1

Stability of Hybrid Systems and Related Questions from Systems Biology*

(*In honor of Professor Pravin Varaiya on his 65th birthday*)

C. Piazza[1] and B. Mishra[2]

[1] Dept. of Mathematics and Computer Science, University of Udine, Udine, Italy
[2] Courant Institute of Mathematical Science and School of Medicine, NYU, New York, U.S.A.
Correspondence to: piazza@dimi.uniud.it, mishra@cs.nyu.edu

1.1 Prologue

Pravin Varaiya's research career is marked by an ever-expanding breadth of interests starting with classical areas of electrical and communication engineering, but frequently intersecting with fields as far apart as highway traffic systems, game theory and economics. Indirectly through his students, postdoctoral fellows, "mentees" and even others, who came in contact with him only in chance encounters, his intellectual reach has gone much further.

From the mid nineties to present, a research theme that Pravin Varaiya has explored deeply concerns with "hybrid automata." These are systems describing a discrete program in a continuous environment. The best natural example that comes to mind would be a description of developmental stages of an organism embedded inside an environment composed of a variety of biological macromolecules (DNA, RNA and protein) synthesizing, duplicating, modulating and degrading each other in a complex manner. The basic developmental program interacts with the environment through injuries, infection, immune interactions, mutations, diseases, aging and evolutionary processes. While unfortunately the asymptotic destinies of these systems and their components are degradation, death, and extinction, the transient behaviors of these hybrid automata remain infinitely fascinating to us for obvious reasons.

Consequently, even though hybrid automata of the kind that Pravin Varaiya explored were motivated by examples from complex engineered sys-

---

* This work was supported by grants from NSF's Qubic program, NSF's ITR program, Defense Advanced Research Projects Agency (DARPA), Howard Hughes Medical Institute (HHMI) biomedical support research grant, the US Department of Energy (DOE), the US air force (AFRL), National Institutes of Health (NIH) and New York State Office of Science, Technology & Academic Research (NYSTAR).

tems, there are many questions that he had raised in the engineering context that remain equally interesting also in the biological situation. In a paper that Pravin Varaiya wrote with Mikhail Kourjanski, they explored the question of how to characterize "stability of hybrid systems." (See [16]) In this paper they studied a particular class of hybrid automata that are now called rectangular automata, and restricted their attention to the ones in which discrete states go through a loop and also contains an infinite trajectory starting from some state. Such a viable system was shown to be exactly characterized by rectangular systems with fixed point or infinite cycle.

Because of our biological motivations, we extend the notion to *functional hybrid automata* whose flow and reset conditions are based on real functions (and even further restricted to semi-algebraic functions when we seek algorithmic solutions). We are now able to ask similar questions about stability (rather simple in this case) and limit cycles.

In particular, we show that functional hybrid automata, which can be used to model biological systems, can be reduced to systems of differential equations. As a consequence many results obtained in dynamical systems theory (e.g., Lyapunov's stability theorems and LaSalle invariance principle [17]) apply *mutatis mutandis*.

The paper is organized as follows: we start with a brief but comprehensive overview of biological system models and one interesting example—*circadian clock*, whose cyclic rhythm governs our daily function (Section 1.2), and follow it with a formal introduction to functional hybrid automata and the question of their stability (Section 1.3). We then focus on our technical approach involving a direct translation of a subclass of functional hybrid automata into systems of differential equations (Section 1.4), thus making our problem amenable to classical approaches. We place our work in the context of other related works (Section 1.5) and conclude in Section 1.6 with a discussion of how new challenges from systems biology may rely on the revolution that Pravin Varaiya and his colleagues started.

1.2 Biological System Models

The central dogma of biology translates easily to a mathematical formalism for biochemical processes involved in gene regulation. This principle states that biochemical information flow in cells is unidirectional—DNA molecules code information that gets transcribed into RNA, and RNA then gets translated into proteins. To model a regulatory system for genes, we must also include an important subclass of proteins (transcription activators), which also affects and modulates the transcription processes itself, thus completing the cycle. We can write down *kinetic mass-action* equations for the time variation of the concentrations of these species, in the form of a system of *ordinary differential equations* (ODE's) [10, 15, 24]. In particular, the transcription process can be described by equations of the *Hill* type, with its Hill coefficient n depending on the *cooperativity* among the transcription binding sites. If the concentration of DNA and RNA are denoted by M_x, M_y, etc., and those of proteins by P_x, P_y, etc., then the relevant equations are of the form:

$$\dot{M}_x = -k_1 x + k_3 \frac{1 + \theta P_y^n}{1 + P_y^n} \tag{1.1}$$

$$\dot{P}_x = -k_2 P_x + k_4 M_x \tag{1.2}$$

Each equation above is an algebraic differential equation consisting of two algebraic terms, a positive term, representing synthesis and a negative term, representing degradation. For both RNA and DNA the degradation is represented by a linear function; for RNA, synthesis through transcription is a highly nonlinear but a rational Hill-type function; and for proteins, synthesis through translation is linear function of the RNA concentration. In the equation for transcription, when $n = 1$, the equations are called *Michaelis-Menten* equations; P_y denotes the concentration of proteins involved in the transcription initiation of the DNA, k_1 and k_2 are the forward rate constants of the degradation of RNA and proteins, respectively, k_3 and k_4 are the rate constants for RNA and protein synthesis and θ models the saturation effects in transcription.

