FORMAL ANALYSIS OF SCIENTIFIC-COMPUTATION METHODS

Gadiel Auerbach [*] **Orna Kupferman** [**]

[*] *IBM Haifa Research Laboratory, Haifa 31905, Israel*
Email: gadiel@il.ibm.com
[**] *School of Engineering and Computer Science,*
Hebrew University, Jerusalem 91940, Israel
Email: orna@cs.huji.ac.il

Abstract: The work examines the possibility of using formal-verification methods and tools for reasoning about scientific-computation methods. The need to verify about infinite-state systems has led to the development of formal frameworks for modeling infinite on-going behaviors, and it seems very likely that these frameworks can also be helpful in the context of numerical methods. In particular, the use of hybrid systems, which model infinite-state systems with a finite control, seems promising.
The work introduces *Probabilistic o-minimal hybrid systems*, which combine flows definable in an o-minimal structure with the probabilistic choices allowed in probabilistic hybrid systems. We show that probabilistic o-minimal hybrid systems have finite bisimulations, thus the reachability and the nonemptiness problems for them are decidable. To the best of our knowledge, this forms the strongest type of hybrid systems for which the nonemptiness problem is decidable, hence also the strongest candidate for modelling scientific-computation methods.

Keywords: Hybrid systems, probabilistic models, formal methods

1. INTRODUCTION

This work examines the possibility of using formal-verification methods and tools for reasoning about scientific-computation methods. The need to reason about infinite-state systems has led to the development of formal frameworks for modeling infinite on-going behaviors, and it seems very likely that these frameworks can also be helpful in the context of numerical methods. In particular, the use of hybrid systems, which model infinite-state systems with a finite control, seems promising.

We introduce some negative results about the appropriateness of existing formal-verification methods and tools. In particular, we show that an extension of the allowed dynamic of hybrid systems with linear differential equations in ways that would enable the modeling of simple numerical schemes result in systems for which the reachability problem (which is at the heart of the nonemptiness problem and other problems to which verification is reduced) is undecidable. These results have led to a sequence of undecidability results for problems in numerical methods (e.g., the problem of deciding whether the Newton method converges for a given function and an initial value is proved as undecidable), which together with similar known results from (Blum *et*

[1] The research is supported by BIKURA grant 032.8967 of the Israeli Science Foundation.

al., 1997) suggested that our attempts should be less ambitious, as many numerical schemes involve undecidable queries.

Accordingly, our efforts have focused in different types of extensions of hybrid systems that seem helpful in the context of numerical analysis and for which the reachability problem is still decidable. A unified approach to decidability questions for verification algorithms of hybrid systems is obtained by the construction of a bisimulation. Bisimulations are finite state quotients whose reachability properties are equivalent to those of the original infinite state hybrid system. *Order-minimal hybrid systems*, which were introduced by (Lafferriere *et al.*, 2000), are initialized hybrid systems whose relevant sets and flows are definable in an order-minimal (o-minimal, for short) structure (that is, every definable subset of $\mathbb{R}$ is a finite, possible unbounded, union of points and intervals). This extends classical hybrid systems whose dynamics consist of linear trajectories (Henzinger, 1996). Lafferriere et al. proved that o-minimal hybrid systems always admit finite bisimulations, and presented other classes of hybrid systems with more complex dynamics for which finite bisimulations exist. As studied in (van den Dries and Miller, 1996), quite many useful theories are o-minimal. This makes o-minimal hybrid systems a good staring point, better than the initial candidates. Rather than extending them by more complex dynamics, we follow an orthogonal approach, of extending o-minimal hybrid systems them with probabilities.

