi}_{\hat{T}/_\sim} \cup X^{\neg\phi}_{\hat{T}/_\sim}\right)$
$X^{\phi}_{\hat{T}/_\sim} := X^{\phi}_{\hat{T}/_\sim} \cup \mathrm{MODEL\text{-}CHECK}\ (\hat{T}/_\sim, \hat{\mathbb{S}}_r, \phi)$
$X^{\neg\phi}_{\hat{T}/_\sim} := X^{\neg\phi}_{\hat{T}/_\sim} \cup \mathrm{MODEL\text{-}CHECK}\ (\hat{T}/_\sim, \hat{\mathbb{S}}_r, \neg\phi)$
until $\mathbb{S}_r = \emptyset$
return $con\left(X^{\phi}_{\hat{T}/_\sim}\right)$
```

Given a region $X \subseteq Q$, $\mathrm{MODEL\text{-}CHECK}\ (T, X, \phi) = \{x \in X | T(x) \models \phi\}$ is the subset of X satisfying the formula. Let

$$X^{\phi}_T = \{x \in Q | T(x) \models \phi\}. \tag{6}$$

Note that $X^{\phi}_T = \mathrm{MODEL\text{-}CHECK}(T, Q, \phi)$ and if $x \notin X^{\phi}_T$, then there exists a word in $\mathcal{L}_T(x)$ that violates ϕ. Therefore, X^{ϕ}_T is the largest region of T satisfying ϕ.

If $T/_\sim$ is a quotient of T, then for any equivalence class $S \in Q/_\sim$ and formula ϕ, we have

$$T/_\sim(S) \models \phi \Rightarrow T(con(S)) \models \phi. \tag{7}$$

In addition, if $\sim$ is a bisimulation, then

$$T/_\sim(S) \models \phi \Longleftrightarrow T(con(S)) \models \phi. \tag{8}$$

Properties (7) and (8) (which follow immediately from (3) and (4)) allow one to model check finite quotients and extend the results to the (possibly infinite) original transition system.

III. Problem Formulation and Approach

Let $\mathcal{X}_l, l \in L$ be a set of open polytopes in $\mathbb{R}^N$, where L is a finite index set, such that $\mathcal{X}_{l_1} \cap \mathcal{X}_{l_2} = \emptyset$ for all $l_1, l_2 \in L$, $l_1 \neq l_2$ and $\mathcal{X} = \bigcup_{l \in L} cl(\mathcal{X}_l)$ is a closed full-dimensional polytope in $\mathbb{R}^N$ ($cl(\mathcal{X}_l)$ denotes the closure of set $\mathcal{X}_l$). A discrete-time PWA system with polytopic parameter uncertainty is defined as:

$$x_{k+1} = A_l x_k + b_l, x_k \in \mathcal{X}_l, l \in L, k = 0, 1, 2, \ldots \tag{9}$$

where parameters A_l and b_l are uncertain, but known to belong to polytopic uncertainty sets $\mathcal{P}^A_l \subset \mathbb{R}^{N \times N}$ and $\mathcal{P}^b_l \subset \mathbb{R}^N$, respectively.

We are interested in properties of (9) specified in terms of the polytopes from its definition. Informally, the semantics of system (9) can be understood in the following sense: a trajectory $x_0x_1x_2\ldots$ starting at $x_0 \in \mathcal{X}_{l_0}$, $l_0 \in L$ can be obtained by arbitrarily selecting parameters $A_{l_0} \in \mathcal{P}_{l_0}^A$, $b_{l_0} \in \mathcal{P}_{l_0}^b$, applying the affine map of (9) to compute x_1, finding $l_1 \in L$ such that $x_1 \in \mathcal{X}_{l_1}$, and repeating this procedure for each subsequent step. A trajectory produces an infinite word $l_0l_1l_2\ldots$, where $l_i \in L$ is the index of the polytope visited at step i (i.e., $x_i \in \mathcal{X}_{l_i}$). An LTL formula over L can then be interpreted over trajectories of the system (see Section II).

In general, it is possible that trajectories of (9) leave polytope $\mathcal{X}$. While we are not interested in such trajectories, we capture them by defining an additional observation Out, and trivial dynamics $x_{k+1} = x_k$ when $x_k \notin \mathcal{X}$ (e.g., a trajectory $x_0x_1x_2x_3\ldots$ satisfying $x_0, x_1 \in \mathcal{X}_{l_0}, x_2 \in \mathcal{X}_{l_1}$ for some $l_0, l_1 \in L$ and $x_3 \notin \mathcal{X}$ produces a word $l_0l_0l_1Out\ldots$, where Out is repeated infinitely). As it will become clear soon, we will be able to specify and forbid such behavior.

In the following, we formalize the satisfaction of LTL formulas by trajectories of (9) through an embedding into a transition system.

Definition 3: The embedding transition system $T_e = (Q_e, \rightarrow_e, O_e, o_e)$ for the PWA system from (9) is defined as:

- $Q_e = \mathbb{R}^N$;
- $(x, x') \in \rightarrow_e$ if and only if $x \notin \mathcal{X}$ and $x = x'$, or there exist $l \in L$ such that $x \in \mathcal{X}_l$ and there exist $A_l \in \mathcal{P}_l^A$, $b_l \in \mathcal{P}_l^b$ such that $x' = A_lx + b_l$;
- $O_e = L \cup \{Out\}$;
- $o_e(x) = l$ if and only if there exist $l \in L$ such that $x \in \mathcal{X}_l$ and $o_e(x) = Out$ otherwise.