If one knew all the species involved in any one pathway, the mass-action equations for the system could be expressed in the following form

$$\dot{X}_i = f_i(X_1, X_2, \ldots, X_n), \qquad i = 1, 2, \ldots, n \tag{1.3}$$

When the number of species becomes large, the complexity of the system of differential equations grows rapidly. Furthermore, the mathematics of the dynamical system becomes increasingly complex. The integrability of the system of equations, for example, depends on the algebraic properties of appropriate bracket operations [20, 19]. We can approximately describe the behavior of such a system using a *hybrid automata* [3, 21]. The discrete states of the hybrid system describe regimes of system behavior which are qualitatively different in terms of which species and reactions predominate, and so forth. The “flows,” “invariants,” “guards,” and “reset” conditions can be approximated by algebraic systems and the decision procedures for determining various properties of these biological systems can be developed using the methods of symbolic algorithmic algebra. As we enlarge the scopes of the biological models by considering metabolic processes, signal transduction processes and subcellular biochemical processes that are specific to locations and transportation between cellular compartmentalizations, the challenges to the algorithmic complexity and approximability deepen the need for better algorithmic algebraic techniques. In the process, we are also forced to explore the connection among constructive approaches for differential algebra, commutative algebra, Tarski-algebra, etc.

As a simple illustrative example, where its limiting cyclic behavior is rather important, consider the following model of “circadian clock.” A widely-studied model of the mechanism for circadian rhythm was first proposed by Goldbeter [14] in terms of the dynamics involved in the degradation of the period protein (PER) and took into account multiple phosphorylation of PER and the negative feedback exerted by PER on the transcription of the period (*per*) gene. Informally, the *per* gene transcribes its corresponding mRNA in the nucleus at a rate negatively governed by nuclear PER protein—more nuclear PER protein implies less *per* mRNA and vice versa. The transcribed *per* mRNA leaves nucleus to get translated into PER protein, which after post-translational modifications (several successive phosphorylation steps) diffuses back into nucleus—more *per* mRNA implies more nuclear PER protein and vice versa. All these effects can be expressed succinctly in the forms of the ODE's we have described earlier. This minimal biochemical model, supported by experimental observations, resulted in a better understanding of the limit cycle of the molecular dynamics inherent to circadian oscillation. The mathematical model, created from the Michaelis-Mentens type kinetic models, is a five-dimensional system of first-order-ODE's and involved algebraic rational functions of low degree.

A more detailed model takes into account the role played by the formation of a complex between the PER and TIM proteins, and requires considering a sequence of steps for TIM similar to the ones shown below. The more complex system is 10 dimensional and omitted from discussion. Including further evidence that the TIM light response is relevant to light-induced phase shifts of the circadian clock, and its modeling through discrete mode switches, bring us back to the realm of hybrid automata. While we do not describe such a complex model here, we do emphasize the fact that understanding the limiting behavior of hybrid models such as these are important if we wish to understand how light acts as a major environmental signal for the entrainment of circadian rhythms.

In the equations below: *per* mRNA, whose cytosolic concentration is denoted by M, is synthesized in the nucleus and transferred into the cytosol, where it is degraded; the rate of synthesis of PER is proportional to M. In order to take into account the fact that PER is multiply phosphorylated, while keeping the model as simple as possible, only three states of the protein are considered: unphosphorylated (P_0), monophosphorylated (P_1) and bisphosphorylated (P_2); P_N is the nuclear PER protein.

Crucial to the mechanism of oscillations in the model is the negative feedback exerted by the nuclear form P_N in the formation of the PER-TIM complex on the synthesis of *per* (and, in the more detailed model, also *tim*) mRNAs. The negative feedback is described by a Hill-type equation. The equations below are also somewhat idealized as they ignore the linear degradation terms characterized by a relatively small, nonspecific rate constant. This rate constant does not play an important role in the system's oscillatory behavior but ensures that a steady state exists even when degradations are inhibited.

$$\dot{M} = \nu_5 \frac{K_1^n}{(K_1^n + P_N^n)} - \nu_m \frac{M}{(K_m + M)} \tag{1.4}$$

$$\dot{P_0} = \kappa_5 M - V_1 \frac{P_0}{(K_1 + P_0)} + V_2 \frac{P_1}{(K_2 + P_1)} \tag{1.5}$$

$$\dot{P_1} = V_1 \frac{P_0}{(K_1 + P_0)} - V_2 \frac{P_1}{(K_2 + P_1)} - V_3 \frac{P_1}{(K_3 + P_1)} + V_4 \frac{P_2}{(K_4 + P_2)} \tag{1.6}$$

$$\dot{P_2} = V_3 \frac{P_1}{(K_3 + P_1)} - V_4 \frac{P_2}{(K_4 + P_2)} - \kappa_1 P_2 + \kappa_2 P_N - \nu_d \frac{P_2}{(K_d + P_2)} \tag{1.7}$$

$$\dot{P_N} = \kappa_1 P_2 - \kappa_2 P_N \tag{1.8}$$

$$P_t = P_0 + P_1 + P_2 + P_N \tag{1.9}$$

The mathematical model indicates that during oscillation, the peak in *per* mRNA precedes by several hours the total PER protein. The key insight was that multiple PER phosphorylation introduces time-delays which strengthen the negative feedback to produce oscillation. An algebraic analysis shows that the rhythm only occurs in a range bounded by two critical values of the "maximum rate of PER degradation." The same analysis can be used to show a "rough homeomorphism" between this high-dimensional system and a simpler two-dimensional van der Pol equation. The other critical parameter was found to be the "average rate of PER transport into the nucleus." The critical dependence of the limit cycle on the degradation parameter was a key for biologists to understand the altered period of *per* mutants.

In future, we may wish to study further extensions of this initial model: the PER-TIM model of Goldbeter, that incorporates the other protein TIM, whose dimerization with PER plays an important role in providing stability to the limit cycle; a better model of Tyson et al., that takes into account the detailed structure of PER-phosphorylation and inherent competition among several key processes and light-sensitivity of TIM. Many of these detailed models will require description in terms of hybrid modes. While these extended models are more complex, they appear to remain homeomorphic to simple van-der-Pol-like system, while adding to the stability to the over-all system.