We extend hybrid systems with linear differential equations with probabilities, which enables their use for getting probabilistic bounds on the performance of controllers and designs (**?**). Such bounds are required when it is impossible to meet the specification deterministically. We introduce *Probabilistic o-minimal hybrid systems*, which combine flows definable in an o-minimal structure with the probabilistic choices allowed in probabilistic hybrid systems. We show that probabilistic o-minimal hybrid systems have finite bisimulations, thus their reachability problem is decidable. Given a probabilistic o-minimal hybrid system $\mathcal{H}$, we construct a finite-state automaton $\mathcal{A}$ such that $\mathcal{H}$ and $\mathcal{A}$ are bisimilar. Verification of properties of $\mathcal{H}$ is then reduced to verification of properties of $\mathcal{A}$.

We suggest several applications to the stronger model of probabilistic o-minimal systems. One type of applications has to do with the initial goal of the work – using them for modeling and reasoning about probabilistic numerical schemes such as Monte Carlo methods (Hammersley and Handscomb, 1964) that have less complicated dynamics but are of probabilistic nature. A second type of applications has to do with the ability of probabilistic o-minimal systems to model complicated realistic dynamics, as demonstrated in Section 3.1. Due to the lack of space, many details are omitted, and can be found in (Auerbach, 2002).

2. UNDECIDABILITY

Newton's method is the most basic "search algorithm" of numerical analysis and scientific computation. Given a real-valued function, Newton's method searches iteratively for real numbers in which the function value is zero. For a function $g : \mathbb{R} \to \mathbb{R}$ and an initial point $n_0 \in \mathbb{R}$, the *Newton sequence for g and n_0* is a real-valued sequence $n_{i+1} = n_i - \frac{g(n_i)}{g'(n_i)}$, where $i = 0, 1, 2, \ldots$ and g' denotes the derivative of g. The *Newton-method convergence problem* is to decide whether Newton's method reaches close enough a zero of the function after a certain number of iterations. It is easy to reduce the Newton-method convergence problem to the reachability problem in a hybrid system with a single real-valued variable that corresponds to the intermediate values of the sequence. The *theory of real computation* described in (Blum *et al.*, 1997) implies that the convergence problem is undecidable. Our alternative proof reduce Newton convergence problem to the halting problem of a one-register machine (Minsky, 1967). It follows that the reachability problem for hybrid systems that model simple scientific-computation methods is undecidable. It also suggests an alternative proof for the undecidability result in (Alur and Dill, 1994).

3. PROBABILISTIC HYBRID SYSTEMS

A *hybrid system* is a computational model that has discrete and continuous components. The state space of a hybrid system consists of discrete and continuous elements. A state of a hybrid system changes according to both discrete and continuous transitions. A discrete transition resets some or all of the state elements. A continuous transition changes the continuous elements of the state according to a system of ordinary differential equations. Hybrid systems are used to model a wide range of application such as real-time circuits (Maler and Yovine, 1996), chemical processes (Engell *et al.*, 2000), air traffic management systems (Livadas *et al.*, 1999; Lygeros *et al.*, 1998*b*; Tomlin *et al.*, 1998), automated highway systems (Horowitz and Varaiya, 2000; Lygeros *et al.*, 1998*a*; Varaiya, 1993), robotics (Alur *et al.*, 2000*a*; Song *et al.*, 2000), biological processes (Ben-Jacob, 1997), and chemical process (Turk *et al.*, 1997). Surveys can be found in (Alur *et al.*, 2000*b*; Henzinger, 2002).

