Volume 6, Number 1
January 2019

Published by the Association for Computing Machinery
Special Interest Group on Logic and Computation

SIGLOG news

TABLE OF CONTENTS

Association for Computing Machinery

Advancing Computing as a Science & Profession

VERIFICATION COLUMN

RANKO LAZIĆ, University of Warwick
R.S.Lazic@warwick.ac.uk

In this timely and comprehensive article, the authors provide an insightful overview of the lively field of automatic verification of hybrid systems. They focus on the fundamental problem of reachability, and consider both exact and approximate approaches. I am sure that Oded Maler, to whom this contribution is very fittingly dedicated, would have been proud to see such an authoritative and useful summary of the state of the art of an area he had helped introduce.

In Memory of Oded Maler: Automatic Reachability Analysis of Hybrid-State Automata

Martin Fränzle
Department of Computing Science
Carl von Ossietzky Universität Oldenburg

Mingshuai Chen
State Key Lab. of Computer Science
Institute of Software, CAS & UCAS

Paul Kröger
Department of Computing Science
Carl von Ossietzky Universität Oldenburg

Hybrid automata are an elegant formal model seamlessly integrating differential equations representing continuous dynamics with automata capturing switching behavior. Since the introduction of the computational model more than a quarter of a century ago [Maler et al. 1992], its algorithmic verification has been an area of intense research. Within this note, which is dedicated to Oded Maler (1957–2018) as one of the inventors of the model, we are trying to delineate major lines of attack to the reachability problem for hybrid automata. Due to its relation to system safety, the reachability problem is a prototypical verification problem for hybrid discrete-continuous system dynamics.

1. INTRODUCTION

Hybrid discrete-continuous dynamic behavior arises when discrete and continuous dynamic processes become connected, as in the case of embedded computers interacting with their physical environment via sensors and actuators. Such interaction may be complex and safety-critical, having sensitive variables of the environment in its sphere of control. Everyday examples include process control at all scales, ranging from household appliances to nuclear power plants, or embedded systems in the transportation domain, such as autonomous driving maneuvers in automotive, aircraft collision avoid-

ance protocols in avionics, or automatic train control applications, as well as a broad range of devices in health technologies, like cardiac pacemakers. The resulting close interaction of computational devices and digital computing has motivated the terms hybrid system or cyber-physical system.

As the behavior of such hybrid discrete-continuous systems may be safety-critical and as it cannot be fully understood without explicitly modeling and analyzing the tight interaction of their discrete switching behavior and their continuous dynamics, models and algorithms for their behavioral verification have been suggested. Verification here amounts to showing that the coupled dynamics of the embedded system and its environment is well-behaved, regardless of the actual disturbance and the influences of the application context, as entering through the open inputs of the system under investigation. Basic notions of being well-behaved demand that the system under investigation may never reach an undesirable state (*safety*), that it will converge to a certain set of states (*stabilization*), or that it can be guaranteed to eventually reach a desirable operational state (*progress*). In the sequel, we will concentrate on reachability or, dually, safety as a prototypic verification goal.

As a formal model of hybrid dynamics facilitating rigorous analysis at design level, the model of hybrid automata has been suggested [Maler et al. 1992; Alur et al. 1993]. In order to facilitate a clean semantics, this model adopts an alternation between discrete actions and continuous evolutions, interleaving durational continuous evolutions with immediate discrete transitions. While providing a clean semantics, this semantic idealization also induces extraneous dynamics that has impact on the possibility of automatic key-press analysis of hybrid systems. A plethora of positive and negative decidability or semi-decidability results has been obtained on this idealized model and a broad range of approximation procedures has been devised. Within this note, we are trying to group the various, mostly historic developments —and we hereby apologize to all colleagues who should have been cited, yet could not due to lack of space— and comment on in how far they rely on apparent artifacts of the idealized formal model rather than inherent properties of the problem domain.