Another interesting avenue to explore concerns the feasibility of synthetic cellular clocks. Is it feasible to design simple oscillating systems of a desired periodicity by genetic engineering in appropriate cell hosts? If so, such a system could be used as a stringent test system of our ability to model complex cellular pathways. We may conceive of a simple transcriptional feedback system, using temperature sensitive competitive inhibitors (so that clocks can be reset by temperature shifts) and fluorescent reporter systems (so that the phase of the cycle can be examined in individual cells and in the population). The advantages of such a system reside in its ease of manipulation, ease of monitoring, coupled to the use of genetic selection to explore unanticipated behaviors.

1.3 Hybrid Automata: Stability and Limit Cycles

1.3.1 Functional Hybrid Automata—Syntax

The notion of *Hybrid Automata* was first introduced in [4] as a model and specification language for systems consisting of a discrete program within a continuously changing environment. For our purpose, it is convenient to introduce a specialized notion of *functional hybrid automata*, whose flow and reset conditions are further restricted to functions over the reals.

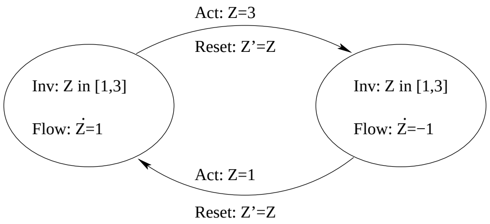
Following notations and conventions will be used through out the paper: Capital letters $Z_1, \ldots, Z_k, Z'_1, \ldots, Z'_k$ will denote variables which range over $\mathbb{R}$. Moreover, Z will denote the vector of variables $\langle Z_1, \ldots, Z_k \rangle$; similarly, Z' will denote the vector $\langle Z'_1, \ldots, Z'_k \rangle$ and Z^n, the vector $\langle Z^n_1, \ldots, Z^n_k \rangle$. The variable T will be used for time, ranging over $\mathbb{R}^+$. The small letters p, q, r, s, ... will denote k-dimensional vectors of real numbers.

Given a formula (function) φ we will use the notation $\varphi(Z_1, \ldots, Z_n)$ to stress the fact that the set of variables occurring in φ is included in $\{Z_1, \ldots, Z_n\}$. By extension, $\varphi(Z^1, \ldots, Z^n)$ will indicate that the variables of φ are included in the set of components of the vectors $Z^1, \ldots, Z^n$. Given a formula (function) $\varphi(Z^1, \ldots, Z^{i-1}, Z^i, Z^{i+1}, \ldots, Z^n)$, the formula (function) obtained by componentwise substitution of the elements of Z^i with the elements of p will be denoted by $\varphi(Z^1, \ldots, Z^{i-1}, p, Z^{i+1}, \ldots, Z^n)$. If the only variables in φ are the elements of Z^i, then after the substitution, the value of $\varphi(p)$ will be assumed to be available.

Definition 1 (Hybrid Automata). *A* hybrid automaton $H = (Z, \dot{Z}, Z', \mathcal{V}, \mathcal{E}, \mathit{Inv}, \mathit{Flow}, \mathit{Act}, \mathit{Reset})$ *of dimension k has following components:*

- *$Z = \langle Z_1, \ldots, Z_k \rangle$, $\dot{Z} = \langle \dot{Z}_1, \ldots, \dot{Z}_k \rangle$, and $Z' = \langle Z'_1, \ldots, Z'_k \rangle$ are vectors of variables ranging over $\mathbb{R}$; Z denotes the values of the continuous variables; $\dot{Z}$ denotes the first-order derivatives taken with respect to the time $T \in \mathbb{R}^+$ during continuous change; Z' denotes the values after a discrete jump;*
- *$\langle \mathcal{V}, \mathcal{E} \rangle$ is a finite directed graph; the nodes, $\mathcal{V}$, are called* control modes*, the edges, $\mathcal{E}$, are called* control switches*;*
- *Each vertex $v \in \mathcal{V}$ is labeled by the formulae $\mathit{Inv}(v)(Z)$ and $\mathit{Flow}(v)(Z, \dot{Z})$; $\mathit{Inv} = \{\mathit{Inv}(v)(Z) \mid v \in \mathcal{V}\}$ and $\mathit{Flow} = \{\mathit{Flow}(v)(Z, \dot{Z}) \mid v \in \mathcal{V}\}$;*
- *Each edge $e \in \mathcal{E}$ is labeled by the formulae $\mathit{Act}(e)(Z)$ and $\mathit{Reset}(e)(Z, Z')$; $\mathit{Act} = \{\mathit{Act}(e)(Z) \mid e \in \mathcal{E}\}$ and $\mathit{Reset} = \{\mathit{Reset}(e)(Z, Z') \mid e \in \mathcal{E}\}$.*

Example 1. Consider the following simple hybrid automaton "oscillating" between two values.



Starting in the control mode to the left Z grows at constant rate of 1. After 3 time units, upon reaching the value of $Z = 3$, it immediately jumps to the alternate control mode to the right, where Z now decreases until it reaches a value of $Z = 1$. Under this condition, it jumps back to the mode to the left. The automaton moves back and forth forever between these two modes.