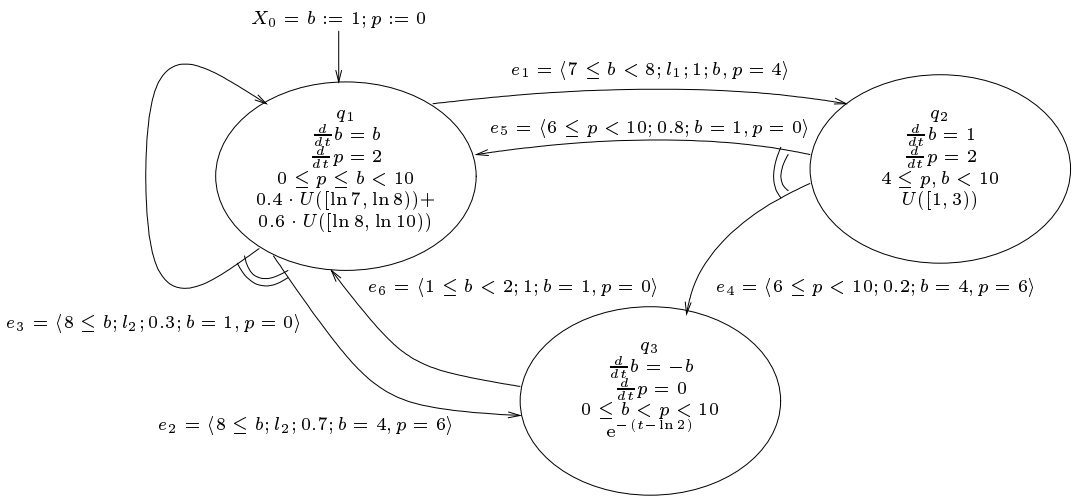

Fig. 1. The bacteria-growing system $\mathcal{H}$.

A *probabilistic* hybrid system is a hybrid system that decides probabilistically about its transitions. The system picks the next discrete transition and the duration of a continuous transition randomly. Probabilistic hybrid system are used in analysis of randomized distributed algorithms (Lehmann and Rabin, 1981), fault-tolerant systems, embedded systems, air traffic management (Hu *et al.*, 2000), traffic control (**?**), and more.

A general hybrid system can model almost every physical system. Yet, the expressiveness of hybrid systems is often impractical, as it involves reasoning about infinite state spaces. Probabilistic hybrid systems with linear differential equations that do not involve nondeterminism are shown to have constructible finite bisimulations (Lafferriere *et al.*, 1999), hence, verifying their properties is decidable. In particular, checking whether a subset of states of a probabilistic hybrid system with linear differential equations is visited infinitely often as time diverges is decidable.

3.1 Overview by example

An overview of our approach example drawn from the biotechnological field. We introduce a process of bacteria growing. A probabilistic hybrid system and a bisimilar finite-state automaton model the process.

The bacteria-growing process involves a control unit and three containers. Each container contains a solution of bacteria, food consumed by the bacteria, and poisons that bacteria produce. The food–poison rate in each container determines the growing conditions of the bacteria as follows.

The probabilistic hybrid system $\mathcal{H}$, which is shown in Figure 1, models the process. The system has three *discrete states* q_1, q_2, and q_3 that model the growing (rich in food and low in poisonous products), neutral (similar quantities of food and poinson), and decreasing (highly poisonous) containers, respectively. The real-value variables b and p denote the amount of bacteria and poison, respectively. The system *continuous state* is a two-dimensional real-valued vector that stores the bacteria and poison quantities. A state of $\mathcal{H}$ is a pair of a discrete state and a continuous state. For example, the *initial state* is $\langle q_1, \langle 1, 0 \rangle \rangle$, i.e., the solution is initially in the growing container, which has 1 unit of bacteria and no poison. The system state changes by transitions of two types as follows. A *continuous transition* changes the system continuous state. A *discrete transition* changes both the discrete and the continuous state. A path of the system is a sequence of alternating continuous and discrete transitions. Each discrete state has an *invariant set*, e.g., the invariant set of q_1 is the triangular set $\{\langle b, p \rangle \in \mathbb{R}^2 : 0 \le p \le b < 10\}$. The discrete state may be q_1 if the bacteria and poison quantities belong to the invariant set of q_1.