Note that the embedding T_e has an infinite number of states and is always non-blocking. Furthermore, if the parameters of the PWA system are fixed, T_e is deterministic.

Definition 4: Given a subset $X \subseteq Q_e$, we say that all trajectories of system (9) originating in X satisfy formula ϕ if and only if $T_e(X)$ satisfies ϕ.

Now, we can formulate the main problem considered in this technical note:

Problem 1: Given a discrete-time PWA system (9) and an LTL formula ϕ over L, find the largest region of initial states, from which all trajectories of the system satisfy ϕ, while always remaining within $\mathcal{X}$.

The solution to Problem 1 amounts to the computation of $X_{T_e}^{\phi'}$ (see (6)), where $\phi' = \phi \wedge \Box\neg Out$. This guarantees that all trajectories originating there satisfy ϕ and always remain within $\mathcal{X}$. In addition, there exist trajectories originating in all states $x \notin X_{T_e}^{\phi'}$ that either violate ϕ or leave $\mathcal{X}$ (i.e., $X_{T_e}^{\phi'}$ is largest satisfying region). Since T_e has an infinite number of states, it cannot be analyzed directly. Our approach involves the construction, iterative refinement, and model checking of finite quotients simulating T_e (see Section II). Algorithms for iterative refinement and model checking are proposed in Section IV, where the results are valid in general for any transition system, while the construction of the quotients and the implementation of the refinement procedure for T_e are discussed in Section V. We consider separately the case when A_l, $l \in L$ are fixed (i.e., $\mathcal{P}_l^A$ are singletons) and the case when all system parameters are fixed (i.e., $\mathcal{P}_l^A$ and $\mathcal{P}_l^b$ are singletons for all $l \in L$). Those particular cases are important in practice since they correspond to PWA systems subjected to additive uncertainty only or no uncertainty at all and we show that these additional constraints can be exploited. As it will become clear later, our approach to Problem 1 is conservative, in the sense that, we can only “try” to find the satisfying region $X_{T_e}^{\phi'}$ but, in general, we can only guarantee to obtain subsets of it.

Remark 1: The two assumptions from the formulation of Problem 1 seem restrictive. First, we capture only the reachability of open full dimensional polytopes in the semantics of the embedding. Arguably, this is enough for practical purposes, since only sets of measure zero are disregarded, and it is unreasonable to assume that equality constraints can be detected in real-world applications. There are two situations in which the boundaries can affect the semantics of the trajectories non-trivially: 1) when trajectories originate and remain in such sets for all times, and 2) when trajectories start in open polytopes and then “vanish” in the boundaries. For both these situations, the system dynamics and the polytopes need to satisfy special conditions, which can be easily derived, but are omitted due to space constraints. Second, the specification is given over the indexes $l \in L$ of the polytopes $\mathcal{X}_l$ from the system definition. However, arbitrary linear inequalities can be accommodated simply by refining the polytopic partition–the resulting PWA will have some polytopes with identical dynamics.

IV. Formal Analysis of Infinite Transition Systems

The embedding transition system T_e (Def. 3) is infinite and therefore $X_{T_e}^{\phi}$ cannot be computed directly. In this section, we consider the following problem:

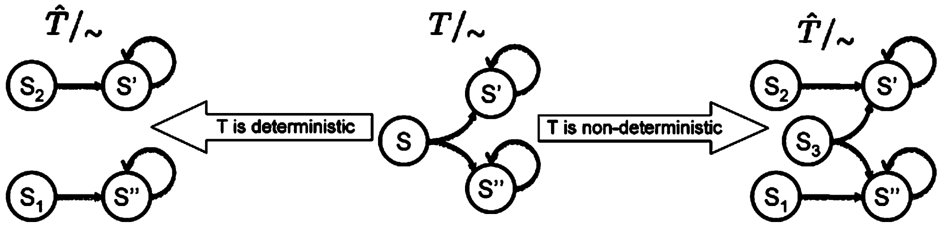
Problem 2: Given an infinite transition system T (Def. 1) and an LTL formula ϕ over its set of observations O, find X_T^{ϕ} (6).

We assume that, given the equivalence relation $\sim$ (see Section II), the finite quotient $T/_\sim$ is computable (its computation for T_e is discussed in Section V). Then, $X_{T/_\sim}^{\phi}$ can be computed by model checking and from (7) it follows that $con(X_{T/_\sim}^{\phi})$ is a satisfying region in T but, in general, it is not the largest satisfying region (i.e., $con(X_{T/_\sim}^{\phi}) \subseteq X_T^{\phi}$). The most intuitive solution to Problem 2 would then be to apply the bisimulation algorithm (see Section II) and refine the quotient $T/_\sim$ to make it bisimilar with T. In this case, following from (8), $con(X_{T/_\sim}^{\phi}) = X_T^{\phi}$ is the solution to Problem 2. However, an infinite transition system does not always have a finite bisimulation quotient, so such a procedure would only work for very particular cases.