Definition 2 (Functional Hybrid Automata and its Syntax). *A* functional hybrid automaton $H = (Z,\ \dot{Z},\ Z',\ \mathcal{V},\ \mathcal{E},\ Inv,\ Flow,\ Act,\ Reset)$ *of dimension k is a hybrid automaton of same dimension satisfying following additional properties:*

- *Each invariant formula $Inv(v)$ characterizes a closed subset of $\mathbb{R}^k$;*
- *Each flow formula $Flow(v)$ is of the form $\dot{Z} = \psi(v)(Z)$ and the Cauchy problem $\dot{Z} = \psi(v)(Z)$ with initial condition $Z(0) = r$ has a unique solution for each r satisfying $Inv(v)$;*
- *For each r on the frontier set of the invariant $\delta(Inv(v))$ the solution $Z = \varphi(T)$ of the Cauchy problem $\dot{Z} = \psi(v)(Z)$ with initial condition $Z(0) = r$ further satisfies the following property:*

$$\forall \epsilon > 0, \varphi(\epsilon) \notin Inv(v);$$

- *Each activity formula $Act(\langle v,\ u\rangle)$ characterizes a subset of the frontier set $\delta(Inv(v))$;*
- *Each reset formula $Reset(e)$ is of the form $Z' = \rho(e)(Z)$, where $\rho(e)$ is an injective function.*

Example 2. The hybrid automaton of Example 1 is a functional hybrid automaton. For another example, see the hybrid automata proposed in [13] to model the Delta-Notch signaling process; these can be rewritten as functional hybrid automata by using closed invariant conditions. This change has no effect on the behaviors of the automata.

Henceforth, we restrict our discussions only to functional hybrid automata.

1.3.2 Hybrid Automata—Semantics

The semantics of functional hybrid automata can be defined in terms of execution traces. Traces are sequences of pairs with each pair consisting of a point and a control mode. Maximal traces are traces which cannot be extended.

Definition 3 (Functional Hybrid Automata and its Semantics).
Let $H = (Z, \dot{Z}, Z', \mathcal{V}, \mathcal{E}, Inv, Flow, Act, Reset)$ be an hybrid automaton of dimension k.

A location ℓ *of H is a pair $\langle v, r\rangle$, where $v \in \mathcal{V}$ is a state and $r = \langle r_1, \ldots, r_k\rangle \in \mathbb{R}^k$ is an assignment of values for the variables of Z. An* admissible location $\langle v, r\rangle$ *is one for which $Inv(v)(r)$ holds.*

The continuous reachability transition relation $\rightarrow_C$ *between admissible locations is defined as follows:*

$$\begin{aligned}&\langle v, r\rangle \rightarrow_C \langle v, s\rangle\\ &\quad \textit{iff}\ \ \exists t > 0, \Big(f(0) = r \wedge f(t) = s \wedge \forall t' \in [0, t]\big(Inv(v)(f(t'))\big)\Big)\end{aligned}$$

where f is the solution of the Cauchy problem $\dot{Z} = \psi(v)(Z)$ with initial condition $Z(0) = r$.

The discrete reachability transition relation $\rightarrow_D$ *between admissible locations is defined as follows:*

$$\begin{aligned}&\langle v, r\rangle \rightarrow_D \langle u, s\rangle\\ &\quad \textit{iff}\ \ \langle v, u\rangle \in \mathcal{E} \wedge Act(\langle v, u\rangle)(r) \wedge s = \rho(\langle v, u\rangle)(r).\end{aligned}$$

A trace *of H is a sequence $\ell_0, \ell_1, \ldots, \ell_n, \ldots$ of admissible locations such that for each $i \geq 0$ either $\ell_i \rightarrow_C \ell_{i+1}$ or $\ell_i \rightarrow_D \ell_{i+1}$. A trace of H is* maximal *if it is not a proper prefix of another trace of H.*

Notice that our definition of trace is rather general: (1) the length of a trace can be either finite or infinite; (2) maximal traces can be of finite length.

1.3.3 Cyclic Traces

As discussed in Section 1.2, well-controlled robust periodic behavior is crucial to many biological systems: cell cycles, circadian clocks, cyclic expression patterns of segmentation clocks (e.g., the Delta/Notch signal transduction system), etc. When we model them with hybrid automata (see, e.g., [13, 2]) periodic behaviors correspond to cyclic traces. Hence, for a given hybrid automaton H, one may wish to determine:

> *Can this hybrid automaton H exhibit a cyclic trace? More formally, does there exist a trace of H taking the form $\ell_0, \ell_1, \ldots, \ell_n, \ell_0$ with $n \geq 0$?*

There are only a handful of results that directly and explicitly address this question in the context of hybrid automata – efforts directed at the question of stability of cyclic traces are even rarer. In fact, since hybrid automata are *highly non-deterministic*, the problem of analyzing cyclic trace in the full generality is difficult. This limitation does not always apply, when it comes to biological systems. Hence, by modeling biochemical processes with functional hybrid automata, we try to limit the non-determinism, and exploit this property to study cyclic traces by suitably modifying results developed in the area of dynamic systems

Let us begin by classifying cyclic traces in order to understand what makes them difficult to detect. If $\langle v, r\rangle$ is an admissible location of H, such that $\psi(v)(r) = 0$, then the trace $\langle v, r\rangle, \langle v, r\rangle$ is a cyclic trace of H. We call such a cyclic trace a *first gender cycle*.

Proposition 1. *Let H be a functional hybrid automaton. If for each vertex v the function $\psi(v)$ and the formula $Inv(v)$ are polynomials over the reals, then the existence of first gender cycles in H is decidable.*

Proof. For each vertex v consider the following first order formula

$$Inv(v)(Z) \wedge \psi(v)(Z) = 0.$$

The solutions of this formula are the points r such that $\langle v, r\rangle, \langle v, r\rangle$ is a first gender cycle. Since the satisfiability of the formula for any vertex v is decidable [22] and since the number of nodes v is finite, the first gender cycle problem is decidable, as claimed. □

We remark parenthetically that the result, shown above, can also be extended to *o-minimal* theories [12].

Assume further that $\psi(v)$ is such that a point r satisfying $Inv(v)$ exists and the solution of the Cauchy problem with initial condition $Z(0) = r$ is a periodic function with its image included in $Inv(v)$. Then the trace $\langle v, r\rangle, \langle v, r\rangle$ is a cyclic trace of H. We call a cyclic trace of this form a *second gender cycle*.