When the discrete state is q_1, a continuous transition occurs. During the transition continuous state of $\mathcal{H}$ changes according to the following ordinary differential equation. $\frac{d}{dt}\langle b, p \rangle = \langle b, 2 \rangle$. The r.h.s. of the differential equation is referred to as the *flow vector field* of q_1, denoted by F_q. The continuous state keeps changing according to the differential equation above until the *escape time* from q_1 is reached. Then, a discrete transition that resets both discrete state and continuous state occurs. The escape time from q_1 is probabilistically picked according to the density function $(q_1)(t) = 0.4 \cdot \mathrm{U}([\ln 7, \ln 8))(t) + 0.6 \cdot \mathrm{U}([\ln 8, \ln 10])(t)$. i.e., the probability to choose the escape time in the interval $[\ln 8, \ln 10]$ is $\int_{\ln 8}^{\ln 10} \mathrm{Dens}(q_1)(t)\,\mathrm{d}t = 0.6$. When the the escape time from q_1 is reached, $\mathcal{H}$ makes a discrete transition induced by either edges e_1, e_2, or e_3. The *guard set* of both edges e_2 and e_3 is $\{\langle b, p \rangle \in \mathbb{R}^2 : b \ge 8\}$. If the state belongs to the guard set, the edges e_2 and e_3 are *enabled*

and the system probabilistically picks each edge according to the edge *probability parameter*, i.e., e_2 is chosen with probability 0.7 and e_3 is chosen with probability 0.3. The *reset set* of e_2 is $\{\langle b, p\rangle \in \mathbb{R}^2 : b = 4 \text{ and } p = 4\}$. The state of $\mathcal{H}$ is reset to $\langle q_2, \langle 4, 4\rangle\rangle$ when the discrete transition induced by e_2 is taken.

For a discrete state q, the discrete state may stays q forever if the density function of q has an unboubnded support. For example, the density function of q_3 is the following exponential density function. $\text{Dens}(q_3)(t) = \mathrm{e}^{-(t-\ln 2)}$ if $t \geq \ln 2$ and zero otherwise. For every $t_1 \geq \ln 2$, the probability to choose an escape time that is greater than t_1 is positive. Thus, the discrete state may be q_3 forever. Some states are designated as *final states*, denoted by $X_F = \{q_3\} \times [0, 0.5) \times \{6\}$.

The system is *non blocking*, that is, all system paths let time diverge. For every state that is reachable form the initial state, there exists either discrete or continuous transition that changes the state.

The question of interest is as follows: does the system stay forever in the set of final states with positive probability? In order to answer the question, a finite-state automaton $\mathcal{A}$ that bisimulates the system $\mathcal{H}$ is constructed. The automaton $\mathcal{A}$, shown in Figure 2, is referred to as the *quotient automaton* of $\mathcal{H}$. The states of $\mathcal{A}$ are sets of states of $\mathcal{H}$, e.g.,the state X_{10} of the automaton $\mathcal{A}$ is the set X_F of final states of $\mathcal{H}$. The automaton has additional special states referred to as *gateway states*. A positive probability value is assigned to every edge of $\mathcal{A}$. The automaton $\mathcal{A}$ *bisimulates* the hybrid system $\mathcal{H}$ in the following sense. There is a positive-probability edge from a state X' to a state X'' of the quotient automaton $\mathcal{A}$ if and only if there exists a positive-probability transition from a state of $\mathtt{x}'$ of $\mathcal{H}$ that belongs to X' to a state $\mathtt{x}''$ of $\mathcal{H}$ that belongs to X''. Bisimilar systems have similar reachability properties. Thus, the hybrid system $\mathcal{H}$ stays forever in X_F with positive probability if and only if the automaton $\mathcal{A}$ stays forever in X_{10} with positive probability.

3.2 Probabilistic o-minimal hybrid systems

A probabilistic hybrid system is an initialized hybrid system (Alur *et al.*, 1995) augmented with edge probability parameters and density functions. The discrete transitions are chosen according to the edge probability parameters. The durations of the continuous transitions are chosen according to continuous density functions.