A. Iterative Model Checking

Even though the quotient $T/_\sim$ cannot always be refined enough to be bisimilar with T, region $X_{T/_\sim}^{\phi}$ can be computed at each step of a refinement procedure. Then, $con(X_{T/_\sim}^{\phi})$ can provide a conservative solution to Problem 2, which can be improved by additional refinement. If $\hat{T}/_\sim = (\hat{Q}/_\sim, \hat{\rightarrow}_\sim, O, \hat{o}_\sim)$ is the quotient after some refinement has been performed, we have $\mathcal{L}_T \subseteq \mathcal{L}_{\hat{T}/_\sim} \subseteq \mathcal{L}_{T/_\sim}$ and $con(X_{T/_\sim}^{\phi}) \subseteq con(X_{\hat{T}/_\sim}^{\phi}) \subseteq X_T^{\phi}$. A related idea was used in [27] for verification from formulas in the universal fragment ACTL of CTL. Such approaches face computational challenges, due to the possible explosion in the number of states of $\hat{T}/_\sim$ as refinement progresses.

Our methods aim at refining and model checking the quotient only at states where this can improve the solution (i.e., increase $X_{\hat{T}/_\sim}^{\phi}$). Refinement of any state $S \in X_{\hat{T}/_\sim}^{\phi}$, is unnecessary, since all trajectories originating there satisfy the formula. Similarly, the set $X_{\hat{T}/_\sim}^{\neg\phi}$ can be computed and refinement of any state $S \in X_{\hat{T}/_\sim}^{\neg\phi}$ is also unnecessary, since only trajectories violating the formula originate there. Refinement of the quotient at any state does not change the satisfaction of the formula at S, where $S \in X_{\hat{T}/_\sim}^{\phi}$ or $S \in X_{\hat{T}/_\sim}^{\neg\phi}$ and, therefore, once a state has been identified as satisfying the formula or its negation it is no longer considered for refinement or model checking. This leads to a procedure that iteratively refines the quotient $\hat{T}/_\sim$ and possibly expands $X_{\hat{T}/_\sim}^{\phi}$ and $X_{\hat{T}/_\sim}^{\neg\phi}$ at each iteration (Algorithm 1). It is important to note that, by using our implementation (MODEL-CHECK), a significant part of the model-checking computation is performed only once and stored for each iterative step.



(a) (b) (c)

Fig. 1. Application of $\mathrm{REFINE}(T/_\sim, S)$ to the quotient $T/_\sim$, shown in (b), where $Post_{T/_\sim}(S) = \{S', S''\}$. The result for a deterministic T is shown in (a), while the result for a nondeterministic T is shown in (c).

The set $\mathbb{S}_r \subseteq Q/_\sim$, computed in Algorithm 1, contains the states of the quotient, where refinement should be targeted and refinement of any state $S \notin \mathbb{S}_r$ will not expand $X^{\phi}_{\hat{T}/_\sim}$. In addition, $\mathbb{S}_r$ contains only states that are "large enough" to undergo refinement (see Section V for a description of such a measure for T_e), guaranteeing that the algorithm will terminate if a sufficient number of iterations is performed. In order to implement Algorithm 1, the finite quotient $T/_\sim$ must be computable, the possibly infinite $con\left(X^{\phi}_{\hat{T}/_\sim}\right)$ must be represented and a refinement procedure for $T/_\sim$ that can be applied locally at a state $S \in Q/_\sim$ is required.

B. Quotient Refinement

If $T/_\sim$ is a finite bisimulation quotient, then an exact solution to Problem 2 can be obtained by applying Algorithm 1. Motivated by this, we formulate a refinement procedure $\mathrm{REFINE}()$ (Algorithm 2) inspired by the bisimulation algorithm (see Section II). Unlike the bisimulation algorithm, which refines the equivalence relation $\sim$ globally, $\mathrm{REFINE}(T/_\sim, S)$, refines the quotient $T/_\sim$ locally at a state $S \in Q/_\sim$. This allows us to target refinement to specific states (as in Algorithm 1), while the quotient is updated instead of recomputed every time refinement is performed.

Algorithm 2 $\hat{T}/_\sim = \mathrm{REFINE}(T/\sim, S)$

```
Initialize $\mathbb{S}_r := \{S\}$
**while** there exist $S_r \in \mathbb{S}_r,\ S' \in Post_{T/_\sim}(S)$ such that
$\emptyset \subset con(S_r) \cap Pre_T(con(S')) \subset con(S_r)$ **do**
    Construct states $S_1, S_2$ such that:
    $con(S_1) := con(S_r) \cap Pre_T(con(S'))$
    $con(S_2) := con(S_r) \setminus Pre_T(con(S'))$
    $\mathbb{S}_r := (\mathbb{S}_r \setminus S_r) \cup \{S_1, S_2\}$
**end while**
update $\hat{\rightarrow}_\sim$ and $\hat{o}_\sim$
**return** $\hat{T}/_\sim$
```

$\mathrm{REFINE}(T/_\sim, S)$ partitions state S in such a way that all resulting subsets of S satisfy the bisimulation property (i.e., for all subsets of S there does not exist a state $S' \in Q/_\sim$ such that (5) is satisfied). It is easy to see that for any states $S,\ S' \in Q/_\sim$, $con(S) \cap Pre_T(con(S')) \neq \emptyset$ if and only if $S' \in Post_{T/_\sim}(S)$ (i.e., S' is reachable from S in $T/_\sim$). Then, all nonempty intersections $con(S) \cap_{S' \in \mathbb{S}'} Pre_T(con(S')) \setminus \cup_{S'' \in \mathbb{S}''} Pre_T(con(S''))$, where $\mathbb{S}' \in 2^{Post_{T/_\sim}(S)}$ and $\mathbb{S}'' = Post_{T/_\sim}(S) \setminus \mathbb{S}'$, provide a partition of S satisfying the bisimulation property. Therefore, applying $\mathrm{REFINE}(T/_\sim, S)$ results in at most $2^{|Post_{T/_\sim}(S)|}$ subsets. In the particular case when T is deterministic, one can easily show that given states $S,\ S',\ S'' \in Q/_\sim$ such that $S',\ S'' \in Post_{T/_\sim}(S)$, $con(S) \cap Pre_T(con(S')) \cap Pre_T(con(S'')) = \emptyset$. Then, $con(S) \cap Pre_T(con(S'))$ of all $S' \in Post_{T/_\sim}(S)$ provide a partition of S satisfying the bisimulation property and applying $\mathrm{REFINE}(T/_\sim, S)$ on the quotient of a deterministic system T results in at most $|Post_{T/_\sim}(S)|$ subsets.