2. HYBRID AUTOMATA: A FORMAL MODEL

We start our investigation by providing a formalization of hybrid discrete-continuous systems by hybrid automata (introduced in [Maler et al. 1992] and given their name in [Alur et al. 1993]), which are finite automata extended with a vector of continuous variables and "decorated" with ordinary differential equations (ODEs) in each location plus assignments to these extra variables upon transitions. In the qualitative case, a *hybrid automaton* is a tuple $\mathcal{H} = (V, X, m, f, Init, Inv, Jump)$, where:

— V is a *finite* set of discrete modes. The elements of V represent the discrete states of the automaton.
— $X = \{x_1, \ldots, x_n\}$ is an (ordered) finite set of continuous variables. A real-valued valuation $\vec{z} \in \mathbb{R}^n$ of $x_1, \ldots, x_n$ represents a continuous state. The overall state space of the automaton is the Cartesian product of the discrete and the continuous state space, i.e. is $V \times \mathbb{R}^n$. We call n the *dimension* of the hybrid automaton.
— $f \in V \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ with $m \in \mathbb{N}$ assigns a vector field with input in $\mathbb{R}^m$ to each mode. The dynamics in mode $v \in V$ is $\frac{\mathrm{d}\vec{x}}{\mathrm{d}t} = f(v, \vec{x}, \vec{u})$, where $\vec{u} \in \mathbb{R}^m$ is an uncontrolled input (which may be absent in case of input-free ODEs, in which case $m = 0$). A straightforward variant adopts a set-valued function $f \in V \times \mathbb{R}^n \times \mathbb{R}^m \to \left(2^{\mathbb{R}^n} \setminus \emptyset\right)$, leading to a differential inclusion $\frac{\mathrm{d}\vec{x}}{\mathrm{d}t} \in f(v, \vec{x}, \vec{u})$.
— $Init \subseteq V \times \mathbb{R}^n$ is the initial condition. $Init$ defines the admissible initial states of $\mathcal{H}$, i.e. those states that runs of the automaton may start in.

— $Inv \subseteq V \times \mathbb{R}^n$ specifies the mode invariants. Inv defines the admissible states of $\mathcal{H}$, i.e. those states that runs of the automaton may traverse through. We denote by $Inv(v)$ the set $\{\vec{x} \mid (v, \vec{x}) \in Inv\}$, called v's mode invariant.
— $Jump \in V \times \mathbb{R}^n \to \mathcal{P}(V \times \mathbb{R}^n)$ is the jump relation.
$Jump$ defines the possible discrete actions of $\mathcal{H}$. The jump relation may be non-deterministic and covers both discrete modes and continuous variables. (For simplicity, we omit inputs in jumps.) It is customary to write jumps as a set of transitions consisting of a source and a target mode plus a pair of an enabling guard condition and a corresponding update.
The interpretation is that whenever $Jump(v, \vec{x}) = \emptyset$ then no discrete jump is enabled from the current state $(v, \vec{x})$. If $Jump(v, \vec{x}) \neq \emptyset$ then a jump is possible and *may* be taken, leading to one or more (if $|Jump(v, \vec{x})| > 1$) possible successor states. There is no requirement that an enabled jump has to be taken: if the invariant permits to stay in the current mode while following the differential equation pertinent to the mode then an enabled transition does not need to be taken.