In order to detect second gender cycles it is necessary to study all the differential systems $\psi(v)$'s and check if they admit periodic solutions. Many results have been developed in the areas of dynamical systems and numerical analysis to detect periodic solutions and study their stability properties. Most of these results are built upon Lyapunov's stability theorems and LaSalle invariance principle [17]. Principles which apply to monotone systems have been recently studied in [6, 7].

In general, a cyclic trace can be $\langle v_0, r_0\rangle, \langle v_1, r_1\rangle, \ldots, \langle v_n, r_n\rangle, \langle v_0, r_0\rangle$ and may contain repeated copies of several discrete nodes internally, i.e., there may exist $i \neq j \leq n$ with $v_i = v_j$. We will call a cyclic trace of this form a *third gender cycle*, a detailed study of which is the key topic of this paper. In particular, we aim to reduce this problem to a more classical problem: namely, that of studying periodic solutions of systems of differential equations, as in the case of second gender cycles.

In a trace there could be many consecutive continuous transitions as well as many consecutive discrete transitions. However, when we are looking for cyclic traces we can restrict our attention to traces in which each continuous transition is followed by a discrete transition.

Definition 4. *Let H be a functional hybrid automaton. A trace ℓ_0, ℓ_1, …, ℓ_n, … is said to be in* normal form *if it holds that $\ell_i \to_C \ell_{i+1}$ implies $\ell_{i+1} \not\to_C ell_{i+2}$, for $i = 0, 1, \ldots$.*

Lemma 1. *Let H be a functional hybrid automaton. If H admits a cyclic trace, then it admits a cyclic trace in normal form.*

Proof. Let $T = \ell_0, \ldots, \ell_n, \ell_0$ be a cyclic trace of H. If $n = 0$, then the trace is already in normal form. Otherwise, $n > 0$, and each place the trace contains a subsequence of the form $\ell_i \to_C \ell_{i+1} \to_C \ell_{i+2}$ in T, we may replace it with $\ell_i \to_C \ell_{i+2}$. By repeated replacement of this kind, until it is no longer possible, we obtain a sequence which is a cyclic trace of H and is in normal form. □

1.4 From Deterministic Hybrid Automata to ODE's

In our definition of functional hybrid automata we limit the non-determinism to the following cases:

1. There exists a point which satisfies more than one invariant condition;
2. There exists a point which satisfies more than one activation condition.

Inside a vertex, the behavior of a functional hybrid automaton, by the second condition of Definition 2, is deterministic, as it imposes existence and uniqueness of the solution for each initial condition. Note, further, that when a functional hybrid automaton reaches the frontier of an invariant, it must jump immediately, since we imposed that the solutions immediately cross the frontier. Once the automaton decides (perhaps nondeterministically) which edge it may take, it uses a reset condition in a deterministic manner, as its reset condition is a function. Thus it remains to show that this second source of non-determinism can be removed, and we can translate a functional hybrid automaton into a system of differential equations.

Definition 5 (Deterministic Functional Hybrid Automata). *Let $H = (Z, \dot{Z}, Z', \mathcal{V}, \mathcal{E}, Inv, Flow, Act, Reset)$ be a functional hybrid automaton. We say that H is* deterministic*, if for each vertex $v \in \mathcal{V}$ and for each pair of edges e_1, $e_2 \in \mathcal{E}$ with a common source vertex v we have*

$$Act(e_1) \cap Act(e_2) = \emptyset.$$

In our definition of deterministic functional hybrid automata there is still an apparent source of non-determinism and it is due to the fact that given a point $r \in \mathbb{R}^k$ it is possible to start from more than one location of the form $\langle v, r\rangle$.

Lemma 2. *Let H be a deterministic functional hybrid automaton and $\langle v, r\rangle$ be an admissible location of H. Then there exists one maximal trace in normal form ℓ_0, ℓ_1, …, ℓ_n, … with $\ell_0 = \langle v, r\rangle$.*

Proof. The sequence $\langle v, r\rangle$ is always a trace of H. Hence, it can be extended to at least one maximal trace Tr. As in the proof of Lemma 1, we can map Tr into a maximal trace in normal form which starts from $\langle v, r\rangle$.

We may derive a contradiction as follows, by assuming that there are two maximal traces in normal form, both starting from $\langle v, r\rangle$. We use $\ell_0, \ell_1, \ldots, \ell_n, \ldots$ and $\ell'_0, \ell'_1, \ldots, \ell'_n, \ldots$ to denote the two traces. Let i be the smallest index such that $\ell_i \neq \ell'_i$. It must be that $i > 0$. Following four cases must be considered:

1. $\ell_{i-1} \to_C \ell_i$ and $\ell_{i-1} = \ell'_{i-1} \to_C \ell'_i$;
2. $\ell_{i-1} \to_D \ell_i$ and $\ell_{i-1} = \ell'_{i-1} \to_D \ell'_i$;
3. $\ell_{i-1} \to_C \ell_i$ and $\ell_{i-1} = \ell'_{i-1} \to_D \ell'_i$;
4. $\ell_{i-1} \to_D \ell_i$ and $\ell_{i-1} = \ell'_{i-1} \to_C \ell'_i$.

Since the last two cases are essentially equivalent, we need consider only the first three cases. The first case can be ruled out since in each control mode the solutions of the differential equations are unique. The second case cannot occur since the activation conditions of H are disjoint and the reset are functional. Finally, the third case cannot occur because from $\ell_{i-1} = \ell'_{i-1} \to_D \ell'_i$ we conclude that $\ell_{i-1} = \langle u, s\rangle$ and s is on the frontier of $\mathit{Inv}(u)$, thus, implying that the solution of $\dot{Z} = \psi(u)(Z)$ goes outside $\mathit{Inv}(u)$. This leads to the desired contradiction: it cannot be that $\ell_{i-1} \to_C \ell_i$. $\square$

Given an admissible location ℓ we use the notation $Tr(\ell)$ to denote the maximal trace in normal form starting from ℓ.