When refinement is performed using the $Pre_T()$ operation, outgoing transitions of the newly formed states are implicitly induced. Given states $S,\ S' \in Q/_\sim$ such that $S' \in Post_{T/_\sim}(S)$, the subset $con(S) \cap Pre_T(con(S'))$ always has a transition to state S' (in fact, this is the only transition possible in the case when T is deterministic). Additionally, any subset of $con(S) \setminus Pre_T(con(S'))$ can never have a transition to state S'. In the particular case when state S has a self transition ($S \rightarrow_\sim S$), transitions from subset $con(S) \cap Pre_T(con(S))$ to all subsets of S resulting from its refinement are possible and must be recomputed (the computation of transitions between any two states is discussed in Section V). Incoming transitions from all states $S'' \in Pre_{T/_\sim}(S)$ reaching S to all newly formed states are also updated, which completes the construction of $\hat{\rightarrow}_\sim$. All subsets of a refined state inherit the observation of the parent and, therefore, $\hat{o}_\sim$ is easily updated.

So far, we have discussed a refinement strategy inspired by the bisimulation algorithm and, therefore, relying on the computation of the $Pre_T()$ operation (see Fig. 1 for an example). If $Pre_T()$ is not computable, any refinement strategy can be used for the function REFINE in Algorithm 1 with the hope that the smaller regions produced at each step separate satisfying and violating trajectories. In this case, when refinement is performed at state S, outgoing transitions from newly formed states are not implicitly induced and must be recomputed but only target states in the set $Post_{T/_\sim}(S)$ (instead of the entire $Q/_\sim$) need to be considered.

V. Formal Analysis of PWA Systems

Through the embedding of the PWA system (9) into an infinite transition system T_e (Def. 3), we reduced Problem 1 to Problem 2. Based on the assumption that finite quotients can be constructed, we proposed an algorithm to solve Problem 2 in Section IV. In this section, we discuss the construction of the quotients, and the implementation of the algorithms from Section IV for T_e.

From the definitions of the equivalence relation $\sim$, induced by the observation map o (Section II) and T_e (Def. 3), the initial set of states $Q_e/_\sim$, of the finite quotient $T_e/_\sim$, is simply the set of observations $Q_e/_\sim = O_e = L \bigcup \{Out\}$ and the observation map is identity. Given a state $l \in Q_e/_\sim$, $l \neq Out$, $con(l) = \mathcal{X}_l$ is a polytope from the system definition (9). In order to finish the construction of the quotient, we need to find the set of transitions $\rightarrow_{e,\sim}$. By the definition of $\rightarrow_\sim$ (Section II) and (2), the transition relation $\rightarrow_{e,\sim}$ can be constructed if

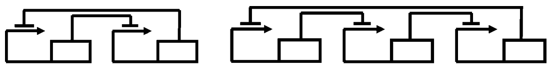

(a) (b)

Fig. 2. Schematic representations of the two genetic networks considered in this case study.

$Post_{T_e}()$ is computable. Explicitly, for any two equivalence classes l, $l' \in (Q_e/_\sim \setminus \{Out\})$, we have:

$$(l,l') \in \rightarrow_{e,\sim} \text{ if and only if } Post_{T_e}(\mathcal{X}_l) \cap \mathcal{X}_{l'} \neq \emptyset. \tag{10}$$

Similarly, given a state $l \in Q_e/_\sim, l \neq Out$, transitions to state Out can be assigned as

$$(l,Out) \in \rightarrow_{e,\sim} \text{ if and only if } Post_{T_e}(\mathcal{X}_l) \not\subseteq \mathcal{X} \tag{11}$$

and state Out only has a transition to itself (i.e., $(Out,Out) \in \rightarrow_{e,\sim}$).

In the particular case when the matrix component of the parameters of the PWA system (9) are fixed, (i.e., the sets $\mathcal{P}_l^A = A_l, l \in L$ are all singletons), given a polytope $\mathcal{X}_l, l \in L, Post_{T_e}(\mathcal{X}_l)$ is convex and can be computed exactly

$$Post_{T_e}(\mathcal{X}_l) = A_l\mathcal{X}_l \oplus \mathcal{P}_l^b \tag{12}$$

where $A_l\mathcal{X}_l$ is the image of the polytope $\mathcal{X}_l$ through the matrix A_l and "$\oplus$" stands for Minkowski (set) sum. Therefore, the set of transitions $\rightarrow_{e,\sim}$ can be computed using polyhedral operations and the finite quotient $T_e/_\sim$ can be constructed and used in Algorithm 1.

When the matrix component of the parameters is allowed to vary, given a polytope $\mathcal{X}_l$, $Post_{T_e}(\mathcal{X}_l)$ is not necessarily convex.