Given an input $u : \mathbb{R}_{\geq 0} \to \mathbb{R}^m$, a *run* of $\mathcal{H}$ over u is a (finite or infinite) sequence $r = \langle (v_0, g_0), (v_1, g_1), \ldots \rangle \in \left(V \times C^1\right)^{*|\omega}$, where C^1 is the set of differentiable functions $g : I \to \mathbb{R}^n$ which have a (bounded or unbounded) closed interval $I \subseteq \mathbb{R}_{\geq 0}$ as domain, with the sequence satisfying the following three conditions:

(1) *Initialization:* $\min \mathrm{dom}(g_0) = 0$ and $(v_0, g_0(0)) \in Init$, i.e. the run starts in time instant 0 and the start point is in the initial state set,
(2) *Consistency with continuous dynamics:* For each $i \leq \mathrm{len}\, r$, where $\mathrm{len}\, r$ is the length of the run r, the trajectory segment g_i satisfies

$$\forall t \in \mathrm{dom}(g_i) \ : \ \frac{\mathrm{d}g_i}{\mathrm{d}t}(t) = f(v_i, g_i(t), u(t)) \ , \tag{1}$$

$$\forall t \in \mathrm{dom}(g_i) \ : \ (v_i, g_i(t)) \in Inv \ . \tag{2}$$

Each segment g_i of the trajectory thus is, first, a solution of the input differential equation associated to the mode v_i held during this time period and, second, remains within the invariant associated to the mode. The extension to differential inclusions is straightforward.
(3) *Consecution:* For each $i < \mathrm{len}\, r$,

$$\max \mathrm{dom}(g_i) \ = \ \min \mathrm{dom}(g_{i+1}) \ , \tag{3}$$

$$(v_{i+1}, \min \mathrm{dom}(g_{i+1})) \ \in \ Jump(v_i, g_i(\max \mathrm{dom}(g)_i)) \ . \tag{4}$$

I.e., the trajectory segments are consecutive in the sense that segment g_{i+1} starts at the same time instant that g_i ends and, furthermore, that trajectory segments are connected by jumps in the sense that the endpoint of segment g_i, together with the mode v_i pertinent to this trajectory segment, is connected to the start point of segment g_{i+1} plus its mode v_{i+1}. Note that jumps may follow each other immediately as trajectory segments may have point intervals as their domain.

We say that $r = \langle (v_0, g_0), (v_1, g_1), \ldots \rangle \in \left(V \times C^1\right)^{*|\omega}$ is a run of $\mathcal{H}$ iff there is an input u —possibly from some restricted class of input signals— such that r is a run of $\mathcal{H}$ over u. Let $\mathrm{dur}\, r = \sup_{i \leq \mathrm{len}\, r} \sup \mathrm{dom}(g_i)$ denote the duration of r. We say that run $r = \langle (v_0, g_0), (v_1, g_1), \ldots \rangle \in \left(V \times C^1\right)^{*|\omega}$ visits state $s \in V \times \mathbb{R}^n$ iff there is an index $i \leq \mathrm{len}\, r$ and $t \in \mathrm{dom}(g_i)$ such that $(v_i, g_i(t)) = s$. We call state s reachable iff there is a run r visiting s.

Particular intricacies of the above model are that hybrid automata evolve over continuous time, are obviously infinite state, feature non-determinism in jumps and pos-

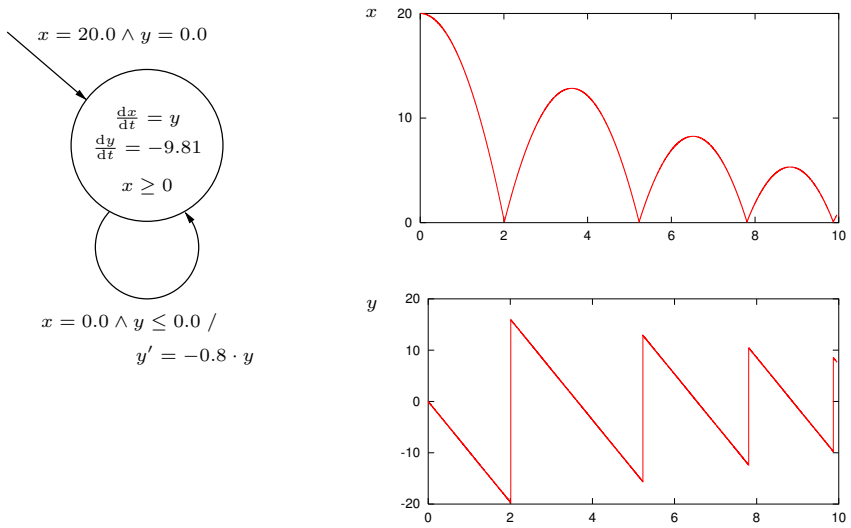

Fig. 1: A simple hybrid automaton (left) and a run thereof (right).