Henceforth, we focus our attention on a deterministic functional hybrid automaton H. We aim to encode H into a system of differential equations whose solutions correspond to the traces of H. We start by encoding the nodes of $\mathcal{V}$. Let $|\mathcal{V}| = n$, and consider an ordering $[v_1, \ldots, v_n]$ of $\mathcal{V}$. We map each vertex of $\mathcal{V}$ to a point in $\mathbb{R}^n$ as follows:

$$\begin{aligned} \mu : \mathcal{V} &\to \mathbb{R}^n \\ v_i &\mapsto \langle 0, 0, \ldots, 1, \ldots, 0\rangle, \end{aligned}$$

where 1 is in position i.

Let $R_1, \ldots, R_n, S_1, \ldots, S_n$ be $2n = |\mathcal{V}|$ fresh variables. Let also $W_1, \ldots, W_k$ be k fresh variables, where k is the dimension of H.

For each vertex $v \in \mathcal{V}$ we consider the system of differential equations $\Psi(v)$ on $\mathbb{R}^{2k+2n}$ defined as:

$$\begin{cases} \dot{Z} &= \psi(v)(Z) \\ \dot{W} &= \psi(v)(W) \\ \dot{R} &= 0 \\ \dot{S} &= 0. \end{cases}$$

This system describes the continuous evolution in v. The variables Z's and W's evolve as described in the mode v. The variables R's and S's do not evolve. They are used simply to encode the fact that the automaton is in vertex v.

Now we can glue together the systems of the different modes, i.e., we will encode the discrete jumps into differential systems. The basic ideas behind the encoding are as follows. Let us assume that we are in a point of the form $\langle z, z, \mu(v_i), \mu(v_i)\rangle$ and z satisfies $Act(\langle v_i, v_j\rangle)$. We use two time instants to jump from $\langle z, z, \mu(v_i), \mu(v_i)\rangle$ to $\langle \rho(\langle v_i, v_j\rangle)(z), \rho(\langle v_i, v_j\rangle)(z), \mu(v_j), \mu(v_j)\rangle$. During the first instant: Z moves on the segment between z and $\rho(\langle v_i, v_j\rangle)(z)$ at constant speed $\rho(\langle v_i, v_j\rangle)(z) - z$; W remains fixed since it is used to determine the constant speed at which Z moves; R moves on the segment between $\mu(v_i)$ and $\mu(v_j)$ at constant speed 1; S does not move so that it is clear that we are moving from $\mu(v_i)$ to $\mu(v_j)$ and not vice versa. During the second instant we need to update W and S. Hence in this case, Z does not move; W moves on the segment between z and $\rho(\langle v_i, v_j\rangle)(z)$ at constant speed; R does not move; S moves on the segment between $\mu(v_i)$ and $\mu(v_j)$ at constant speed. In particular, to determine the segment on which W has to move we need to use the values of S and R after one instant (these encode the edge) and the value of Z after one instant (to determine the constant speed).

We start with the system for the first instant. For each edge $\langle v_i, v_j\rangle$ we consider the system $\Psi_1(\langle v_i, v_j\rangle)$ defined as:

$$\begin{cases} \dot{Z} &= \rho(\langle v_i, v_j\rangle)(W) - W \\ \dot{W} &= 0 \\ \dot{R} &= \mu(v_j) - \mu(v_i) \\ \dot{S} &= 0 \end{cases}$$

As far as the second instant is concerned, we proceed as follows: For each edge $\langle v_i, v_j\rangle$ we consider the system $\Psi_2(\langle v_i, v_j\rangle)$ defined as:

$$\begin{cases} \dot{Z} &= 0 \\ \dot{W} &= Z - \rho^{-1}(\langle v_i, v_j\rangle)(Z) \\ \dot{R} &= 0 \\ \dot{S} &= \mu(v_j) - \mu(v_i) \end{cases}$$

To conclude our construction, we collect and assemble the systems $\Psi(v)$, $\Psi_1(e)$, and $\Psi_2(e)$ combining the invariant and activation conditions of H. For a given formula $\gamma(Z)$ whose solutions denote a subset $G \subseteq \mathbb{R}^k$, we use $Op(\gamma)(Z)$ to denote the formula associated with the interior of G. Moreover, consider $\emptyset$, the system of differential equations which equates all the derivatives to 0. Let the system $\mathcal{H}$ be defined as follows:

$$\begin{cases} \Psi(v_i), & \text{if } Op(Init(v_i))(Z) \wedge R = S = \mu(v_i); \\ \Psi_1(\langle v_i, v_j\rangle), & \text{if } Act(\langle v_i, v_j\rangle)(W) \wedge S = \mu(v_i) \wedge R_j < 1; \\ \Psi_2(\langle v_i, v_j\rangle), & \text{if } R = \mu(v_j) \wedge S_i > 0; \\ \emptyset, & \text{otherwise.} \end{cases}$$

Notice that this construction uses $2n$ variables to encode the discrete part of the automaton. This construction avoids intersections of the solutions during the jumps. We could obtain the same result using only 6 variables, since given n points in $\mathbb{R}^3$ we can always connect them with n^2 non-intersecting curves.

We prove that the solutions of the system $\mathcal{H}$ and the traces of the deterministic functional hybrid automaton H *correspond* to each other, i.e., they are in a sense equivalent. We limit our arguments to traces of infinite length, since for cyclic solutions this suffices. The definitions and results can be modified appropriately to deal with traces of finite length.