Proposition 1: Given a polytope $\mathcal{X}_l$, the smallest convex over-approximation of $Post_{T_e}(\mathcal{X}_l)$ can be computed as

$$\overline{Post_{T_e}(\mathcal{X}_l)} = hull\left\{Ax | A \in \mathcal{V}\left(\mathcal{P}_l^A\right), x \in \mathcal{V}(\mathcal{X}_l)\right\} \oplus \mathcal{P}_l^b \tag{13}$$

where $hull()$ and $\mathcal{V}()$ denote the convex hull and set of vertices, respectively.

A proof of Proposition 1 can be found in [2] and a related treatment in [28]. Using the over-approximation $\overline{Post_{T_e}(\mathcal{X}_l)}$, an over-approximation quotient $\overline{T_e/_\sim} = \{Q_e/_\sim, \overline{\rightarrow_{e,\sim}}, O, o_\sim\}$ can be constructed. Since $Post_{T_e}(\mathcal{X}_l) \subseteq \overline{Post_{T_e}(\mathcal{X}_l)}$ for all $l \in Q_e/_\sim$, we have $\rightarrow_{e,\sim} \subseteq \overline{\rightarrow_{e,\sim}}$, which leads to

$$\mathcal{L}_{T_e} \subseteq \mathcal{L}_{T_e/_\sim} \subseteq \mathcal{L}_{\overline{T_e/_\sim}} \Rightarrow X^{\phi}_{\overline{T_e/_\sim}} \subseteq X^{\phi}_{T_e/_\sim} \subseteq X^{\phi}_{T_e} \tag{14}$$

and, therefore, the over-approximation $\overline{T_e/_\sim}$ can be used in Algorithm 1 instead of $T_e/_\sim$ but the results become more conservative.

In order to implement the function $\mathrm{REFINE}()$ (Algorithm 2), given states $l_1, l_2 \in Q_e/_\sim$ such that $l_2 \in Post_{T_e/_\sim}(l_1)$, we need to be able to construct a state l', such that $con(l') = con(l_1) \cap Pre_{T_e}(con(l_2))$ or equivalently $con(l') = \mathcal{X}_{l_1} \cap Pre_{T_e}(\mathcal{X}_{l_2})$. If the matrix component of the parameters is fixed and $A_l, l \in L$ are all invertible, this intersection is computable as

$$\mathcal{X}_{l_1} \cap Pre_{T_e}(\mathcal{X}_{l_2}) = \mathcal{X}_{l_1} \cap A_{l_1}^{-1}\left(\mathcal{X}_{l_2} \oplus \left(-\mathcal{P}_{l_1}^b\right)\right). \tag{15}$$

Therefore, $\mathrm{REFINE}(T_e/_\sim, X)$ can be implemented using polyhedral operations and applied iteratively. As already discussed in Section IV, if $\mathcal{P}_l^b, l \in L$ are fixed then T_e is deterministic and refinement can be performed more efficiently.

Although a finite over-approximation quotient $\overline{T_e/_\sim}$ can be computed when the parameters of the system are uncertain, $Pre_{T_e}()$ might be nonconvex, even when applied to a convex set. In this case, we use a 2^N-tree inspired refinement approach, where each state is split along each dimension and transitions are recomputed using the over-approximation $\overline{Post_{T_e}()}$ in (10).

Finally, in order to implement Algorithm 1, we need to be able to decide if a state is "large enough" to undergo additional refinement. Given a state l, we compute the radius of the largest sphere inscribed in polytope $con(l)$ and apply the refinement procedure only if it is larger than a certain predefined limit ϵ.

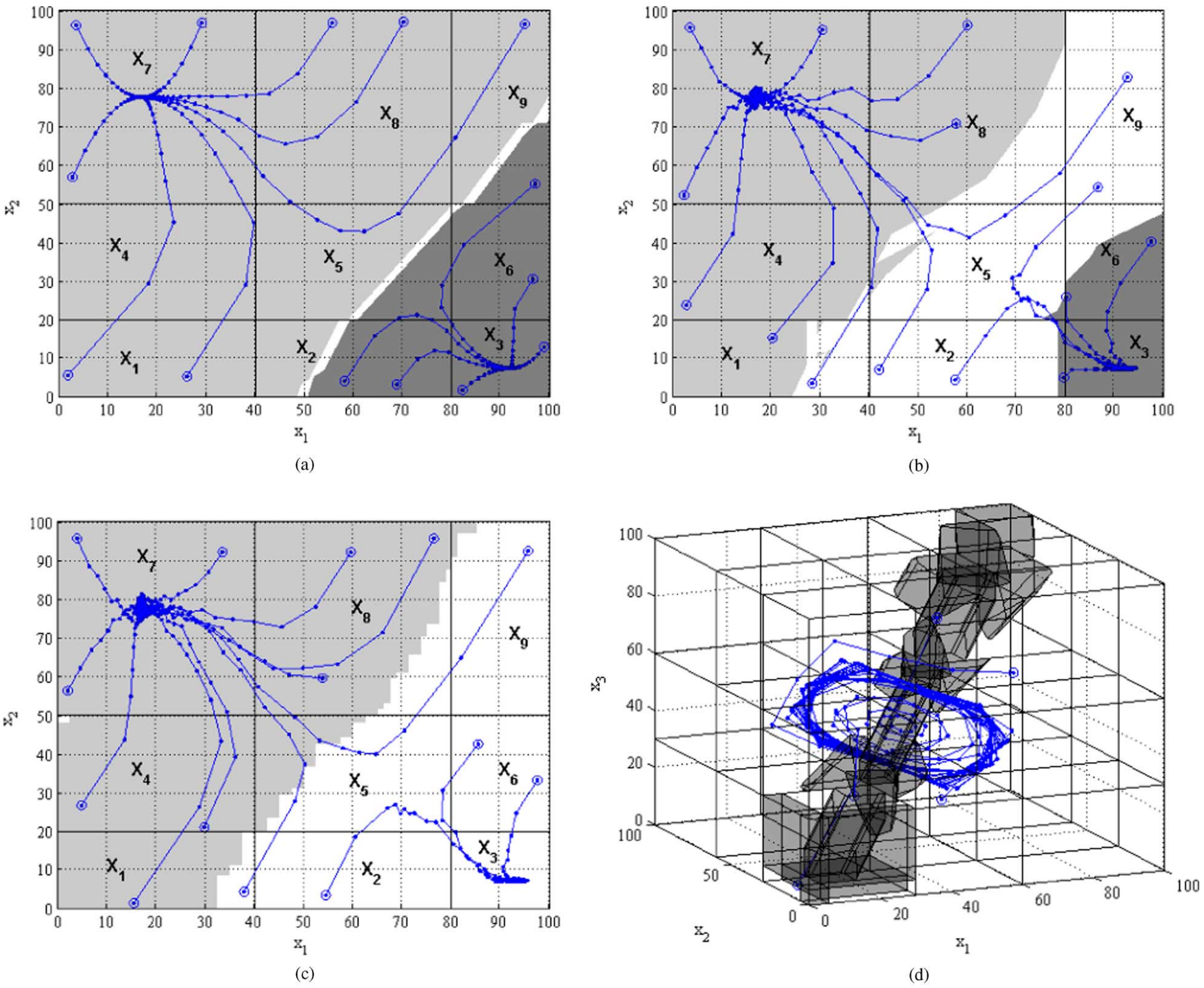
VI. Case Study: Analysis of PWA Models of Genetic Networks

In this section, we present results from the analysis of two PWA models inspired by the synthetic networks of repressor genes known as the Genetic Toggle Switch [20] [Fig. 2(a)] and the Repressilator [21] [Fig. 2(b)]. Gene regulation is modeled by ramp functions, which are PWA functions defined by two threshold values, inducing three regions of different dynamics. At low repressor concentrations (below threshold 1) the regulated gene is fully expressed, at high repressor concentrations (above threshold 2) expression is only basal and the response between the two thresholds is graded. The resulting PWA models capture the bistability [Fig. 3(a)] and oscillations [Fig. 3(d)] characteristic of the two systems[1].