sibly in continuous evolutions due to open inputs, and that they involve a feedback between the continuous and the discrete dynamics, as apparent from conditions (1) and (4) which connect the endpoints of continuous evolutions to discrete jumps and vice versa. See Fig. 1 for a simplistic example of a hybrid automaton (in an obvious graphical notation) and a sample run thereof.

Among the dynamic properties of interest to engineers is the question whether a hybrid system may ever reach an undesirable, hazardous state. While other properties, like convergence and stability are also of interest, we do in this note concentrate on such reachability questions and therefore define the following set of decision problems:

— *Reachability:* Given a set $S \subset V \times \mathbb{R}^n$, sometimes called the *target*, decide whether some state $s \in S$ is reachable.
— *Step-bounded reachability:* Given a step bound $k \in \mathbb{N}$ and a set $S \subset V \times \mathbb{R}^n$, decide whether S is reachable within k steps, i.e., whether there is a run r of length $\operatorname{len} r \leq k$ visiting some state $s \in S$.
— *Horizon-bounded reachability:* Given a time horizon $t \in \mathbb{R}_{\geq 0}$ and a set $S \subset V \times \mathbb{R}^n$, decide whether S is reachable within time t, i.e., whether there is a run r of duration $\operatorname{dur} r \leq t$ visiting some state $s \in S$.
— *(Step-bounded) Mode reachability:* Given a mode $v \in V$ (and a bound $k \in \mathbb{N}$, resp.), decide whether $\{v\} \times \mathbb{R}^n$ is reachable (reachable within k steps, resp.).
— *Pareto-dominant reachability:* Given a mode $v \in V$ and a conjunction of simple bounds (one per continuous variable) $\phi = \bigwedge_{i=1}^{n} x_i \sim_i n_i$, where $\sim_i \in \{\leq, \geq\}$ and $n_i \in \mathbb{N} \cup \{\pm\infty\}$, decide whether a state $s = (v, \vec{x})$ is reachable such that $\vec{x}$ satisfies ϕ.

For full-fledged hybrid automata permitting appropriate guard conditions, invariants, and state updates, the unbounded versions of the above decision problems can mutually be reduced to each other, as can the bounded versions. This does not apply necessarily to subclasses of hybrid automata restricting guards, invariants, and/or updates. There, the latter, relaxed conditions sometimes induce inherently different decision problems. The subclasses we are going to address in the sequel are the following:

— *Linear Hybrid Automata (LHA):* A linear hybrid automaton is a hybrid automaton where transition guards and mode invariants are given by linear inequation systems (with rational coefficients) over X, transition updates are given by (rational coefficient) linear inequation systems over X and X', where X' denotes values of the variables in X after the jump, and the inclusion f defining the vector field is constant for each mode in that for each mode v_i we have $f(v_i, \cdot, \cdot) \in p_i$ for some mode-dependent polyhedron $p_i \subset \mathbb{R}^n$ with rational vertices.
— *Piecewise Constant Derivative Systems (PCDs)* are hybrid automata without jumps (i.e., $Jump \equiv \emptyset$) and with just a single mode (i.e., $|V| = 1$), yet a piecewise constant (rational-valued) vector field f defined over a finite polyhedral partition $P = \{p_1, \ldots, p_m\}$ of the $\mathbb{R}^n$, where the vertices of the polyhedra are rational. That is, $\bigcup_{i=1}^{m} p_i = \mathbb{R}^n$ and $\forall i, j \in \{1, \ldots, m\} : p_i \cap p_j = \emptyset$ holds and there are rational constants c_1 to c_m such that $f(\cdot, x, \cdot) \equiv c_i$ whenever $x \in p_i$.
— *Rectangular Automata (RA)* are linear hybrid automata where all guards, updates, mode invariants, and dynamics are rectangular regions. A *rectangular region* is the Cartesian product of a set of possibly unbounded real-valued intervals with rational endpoints. We call a rectangular automaton *initialized* iff the updates in the jumps do only depend on modes, not on the continuous pre-state of the jump, i.e., iff $Jump(v, \vec{x}_1) \cap (\{v'\} \times \mathbb{R}^n) \neq \emptyset$ and $Jump(v, \vec{x}_2) \cap (\{v'\} \times \mathbb{R}^n) \neq \emptyset$ implies $Jump(v, \vec{x}_1) \cap (\{v'\} \times \mathbb{R}^n) = Jump(v, \vec{x}_2) \cap (\{v'\} \times \mathbb{R}^n)$.
— *Timed Automata (TA)* are linear hybrid automata where all continuous have slopes 1 throughour, i.e., $f(v, \cdot, \cdot) \equiv 1$ for each mode v, and all guards and invariants are conjunctions of simple bounds $x_i \sim_i k_i$ with $\sim_i \in \{\leq, \geq\}$[1] and $k_i \in \mathbb{N} \cup \{\infty\}$. All updates are conjunctions of predicates of the two possible forms $x' = x$ (retain value of x) or $x' = 0$ (reset x to 0). Continuous variables with slope 1 are generally called *clocks*. The usual convention is that clocks are initialized to 0, i.e. $Init \subseteq V \times \{\vec{0}\}$. A *multirate TA* is a generalized timed automaton where $f(\cdot, x_i, \cdot) = k_i \in \mathbb{N}_{\geq 0}$, i.e. different clocks may have different, yet constant slopes.
— *Stopwatch Automata (SWA)* are akin to timed automata, but allow to switch slope for each continuous variable individually between 0 and 1 when changing mode, i.e., for each mode v and each continuous variable x_i either $f_i(v, \cdot, \cdot) \equiv 1$ or $f_i(v, \cdot, \cdot) \equiv 0$ applies. The restrictions to guards, invariants, and updates are as in TA. The two slopes permit starting and stopping clocks depending on the current mode; hence the name *stopwatch* for the resulting continuous devices.
— *Multi-priced Timed Automata (MPTA)* are linear hybrid automata where the vector of continuous variables is partitioned into clocks and price observers. For clocks, the same rules as in timed automata apply: they have slope 1 throughout, can appear in conjunctions of simple bounds defining guards and invariants, and may only be reset to 0 or maintain their value upon transitions. *Price observers* may take arbitrary constant rates in each mode, as the continuous variables in linear hybrid automata do, but can neither be queried in invariants or guards nor reset; instead they maintain their values across transitions: $c' = c$ holds for any observer on any transition.[2] As in timed automata, all continuous variables start at 0, regardless of whether they are clocks or price observers. If the slopes being charged for price observers are 0 or 1 throughout then we call the MPTA a *stopwatch-MPTA*.

[1] Strict bounds can be admitted, but are not relevant to this article.
[2] Additive updates of price observers on transitions can be admitted, but do not change expressiveness as they can be simulated by prices charged during additional continuous evolutions of finite duration.

3. UNDECIDABILITY ISSUES

Undecidability may be considered an obstacle to full automation of verification, as it confines the sound key-press procedures to either semi-decision procedures or even approximation schemes. In that respect, the many different causes for undecidability of reachability problems in hybrid automata are daunting.