Definition 6. *Let H be a functional hybrid automaton of dimension k with n control modes. Let $f : \mathbb{R}^+ \to \mathbb{R}^{n+k}$ be a function and $Tr = \ell_0, \ell_1, \ldots, \ell_m, \ldots$ be a trace of H of infinite length. We say that f and Tr* agree *if there exists an increasing sequence $t_0, t_1, \ldots, t_m, \ldots$ of positive reals such that for each i it holds $\ell_i = f(t_i)$.*

Theorem 1. *Let H be a deterministic functional hybrid automaton of dimension k and $\langle v, r\rangle$ be an admissible location of H such that $Tr(\langle v, r\rangle)$ has infinite length. The solution $Z = f^1_{\langle v,r\rangle}(t)$, $R = f^3_{\langle v,r\rangle}(t)$ of $\mathcal{H}$ with initial conditions $Z = W = r$ and $R = S = \mu(v)$ and the trace $Tr(\langle v, r\rangle)$ agree.*

Proof. We use $Z = f^1_{\langle v,r\rangle}(t)$, $W = f^2_{\langle v,r\rangle}(t)$, $R = f^3_{\langle v,r\rangle}(t)$, and $S = f^4_{\langle v,r\rangle}(t)$ to denote the solution of $\mathcal{H}$ with initial conditions $Z = W = r$ and $R = S = \mu(v)$. We have to define the sequence $t_0, t_1, \ldots, t_m, \ldots$ satisfying Definition 6. Let $Tr(\langle v, r\rangle)$ be of the form $\langle v, r\rangle, \langle w_1, s_1\rangle, \ldots, \langle w_m, s_m\rangle, \ldots$. We define $t_0 = 0$. The initial value clearly satisfies Definition 6. Let us assume inductively that we have defined $t_0, \ldots, t_i$ satisfying Definition 6; we define t_{i+1} as follows:

- if $\langle w_i, s_i\rangle \to_D \langle w_{i+1}, s_{i+1}\rangle$, then $t_{i+1} = t_i + 2$;
- if $\langle w_i, s_i\rangle \to_C \langle w_{i+1}, s_{i+1}\rangle$, then $t_{i+1} = min\{t > t_i \mid f^1_{\langle w_i,s_i\rangle}(t) = s_{i+1}\}$.

In the first case we see that we still satisfy Definition 6, since $\langle w_i, s_i\rangle$ is in the activation region of $\langle w_i, w_{i+1}\rangle$ and after two time units the system $\mathcal{H}$ reaches the point reachable with the discrete jump. As far as the second case is concerned, we get the same conclusion as a consequence of the facts that we are considering autonomous systems and that the trace $Tr(\langle v, r\rangle)$ is in normal form (hence the next transition is discrete). $\square$

Thus, we conclude that cyclic traces of H *agrees* with periodic orbits of $\mathcal{H}$.

Corollary 1. *Let H be a deterministic functional hybrid automaton. H admits a cyclic trace if and only if $\mathcal{H}$ has a periodic orbit.*

Notice that if the second condition of Definition 2 fails, for example, because the flows can have either no solution or more than one solution, then Lemma 2 is false. Nonetheless, we can still construct $\mathcal{H}$ and prove correspondence between traces of H and solutions of $\mathcal{H}$.

1.5 Related Literature

To place the results described here in the context of a growing literature, we mention few related results.

The closest in spirit to our results are those in [18]. There, hybrid automata are studied from a dynamical systems perspective. The paper rigorously proves necessary and sufficient conditions for existence, uniqueness, and continuity of traces. Under these assumptions, Lyapunov's theorem on stability via linearization and LaSalle's invariance principle are generalized to hybrid automata. While our notion of deterministic functional hybrid automata is intuitively similar to the notion of deterministic hybrid automata introduced in [18], there are many fundamental differences: we do not impose that the flows are globally Lipschitz continuous, but we assume that they have a unique solution for each initial condition; we impose on the resets an injectiveness condition. When the flows of a deterministic functional hybrid automaton H are globally Lipschitz continuous all the results proved in [18] apply to H. In the general case we can map H into the dynamic system $\mathcal{H}$ and try to directly apply stability and invariance results to $\mathcal{H}$.

In [11] hybrid systems are defined as sets of systems of differential equations. Which system has to be used is decided by the initial conditions and by a discrete control. On these hybrid systems, stability conditions are studied explicitly. The systems in [11] are not continuously linked in the following sense: when there is a switch in the discrete part, there is a jump in the continuous part, hence stability results for dynamic systems cannot be directly applied. The main difference with our construction is that we connect the flows continuously so that we get a piecewise defined dynamic system.

In [1] an affine hybrid automaton H is mapped into a new automaton $Bl(H)$ which has the same periodic orbits and equilibrium points, but no Zeno behaviors. The basic idea behind the mapping is to split each control switch adding a new control mode and to introduce a time delay in the new modes. This is similar to what we do in our construction when we use 2 time instants for each edge crossing. In fact, we can prove that the Zeno behaviors of H corresponds to solutions of $\mathcal{H}$ in which the time flow is unbounded.

In [9] domains of convergence are studied by mapping systems of differential equations into discrete automata with an infinite number of states. By combining the construction we describe in this paper with that defined in [9] we get a discretization method for hybrid automata. Relationships with other discretization methods (e.g., [5, 23]) remain to be analyzed.

1.6 Conclusion

Finally, we return to the biological questions that initiated this journey into the stability of hybrid automata. At present, we lack the ability to analyze all but the simplest regulatory structures composed of handful of genes and we have no means of even intelligently conjecturing what universal principles unify biology. Our notions of biological robustness and arguments in its favor are often anecdotal, speculative and unsupported by data. For instance, there have been raging debates about the nature of the robustness exhibited by a circadian clock model that is composed of analogs of both PER and TIM, but also taking into account the reality that the copy number of PER-TIM complexes can only assume a small and random number. For instance, in the work of Naama Barkai and Stan Leibler [8], they speculate existence of an unmodeled hysteresis mechanism in circadian clock models to confer on it some degree of robustness. And yet, there are others, who using similar simulations, have argued that the original model is already robust as it is. Clearly, if the truth must be found, it will need formal methods that no amount of simulation can deliver. Pravin Varaiya's insights and instincts, buried among his results on engineering hybrid systems, may provide the methods we seek to solve such problems in systems biology.