The first system we consider [Fig. 2(a)] includes two mutually inhibiting genes and acts as a switch, allowing only one of the genes to be expressed depending on initial conditions [see the simulated trajectories in Fig. 3(a)]. Initially, we construct a fixed parameter PWA model with two state variables ($N = 2$) representing the concentrations of the proteins produced by the two genes. Gene regulation is captured through two ramp functions and, therefore, the model has a total of nine rectangular regions [denoted $\mathcal{X}_1, \ldots, \mathcal{X}_9$ with $L = \{1, 2, \ldots, 9\}$ in Fig. 3(a)]. Dynamics 3 and 7 have unique, asymptotically stable equilibria inside rectangles $\mathcal{X}_3$ and $\mathcal{X}_7$, respectively [see Fig. 3(a)]. We attempt to find regions of initial conditions guaranteeing that the system will settle in a specific equilibrium, thereby identifying the attractor regions for the two equilibria. By exploiting convexity properties of affine functions on polytopes, it can be shown that under the fixed parameters, $\mathcal{X}_3$ and $\mathcal{X}_7$ are invariants for dynamics 3 and 7. From this, we can immediately conclude that $\mathcal{X}_3$ and $\mathcal{X}_7$ are regions of attraction for the two equilibria. Therefore, our problem reduces to finding maximal regions satisfying LTL formulas $\phi_1 = \diamondsuit\square 3$ and $\phi_2 = \diamondsuit\square 7$ (note that the specifications are automatically extended to guarantee that trajectories of the system do not leave $\mathcal{X}$ as described in Section III). In other words, we want to find maximal sets of initial conditions guaranteeing that all trajectories of the system eventually reach regions $\mathcal{X}_3$ or $\mathcal{X}_7$, respectively, while always remaining within $\mathcal{X}$. The results of the analysis of the fixed parameter model are presented in Fig. 3(a).

Hyper-rectangular parameter uncertainty is then introduced in the model by allowing each component of the fixed parameters to vary in a small range. Results for an additive noise only model (i.e., where $\mathcal{P}_l^b$ are polytopes, while $\mathcal{P}_l^A$ are singletons for all $l \in L$) and uncertain parameters model (i.e, where both $\mathcal{P}_l^A$ and $\mathcal{P}_l^b$ are polytopes for all $l \in L$) are presented in Fig. 3(b) and (c), respectively. Because of the

[1] Due to space constraints the explicit PWA dynamics of the two systems are omitted but they are available at http://hyness.bu.edu/software



(a)

(b)

(c)

(d)

Fig. 3. (a)–(c) Simulated trajectories of the Genetic Toggle Switch PWA model go towards one of the two stable equilibria located inside regions $\mathcal{X}_3$ and $\mathcal{X}_7$. Trajectories originating in the dark gray and light gray regions are guaranteed to satisfy specification $\phi_1 = \Diamond\Box 3$ and $\phi_2 = \Diamond\Box 7$, respectively. (d) Simulated trajectories of the Repressilator PWA model oscillate, visiting regions where the concentrations of protein 1 are high. Trajectories originating everywhere, except the shaded region are guaranteed to satisfy specification ϕ_4. Initial conditions for all trajectories are marked with circles.