Given the components available in a full-fledged hybrid automaton, it should come as no surprise that its reachability problem is undecidable. This is easy to see when the continuous variables have unbounded range (in the sense of invariants not generally confining them), as there are obviously numerous ways to implement increment and decrement operations on continuous variables with the help of deterministic differential dynamics. One way is to let time pass for one time unit —which can easily be controlled by the use of a clock variable— while imposing constants slopes of 1 or -1, resp., for the variable to increment or decrement, and while fixing the values of other variables by imposing a slope of 0. Add to this guards testing for 0 and you get a straightforward encoding of two-counter machines [Wilke 1994]. These and related encodings of two-counter machines, e.g. by stopwatch automata encoding integers by differences between pairs of continuous variables and resorting to an additional equality test between stop-watches for implementing the 0 test, have however been criticized due to their need for unbounded range of continuous variables. Unbounded variables are either thought to lack physical interpretation at all or at least to be atypical and thus of limited practical interest in engineering, given for example the fixed geometric extent, and thus bounded reachable space, of a robot.

This criticism has been addressed in a sequence of refined undecidability results that encode integers with continuous variables of bounded range. Such encodings can be implemented using surprisingly simple subclasses of hybrid automata: using K. Cerans' wrapping construction [Henzinger et al. 1998] that permits to exactly copy clock (or stopwatch) values between bounded clocks (or bounded clocks and stopwatches) in timed or stopwatch automata, SWA gadgets doubling or halving a clock value can be implemented, such that two-counter machines can be encoded using the mapping $(c_1, c_2) \mapsto (2^{-c_1}, 2^{-c_2})$ of counter-value pairs to clock values [Henzinger et al. 1998]. Variants of this scheme have been developed for various other fragments of hybrid automata featuring just clock- or stopwatch-like continuous dynamics. Recently it has been shown in [Fränzle et al. 2018] that even Pareto-dominant reachability in bounded stopwatch-MPTA suffices for encoding Diophantine equations by making use of vectors of reciprocals $\left({n_1}^{-1}, \ldots, {n_m}^{-1}\right)$ encoding the values of a vector of positive integer variables $(n_1, \ldots, n_m)$. These results show that even the reachable state sets of confined hybrid automata just featuring stop-watch dynamics can be amazingly complex.

The physical interpretation of undecidability results exploiting encodings of unbounded integers by bounded continuous variables, however, is as questionable as that of the encodings employing unbounded continuous range. As encodings of integers in a compact subset of the $\mathbb{R}^n$ need to feature at least one accumulation point, such encodings would not be robust against even the smallest amount of noise. This has sparked a search for robust, noise-resistant variants of the hybrid automaton model, which we will discuss in Section 5. The hope that such automata models may yield decidable reachability problems has, however, remained mostly elusive.

Given that the automata classes in the aforementioned undecidability results invariably used very simple continuous behavior of stopwatch shape, their undecidability obviously hinges on the interplay between their (simple) continuous dynamics and the state updates within transitions. Numerous investigations have consequently been pursued concerning the impact of restricted state updates within transitions on decidability. Again it turns out that even very restricted classes still feature an undecidable

reachability problem, among them piecewise constant derivative systems in three or more dimensions [Asarin et al. 1995]. This shows that already with simple piecewise constant dynamics, neither state updates nor mode-dependent behavior are crucial for undecidability; finitely many discontinuities in the vector field suffice. It should finally be noted that when permitting non-linear continuous dynamics, undecidability even arises with purely continuous dynamics, as is known from the undecidability of reachability in the three-body problem [Smith 2006].

4. DECIDABLE CASES

Despite hybrid automata dynamics being prone to undecidability, interesting subclasses have been identified which facilitate decision procedures for reachability problems. We provide a brief survey of some of those subclasses and the restrictions they impose on the type of dynamics.