Definition 1. (Probabilistic hybrid system). A *Probabilistic hybrid system* is a tuple

$\mathcal{H} = \langle X_D, X_C, X_0, X_F, F, \text{Inv}, \text{Dens}, L, E\rangle$, where

- $X_D = \{q_1, \ldots, q_m\}$ is a set of discrete states.
- $X_C = \mathbb{R}^n$ is a set of continuous states. The system state space is $X = X_D \times X_C$.
- $X_0 \subseteq X$ is a set of initial states.
- $X_F \subseteq X$ is a set of final states.
- $F = \{F_{q_1}, \ldots, F_{q_m}\}$ is a flow set, each $F_{q_i} : \mathbb{R}^n \to \mathbb{R}^n$ is a vector field.
- $\text{Inv} : X_D \to 2^{\mathbb{R}^n}$ assigns to a discrete state an invariant set.
- Dens assigns to a discrete state q a continuous density functions $\text{Dens}(q) : \mathbb{R} \to \mathbb{R}_{\geq 0}$.
- $L = \{l_1, \ldots, l_N\}$ is a set of edge labels.
- $E \subseteq X_D \times X_D \times 2^{\mathbb{R}^n} \times 2^{\mathbb{R}^n} \times [0, 1] \times L$ is a set of edges. Each edge $e = \langle q, q', g, r, p, l\rangle$ consists of the following elements.
  - $q \in X_D$ is a source state of e.
  - $q' \in X_D$ is a target state of e.
  - $g \subseteq \mathbb{R}^n$ is a guard of e, which consists of all continuous sources of e.
  - $r \subseteq \mathbb{R}^n$ is a reset of e, which consists of all continuous targets of e.
  - $p \in [0, 1]$ is a probability of e.
  - $l \in L$ is a label of e.

For an edge e, the *predecessor* of e, denoted by $\text{Pre}(e)$, is the collection of precondition states of e, that is, all states whose discrete elements are source of e and their continuous elements belong to the guard of e. The *successor* of e, denoted by $\text{Post}(e)$, is the set of all states whose discrete elements are the target of e and continuous elements belong to the reset of e.

Discrete transitions involve both a nondeterministic and a probabilistic decisions as follows. A nondeterministic decision is made between classes of edges, which have the same edge labels. Once a class is chosen, an edge is chosen from that class according to the edge probability.

For a state $\mathtt{x} = \langle q, x\rangle$, the integral path of the flow vector field F_q and the point x is the path $\gamma_{\mathtt{x}} : \mathbb{R} \to \mathbb{R}^n$ such that $\frac{d}{dt}\gamma_{F\mathtt{x}}(t) = F(\gamma_F q(t))$ and $\gamma_{\mathtt{x}}(0) = x$. The target of a continuous transition that originates at $\mathtt{x}$ and take t time units is denoted by $\text{Target}(\mathtt{x}, t) = \langle q, \gamma_{\mathtt{x}}(t)\rangle$. The target state of a infinite-duration transition is denoted by $\text{Target}(\mathtt{x}, \inf)$.

Definition 2. (Probabilistic hybrid-system path). A path of a probabilistic hybrid system H is a finite or infinite sequence

$$\pi_H = \mathtt{x}_0 \xrightarrow{t_0} \mathtt{x}'_0 \xrightarrow{e_0} \mathtt{x}_1 \xrightarrow{t_1} \mathtt{x}'_1 \xrightarrow{e_1} \mathtt{x}_2 \xrightarrow{t_2} \cdots$$

Initial state $\mathtt{x}_0$ is in X_0 and for $i = 0, 1, 2, \ldots$, the following hold.