rectangular initial partition of the state space resulting from the definition of the PWA system, 2^N-trees are a suitable refinement strategy in the uncertain parameter case.

The second system we consider [Fig. 2(b)] includes three inhibiting genes and can be shown to produce oscillations [see simulated trajectories in Fig. 3(d)]. We construct a fixed parameter PWA model with three state variables ($N = 3$) and use three ramp functions to capture the effects of gene regulation, which results in a model with a total of 27 hyper-rectangular regions [see Fig. 3(d)]. We are interested in identifying regions of initial conditions from which all trajectories of the system oscillate, visiting regions where the concentrations of protein 1 are high. We consider the specification $\phi_4 = \Box(\Diamond(\phi_3 \wedge \Diamond\neg\phi_3))$, where ϕ_3 is satisfied whenever protein 1 concentrations are high (i.e., ϕ_3 is a disjunction of the labels of the rectangles on the right of the plane $x_1 = 60$). The results of the analysis are presented in Fig. 3(d).

VII. Complexity and Implementation

The algorithms presented in this technical note were implemented as a software tool for FAPAS, which is freely downloadable at http://hyness.bu.edu/software. The tool is built under MATLAB, and uses LTL2BA [26] for the conversion of an LTL formula to a Büchi automaton and the MPT toolbox [29] for polyhedral operations.

Our method involves model checking of the finite quotient $T_e/_\sim$ at each step of the iterative procedure. Even though the worst case complexity of LTL model checking is exponential in the size of the formula, this upper limit is rarely reached in practice. In addition, as already mentioned, we use an in-house model checker, which allows us to model check $T_e/_\sim$ from specific states only and perform computation (such as the construction of Büchi automata) only once instead of recomputing at each step. The construction and refinement of finite quotients used in our approach is based on polyhedral operations, which also have an exponential upper bound. Therefore, the applicability of the method depends on controlling the number of states as refinement progresses. When applied to a state S, the refinement procedure $\mathrm{REFINE}(T_e/_\sim, S)$ can, in general, produce a maximum of 2^k subsets, where $k = |Post_{T_e/_\sim}(S)|$ is the number of states reachable from S. In the particular case when the parameters of the PWA system are fixed, only k subsets can be produced. To limit the explosion in the number of states in the quotient, we only refine states when this can improve the solution. Even so, due to its inherent complexity, this method is not suitable for the analysis of systems in high dimensions or when many iterations are required to find a solution. As expected, the method performs best if large portions of the state space can be characterized as satisfying the formula or its negation during earlier iterations.

For the Genetic Toggle Switch ($N = 2$) presented in Section VI the computation required under 20 sec for the fixed parameter model and under 10 min for all the uncertain parameter ones, where the limit on refinement was set to $\epsilon = 1$ and $\epsilon = 5$. For the Repressilator ($N = 3$) the computation required under 20 min where $\epsilon = 5$. All computation was performed on a 3.4 GHz machine with 1 GB of memory.

It is important to note that some specifications (such as ϕ_1 and ϕ_2 in Section VI) can be formulated as invariance and reachability properties and checked using more efficient tools [18], [19], [29]. However, such an approach does not apply to general LTL specifications (such as ϕ_4 in Section VI).

References

[1] B. Yordanov, C. Belta, and G. Batt, "Model checking discrete time piesewise affine systems: Application to gene networks," in *Proc. Eur. Control Conf. (ECC)*, Kos, Greece, 2007, [CD ROM].

[2] B. Yordanov and C. Belta, "Formal analysis of piecewise affine systems under parameter uncertainty with application to gene networks," in *Proc. Amer. Control Conf.*, 2008, pp. 2767–2772.

[3] E. M. Clarke, D. Peled, and O. Grumberg, *Model Checking*. Cambridge, MA: MIT Press, 1999.

[4] J. M. Davoren, V. Coulthard, N. Markey, and T. Moor, "Non-deterministic temporal logics for general flow systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, R. Alur and G. J. Pappas, Eds. Berlin/Heidelberg, Germany: Springer, 2004, vol. 2993, pp. 107–121.

[5] P. Tabuada and G. Pappas, "Linear time logic control of discrete-time linear systems," *IEEE Trans. Autom. Control*, vol. 51, no. 12, pp. 1862–1877, Dec. 2006.

[6] M. Kloetzer and C. Belta, "A fully automated framework for control of linear systems from temporal logic specifications," *IEEE Trans. Autom. Control*, vol. 53, no. 1, pp. 287–297, Feb. 2008.

[7] S. Loizou and K. Kyriakopoulos, "Automatic synthesis of multi-agent motion tasks based on LTL specifications," in *Proc. 43rd IEEE Conf. Decision Control (CDC).*, 14–17, 2004, vol. 1, pp. 153–158.

[8] G. Fainekos, H. Kress-Gazit, and G. Pappas, "Hybrid controllers for path planning: A temporal logic approach," in *Proc. 44th IEEE Conf. Decision Control (CDC-ECC'05)*, 2005, pp. 4885–4890.