4.1. Decidable unbounded reachability problems

Timed automata. The most widely known subclass, with mature verification tools being available [Behrmann et al. 2011], is the class of timed automata [Alur and Dill 1990]. Location reachability (and with it, by an obvious encoding via transition guards, reachability of any rectangular or diagonally bounded state set) has originally been shown decidable by establishing a finite time-abstract bisimulation quotient known as the region graph [Alur and Dill 1990]. Scalability has later been enhanced by encoding sets of reachable clock states in difference logic and applying widening to enforce termination [Larsen et al. 1995]. Note that due to the widening applied when clocks exceed the maximum constants mentioned in guards of the timed automaton, these algorithms do not compute the exact reachable state-set of the TA, but an overapproximation that is neutral on location reachability. Algorithms computing the exact reach set (or rather the exact binary reachability relation) by effectively constructing a formula of first-order linear arithmetic over the reals and integers have been contributed by [Comon and Jurski 1999] based on a syntactic transformation of the TA flattening out nested loops and by [Quaas et al. 2017] providing a direct symbolic procedure.

Results for timed automata carry over to *multirate automata*, as these can be translated into a TA by normalizing the clock rates and correspondingly scaling the constants appearing in guards and invariants, thereafter eliminating fractional constants by multiplication with a common denominator. These transformations are neutral on location reachability and have the obvious scaling effect on the continuous reach set.

Multi-priced timed automata. Multi-priced timed automata lie at the frontier between timed automata, featuring a decidable reachability problem (see above) and linear hybrid automata, for which reachability is undecidable (Section 3). While reachability is known to be undecidable even for stopwatch-MPTA [Fränzle et al. 2018], some positive decidability results exist nevertheless for monotonically price-charging MPTA. These positive results are based on appropriately constraining the shapes of the reachability target set or the dimensionality of the MPTA. If one restricts interest to Pareto-dominant reachability and furthermore confines the Pareto targets $\phi = \bigwedge_{i=1}^{n} x_i \sim_i n_i$ to either contain only upper bounds ($\sim_i = \leq$ for all i) or only lower bounds ($\sim_i = \geq$ for all i) then Pareto-dominant reachability becomes decidable for stopwatch-MPTA [Fränzle et al. 2018] and for monotonic (i.e., all slopes of cost observers are non-negative) MPTA under appropriate price-divergence and run-confluence conditions [Larsen and Rasmussen 2008]. Under Pareto targets comprising both types of bounds, only low-dimensional MPTA are known to be decidable: [Fränzle et al. 2018] show that for stopwatch-MPTA with up to three price observers (yet an arbitrary number of clocks),

Pareto-dominant reachability is decidable. For higher dimensions, only strong approximation procedures do currently exist (see Section 5).

Planar piecewise constant derivative systems. Another subclass of hybrid automata for which the reachability problem is decidable are planar piecewise constant derivative systems, i.e. PCDs of at most dimension 2 [Asarin et al. 1995]. The bound here is sharp as reachability is undecidable from dimension 3.

More recently, decidability results were achieved for a more general class, namely *simple monotonic planar linear hybrid systems* [Prabhakar et al. 2015], which relax the restrictions of planar PCDs by allowing discrete modes and polyhedral guards and invariants as mode switching conditions. Under certain conditions, namely a restriction to monotonic vector fields, reachability is decidable in the 2-dimensional case while undecidability is known from dimension 4 only.

Initialized rectangular automata. The reachability problem is decidable for initialized rectangular automaton, i.e. rectangular automata where across a transition, the slope of a continuous variable may only change if the transition simultaneously resets that variable [Henzinger et al. 1998]. The key point here is that an initialized RA can be translated into a multirate timed automaton where each variable x_i is represented by two variables $x_{i,\ell}$ and $x_{i,u}$ s.t. $x_{i,\ell}$ and $x_{i,u}$ track the lower (upper, resp.) bound of the possible valuations of x_i.