(1) The i-th primary state is $\mathtt{x}_i = \langle q_i, x_i\rangle \in X$, the *time-successor* state is $\mathtt{x}'_i = \langle q_i, x'_i\rangle \in X$,

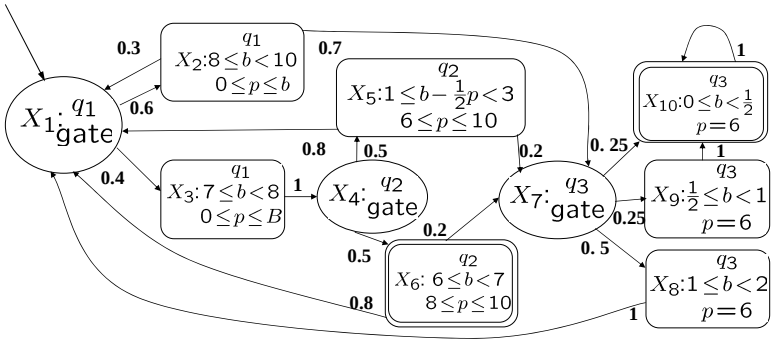

Fig. 2. The quotient automaton $\mathcal{A}$.

the edge is $e_i \in E$, and the escape time of the continuous transition is $t_i \in \mathbb{R}_{\ge 0} \cup \{+\infty\}$.

(2) The i-th time-successor state $\mathtt{x}'_\mathtt{i} i$ is the target of the continuous transition that originates at $\mathtt{x_i}$ and escapes from q_1 after t_i time units, i.e., $\mathtt{x}'_\mathtt{i} = \mathrm{Target}(\mathtt{x_i}, t_i)$ if t_i is finite and $\mathtt{x}'_\mathtt{i} = \mathrm{Target}(\mathtt{x_i}, \mathrm{inf})$ if $t_i = \infty$.

(3) The $\mathtt{x}'_\mathtt{i}$ enables the edge e_i and the state $\mathtt{x_{i+1}}$ belongs to the target region of e_i, i.e., $\mathtt{x}'_\mathtt{i} \in \mathrm{Pre}(e_i)$ and $\mathtt{x_{i+1}} \in \mathrm{Post}(e_i)$.

For example, the path $\langle q_1, \langle 1, 0\rangle\rangle \xrightarrow{\ln 7} \langle q_1, \langle 7, 2\ln 7\rangle\rangle \xrightarrow{e_1} \langle q_2, \langle 4, 4\rangle\rangle \xrightarrow{2} \langle q_2, \langle 6, 8\rangle\rangle \xrightarrow{e_4} \langle q_3, \langle 4, 6\rangle\rangle \xrightarrow{\infty}$ originates from the initial state of the bacteria-growing system $\mathcal{H}$ and stays forever in q_3.

Elementary questions about the behavior of hybrid systems are undecidable (Henzinger *et al.*, 1998). Undecidability follows from too complicated topology of the sets and functions of hybrid systems. O-minimal (order minimal) sets and functions have fine topological properties (van den Dries, 1998). An o-minimal subset of $\mathbb{R}$ is a finite union of intervals. A *bisimulation* is an equivalence relation on a state space of a transition system. Hybrid systems that have constructible finite bisimulations are decidable. O-minimal hybrid systems, whose sets and functions are o-minimal admit finite bisimulations (Lafferriere *et al.*, 2000). Hybrid systems with linear differential equations have constructible finite bisimulations (Lafferriere *et al.*, 1999).

Reasoning about the behavior of a hybrid system involves exploration of an infinite state space. For a probabilistic hybrid system with linear differential equations $\mathcal{H}$, consider the following qualitative question. Does $\mathcal{H}$ have a positive-probability path that reaches the set of final states and stays there forever? In order to decide the question, a finite-state automaton with edge probabilities $\mathcal{A}$ that satisfies the following requirements is constructed. The automaton $\mathcal{A}$ is an *abstraction* of $\mathcal{H}$, i.e., there is an equivalence relation $\sim \subseteq X_\mathcal{H} \times X_\mathcal{H}$ such that the state space of $\mathcal{A}$ is the quotient space $X_\mathcal{H}/\sim$. In addition, the automaton $\mathcal{A}$ probabilistically bisimulates the system $\mathcal{H}$, i.e., the set of initial and final states $X_0, X_F \in X_\mathcal{H}$ are unions of states of $\mathcal{A}$ and for states X', X'' of $\mathcal{A}$, there exists a positive-probability path in $\mathcal{A}$ from X' to X'' if and only if there exist states $\mathtt{x}' \in X'$ and $\mathtt{x}'' \in X''$ such that there exists a positive-probability path in $\mathcal{H}$ from $\mathtt{x}'$ to $\mathtt{x}''$.