[9] M. Antoniotti, F. Park, A. Policriti, N. Ugel, and B. Mishra, "Foundations of a query and simulation system for the modeling of biochemical and biological processes," in *Proc. Pacific Symp. Biocomp. (PSB'03)*, 2003, pp. 116–127.

[10] G. Batt, D. Ropers, H. de Jong, J. Geiselmann, R. Mateescu, M. Page, and D. Schneider, "Validation of qualitative models of genetic regulatory networks by model checking : Analysis of the nutritional stress response in escherichia coli," *Bioinformatics*, vol. 21, no. 1, pp. I19–I28, 2005.

[11] W. P. M. H. Heemels, B. D. Schutter, and A. Bemporad, "Equivalence of hybrid dynamical models," *Automatica*, vol. 37, no. 7, pp. 1085–1091, 2001.

[12] A. L. Juloski, W. Heemels, G. Ferrari-Trecate, R. Vidal, S. Paoletti, and J. Niessen, "Comparison of four procedures for the identification of hybrid systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, M. Morari and L. Thiele, Eds. Berlin/Heidelberg, Germany: Springer, 2005, vol. 3414, pp. 354–369.

[13] G. Frehse, S. Jha, and B. Krogh, "A counterexample-guided approach to parameter synthesis for linear hybrid automata," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, M. Egerstedt and B. Mishra, Eds. Berlin/Heidelberg, Germany: Springer, 2008, vol. 4981, pp. 187–200.

[14] B. Yordanov and C. Belta, "Parameter synthesis for piecewise affine systems from temporal logic specifications," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, M. Egerstedt and B. Mishra, Eds. Berlin/Heidelberg, Germany: Springer, 2008, vol. 4981, pp. 542–555.

[15] R. Alur, T. A. Henzinger, G. Lafferriere, and G. J. Pappas, "Discrete abstractions of hybrid systems," *Proc. IEEE*, vol. 88, no. 7, pp. 971–984, Jul. 2000.

[16] G. J. Pappas, "Bisimilar linear systems," *Automatica*, vol. 39, no. 12, pp. 2035–2047, 2003.

[17] E. Clarke, A. Fehnker, Z. Han, B. Krogh, J. Ouaknine, O. Stursberg, and M. Theobald, "Abstraction and counterexample-guided refinement in model checking of hybrid systems," *Int. J. Found. Comput. Sci.*, vol. 14, no. 4, pp. 583–604, 2003.

[18] A. Bemporad, L. Giovanardi, and F. Torrisi, "Performance driven reachability analysis for optimal scheduling and control of hybrid systems," in *Proc. 39th IEEE Conf. Decision Control*, 2000, vol. 1, pp. 969–974.

[19] P. Grieder, "Efficient Computation of Feedback Controllers for Constrained Systems," Ph.D. dissertation, ETH Zürich, Zürich, Switzerland, 2004.

[20] T. Gardner, C. Cantor, and J. Collins, "Construction of a genetic toggle switch in escherichia coli," *Nature*, vol. 403, pp. 339–342, 2000.

[21] M. Elowitz and S. Leibler, "A synthetic oscillatory network of transcriptional regulators," *Nature*, vol. 403, pp. 335–338, 2000.

[22] G. Bernot, J.-P. Comet, A. Richard, and J. Guespin, "Application of formal methods to biological regulatory networks: Extending thomas' asynchronous logical approach with temporal logic," *J. Theor. Biol.*, vol. 229, no. 3, pp. 339–347, 2004.

[23] N. Chabrier-Rivier, M. Chiaverini, V. Danos, F. Fages, and V. Schächter, "Modeling and querying biomolecular interaction networks," *Theor. Comput. Sci.*, vol. 325, no. 1, pp. 25–44, 2004.

[24] A. Bouajjani, J.-C. Fernandez, and N. Halbwachs, "Minimal model generation," in *Computer-Aided Verification*, ser. Lecture Notes in Computer Science, E. Clarke and R. Kurshan, Eds. Berlin/Heidelberg, Germany: Springer, 1991, vol. 531, pp. 197–203.

[25] A. Cimatti, E. Clarke, E. Giunchiglia, F. Giunchiglia, M. Pistore, M. Roveri, R. Sebastiani, and A. Tacchella, "NuSMV 2: An opensource tool for symbolic model checking," in *Computer Aided Verification*, ser. Lecture Notes in Computer Science, E. Brinksma and K. Larsen, Eds. Berlin/Heidelberg, Germany: Springer, 2002, vol. 2404, pp. 241–268.

[26] P. Gastin and D. Oddoux, "Fast LTL to Büchi automata translation," in *Computer Aided Verification*, ser. Lecture Notes in Computer Science, G. Berry, H. Comon, and A. Finkel, Eds. Berlin/Heidelberg, Germany: Springer, 2001, vol. 2102, pp. 53–65.

[27] A. Chutinan and B. H. Krogh, "Verification of infinite-state dynamic systems using approximate quotient transition systems," *IEEE Trans. Autom. Control*, vol. 46, no. 9, pp. 1401–1410, Sep. 2001.

[28] B. Barmish and J. Sankaran, "The propagation of parametric uncertainty via polytopes," *IEEE Trans. Autom. Control*, vol. AC-24, no. 2, pp. 249–346, Apr. 1979.

[29] M. Kvasnica, P. Grieder, and M. Baotić, "Multi-Parametric Toolbox (MPT)," Tech. Rep., 2004 [Online]. Available: http://control.ee.ethz.ch/mpt/