References

1. A. Aaron and S. Sastry. Blowing up Affine Hybrid Systems. In *IEEE Conference on Decision and Control (CDC'04)*, pages 473–478. IEEE Press, 2004.
2. R. Alur, C. Belta, F. Ivancic, V. Kumar, M. Mintz, G. J. Pappas, H. Rubin, and J. Schug. Hybrid Modeling and Simulation of Biomolecular Networks. In M. D. Di Benedetto and A. L. Sangiovanni-Vincentelli, editors, *Hybrid Systems: Computation and Control (HSCC'01)*, volume 2034 of *LNCS*, pages 19–32. Springer-Verlag, 2001.
3. R. Alur, C. Courcoubetis, N. Halbwachs, T. A. Henzinger, P. H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The Algorithmic Analysis of Hybrid Systems. *Theoretical Computer Science*, 138(1):3–34, 1995.
4. R. Alur, C. Courcoubetis, T. A. Henzinger, and P. H. Ho. Hybrid Automata: An Algorithmic Approach to the Specification and Verification of Hybrid Systems. In R. L. Grossman, A. Nerode, A. P. Ravn, and H. Richel, editors, *Hybrid Systems (HS'92)*, volume 736 of *LNCS*, pages 209–229. Springer-Verlag, 1993.
5. R. Alur, T. Dang, and F. Ivancic. Progress on Reachability Analysis of Hybrid Systems Using Predicate Abstraction. In O. Maler and A. Pnueli, editors, *Hybrid Systems: Computation and Control (HSCC'03)*, volume 2623 of *LNCS*, pages 4–19. Springer-Verlag, 2003.
6. D. Angeli and E. D. Sontag. Monotone Control Systems. *IEEE Transaction on Automatic Control (TAC)*, 48:1684–1698, 2003.
7. D. Angeli and E. D. Sontag. An Analysis of a Circardian Model Using the Small-Gain Approach to Monotone Systems (I). In *IEEE Conference on Decision and Control (CDC'04)*, pages 575–578. IEEE Press, 2004.
8. N. Barkai and S. Leibler. Circadian Clocks Limited by Noise. *Nature*, 403:267–268, 2000.
9. R. W. Brockett. Pulses, Periods, and Cohomological Terms in Functional Expansions. In *System Theory.* Springer-Verlag, 1993.
10. A. Cornish-Bowden. *Fundamentals of Enzyme Kinetics (3rd edition).* Portland Press, 2004.

11. R. DeCarlo, M. Branicky, S. Pettersson, and B. Lennartson. Perspectives and Results on the Stability and Stabilizability of Hybrid Systems. *Proceedings of IEEE, Special Issue on Hybrid Systems*, 88(7):1069–1082, 2000.
12. L. Van Den Dries. *Tame Topology and O-minimal Structures*, volume 248 of *London Mathematical Society Lecture Note*. Cambridge University Press, 1998.
13. R. Ghosh, A. Tiwari, and C. Tomlin. Automated Symbolic Reachability Analysis; with Application to Delta-Notch Signaling Automata. In O. Maler and A. Pnueli, editors, *Hybrid Systems: Computation and Control (HSCC'03)*, volume 2623 of *LNCS*, pages 233–248. Springer-Verlag, 2003.
14. A. Goldbeter. A Model for Circadian Oscillations in the Drosophila Period Protein (PER). *Proc R Soc Lond B Biol Sci.*, 261:319–324, 1995.
15. J. P. Keener and J. Sneyd. *Mathematical Physiology.* Springer-Verlag, 1998.
16. M. Kourjanski and P. Varaiya. Stability of Hybrid Systems. In P.J. Antsaklis, W. Kohn, and S. Sastry, editors, *Hybrid Systems (HS'96)*, volume 1066 of *LNCS*, pages 413–423. Springer-Verlag, 1996.
17. J. P. LaSalle. Stability Theory for Ordinary Differential Equations. *J. Diff. Equat.*, 4:57–65, 1968.
18. J. Lygeros, K. H. Johansson, S. N. Simić, J. Zhang, and S. Sastry. Dynamical Properties of Hybrid Automata. *IEEE Transactions on Automatic Control (TAC)*, 48(1):2–14, 2003.
19. B. Mishra. *Computational Differential Algebra*, pages 111–145. World-Scientific, Singapore, 2000.
20. B. Mishra. A Symbolic Approach to Modeling Cellular Behavior. In S. Sahni, V. K. Prasanna, and U. Shukla, editors, *High Performance Computing (HiPC'02)*, volume 2552 of *LNCS*, pages 725–732. Springer-Verlag, 2002.
21. A. Nerode and W. Kohn. Hybrid Systems and Constraint Logic Programming. In D. S. Warren, editor, *International Conference on Logic Programming (ICLP'93)*, pages 18–24. MIT Press, 1993.
22. A. Tarski. *A Decision Method for Elementary Algebra and Geometry.* Univ. California Press, 1951.
23. A. Tiwari and G. Khanna. Series of Abstraction for Hybrid Automata. In C. J. Tomlin and M. Greenstreet, editors, *Hybrid Systems: Computation and Control (HSCC'02)*, volume 2289 of *LNCS*, pages 465–478. Springer-Verlag, 2002.
24. E. O. Voit. *Computational Analysis of Biochemical Systems. A Pratical Guide for Biochemists and Molecular Biologists.* Cambridge University Press, 2000.