The construction of the automaton $\mathcal{A}$ is as follows. The behavior of $\mathcal{A}$ is purely probabilistic and there are no nondeterministic choices–a positive real number is assigned to each of the edges of $\mathcal{A}$. The transition relation of $\mathcal{A}$ is complete, that is, paths of $\mathcal{A}$ never end and may reach a self loop. Constructing $\mathcal{A}$ involves constructing an equivalence relation $\sim \subseteq X_\mathcal{H} \times X_\mathcal{H}$ and defining the probability parameters. The bisimulation algorithm for hybrid systems with linear differential equations of (Lafferriere *et al.*, 1999) is used to construct the state space of $\mathcal{A}$. In addition, for each discrete state q of $\mathcal{H}$, an auxiliary state is added to $\mathcal{A}$, referred to as the *gateway of* q. The edges of $\mathcal{A}$ simulates the transitions of $\mathcal{H}$ as follows. An edges that enters a gateway state simulates a discrete transition of $\mathcal{H}$. An edge that leaves a gateway state simulates a continuous transition. A self loop simulates a continuous transition that stays forever in region.

Since the system $\mathcal{H}$ and the automaton $\mathcal{A}$ are bisimilar, there is a positive-probability path in $\mathcal{H}$ that stays forever in the final state of $\mathcal{H}$ if and only if there is a positive-probability a path of $\mathcal{A}$ that stays forever in the states of $\mathcal{A}$ that simulate the final states of $\mathcal{H}$. According to (Alur *et al.*, 1991; Vardi, 1999) if these states of $\mathcal{A}$ are a terminal maximal strongly connected component of $\mathcal{A}$ that is reachable from the initial state of $\mathcal{A}$ then the required path of $\mathcal{A}$ exists. This algorithm decides the question about probabilistic hybrid systems with linear differential equations.

Alur, R. and D. Dill (1994). A theory of timed automata. *TCS* **126**(2), 183–236.

Alur, R., C. Courcoubetis and D.L. Dill (1991). Model-checking for probabilistic real-time systems. In: Proc. 18th ICALP. LNCS 510, pp. 115–136.

Alur, R., C. Courcoubetis, N. Halbwachs, T. A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis and S. Yovine (1995). The algorithmic analysis of hybrid systems. *TCS* **138**(1), 3–34.

Alur, R., R. Grosu, Y. Hur, V. Kumar and I. Lee (2000*a*). Modular specification of hybrid systems in CHARON. In: *Hybrid Systems: Computation and Control, Proc. of the Third Int. Conf.*. Vol. 1790 of *LNCS*. pp. 6–19.

Alur, R., T. A. Henzinger, G. Lafferriere and G. J. Pappas (2000*b*). Discrete abstractions of hybrid system. *IEEE* **88**(2), 971–984.

Auerbach, Gadiel (2002). Formal analysis of mechanical system. Master's thesis. The Hebrew University.

Ben-Jacob, E. (1997). From snowflake formation to growth of bacterial colonies Part II: Cooperative formation of complex colonial patterns. *Contemporary Physics* **38**, 205–241.

Blum, L., F. Cucker, M. Shub and S. Smale (1997). *Complexity and Real Computation*. Springer.

Engell, S., S. Kowalewski, C. Schulz and O. Stursberg (2000). Simulation, analysis and optimization of continuous-discrete interactions in chemical processing plants. In: *Proc. of IEEE*.

Hammersley, J. M. and D. C. Handscomb (1964). *Monte Carlo Methods*. Methuen. London.

Henzinger, T. A. (1996). The theory of hybrid automata. In: *Proc. 11th LICS*. pp. 278–292.

Henzinger, T. A. (2002). Invited tutorial, The symbolic approach to hybrid systems. In: *Proc 14th CAV*. LNCS 2404.

Henzinger, T. A., P. W. Kopke, A. Puri and P. Varaiya (1998). What's decidable about hybrid automata?. *J. of Comput. and System Sci.* **57**(1), 94–124.

Horowitz, R. and P. Varaiya (2000). Control design of an automated highway system. *Proc. of IEEE: Special Issue on Hybrid Systems* **88**(7), 913–925.

Hu, J., J. Lygeros, M. Prandini and S. Sastry (2000). A probabilistic approach to aircraft conflict detection. *IEEE Trans. on Intelligent Transportation Systems. Special issue on Air Traffic Control - Part I.*

Lafferriere, G., G. J. Pappas and S. Youvine (1999). A new class of decidable hybrid systems. In: *Hybrid Systems: Computation and Control*. Vol. 1569 of *LNCS*. Springer. pp. 137–151.

Lafferriere, G., G.J. Pappas and S. Sastry (2000). O-minimal hybrid systems. *Math. Control Signals Systems* **13**, 1–21.

Lehmann, D. and M. Rabin (1981). On the advantages of free choice: A symmetric and fully distributed solution to the dining philosophers problem. In: *8th POPL*. pp. 133–138.

Livadas, C., J. Lygeros and N. A. Lynch (1999). High-level modeling and analysis of TCAS. In: *IEEE Real-Time Systems Symposium*. pp. 115–125.

Lygeros, J., D. Godbole and S. Sastry (1998*a*). A verified hybrid controller for automated vehicles. *IEEE Trans. on Automatic Control* **43**(4), 522–539.

Lygeros, J., G. J. Pappas and S. Sastry (1998*b*). An approach to the verification of the center-TRACON automation system. In: *Hybrid Systems: Computation and Control*. pp. 289–304.

Maler, O. and S. Yovine (1996). Hardware timing verification using KRONOS. In: *Proc. of the IEEE 7th Israeli Conference on Computer Systems and Software Engineering*. Israel.

Minsky, M.L. (1967). *Finite and Infinite Machines*. Prentice-Hall.

Prandini, M., J. Lygeros, A. Nilim and S. Sastry (1999). A probabilistic framework for aircraft conflict detection. In: *AIAA on Guidance Navigation and Control*.

Song, M., T-J. Tarn and N. Xi (2000). Intelligent scheduling, planning and control: Analytical integration of hybrid systems. In: *Proc. of the IEEE*. Vol. 88.

Tomlin, C., G. J. Pappas and S. Sastry (1998). Conflict resolution for air traffic management: A study in multiagent hybrid systems. *IEEE Transactions on Automatic Control* **43**(4), 509–521.

Turk, A. L., S. T. Probst and G. J. Powers (1997). Verification of a chemical process leak test procedure. In: *Proc 9th CAV*. LNCS 1254. pp. 84–94.

van den Dries, L. (1998). *Tame Topology and O-minimal Structures*. Cambridge Univ. Press.

van den Dries, L. and C. Miller (1996). Geometric categories and o-minimal structures. *Duke Math. J* **84**(2), 497–540.

Varaiya, P. (1993). Smart cars on smart roads: Problems of control. *IEEE Trans. on Automatic Control* **38**(2), 195–207.

Vardi, M. Y. (1999). Probabilistic linear-time model checking: An overview of the automata-theoretic approach. In: *ARTS: Int. Workshop on Formal Methods for Real-Time and Probabilistic Systems*. LNCS.