Decidable families of vector fields. Investigations concerning decidable reachability problems in continuous dynamics obviously started from the case of *linear differential equations*. [Lafferriere et al. 2001] investigated vector fields of the form $\frac{\mathrm{d}\vec{x}}{\mathrm{d}t} = A\vec{x} + B\vec{u}$, where $\vec{x}(t) \in \mathbb{R}^n$ is the state of the system at time t, $A \in \mathbb{R}^{n\times n}$ is the system matrix, and the input $\vec{u} : \mathbb{R}_{\geq 0} \to \mathbb{R}^m$ is a piecewise continuous function. By reduction into Tarski's algebra [Tarski 1951] they obtained the decidability of the reachability problems of the following three families of vector fields: (1) A is *nilpotent*, i.e., $A^n = 0$, and each component of $\vec{u}$ is a polynomial; (2) A is *diagonalizable* with rational eigenvalues, and each component of $\vec{u}$ is of the form $\sum_{i=1}^{m} c_i \mathrm{e}^{\lambda_i t}$, where λ_is are rationals and the c_i are subject to semi-algebraic constraints; (3) A is diagonalizable with purely imaginary eigenvalues, whose imaginary parts are rationals, and each component of $\vec{u}$ of the form $\sum_{i=1}^{m} c_i \sin(\lambda_i t) + d_i \cos(\lambda_i t)$, where λ_is are rationals and c_is and d_is are subject to semi-algebraic constraints.

[Anai and Weispfenning 2001] and [Gan et al. 2015; Gan et al. 2016] have subsequently relaxed some of the above conditions by exploiting reductions to the transcendental implicitization problem [Becker et al. 1993; Manocha and Canny 1992] or to the extension of Tarski's algebra with polynomial-exponential functions as well as exploiting density results in number theory [Hardy et al. 1979] within the reduction. These extensions eliminate the restriction to rational eigenvalues and coefficients in the above results, yielding decidability of the cases (2') A is diagonalizable with *real* eigenvalues, and each component of $\vec{u}$ is of the form $\sum_{i=1}^{m} c_i \mathrm{e}^{\lambda_i t}$, where λ_is are *reals* and c_is are subject to semi-algebraic constraints and (3') A is diagonalizable with purely imaginary eigenvalues, whose imaginary parts are *reals*, and each component of $\vec{u}$ is of the form $\sum_{i=1}^{m} c_i \sin(\lambda_i t) + d_i \cos(\lambda_i t)$, where λ_is are *reals* and c_is and d_is are subject to semi-algebraic constraints.

Positive decidability results concerning reachability over *non-linear vector fields* are naturally more scarce. Most of them hinge on the notion of *solvability* first proposed in [Rodríguez-Carbonell and Kapur 2007] for a class of polynomial programs, and extended to dynamical and hybrid systems in [Xu et al. 2013]. Formally, a dynamical system $\frac{\mathrm{d}\vec{x}}{\mathrm{d}t} = f(\vec{x}, \vec{u}(t))$ is called solvable if the state vector $\vec{x} = (x_1, \ldots, x_n)$ can be classified into $k \leq n$ groups $\chi_1 = (x_{11,}, \ldots, x_{1,n_1}), \ldots, \chi_k = (x_{k,1}, \ldots, x_{k,n_k})$ and the

dynamical system can be represented in the form

$$\frac{\mathrm{d}\vec{x}}{\mathrm{d}t} = \begin{bmatrix} \frac{\mathrm{d}\chi_1}{\mathrm{d}t} \\ \frac{\mathrm{d}\chi_2}{\mathrm{d}t} \\ \vdots \\ \frac{\mathrm{d}\chi_k}{\mathrm{d}t} \end{bmatrix} = \begin{bmatrix} A_1\chi_1 + u_1(t) \\ A_2\chi_2 + u_2(t,\chi_1) \\ \vdots \\ A_k\chi_k + u_k(t,\chi_1,\ldots,\chi_{k-1}) \end{bmatrix}, \tag{5}$$

where $0 < n_1 < \ldots < n_k = n$ are integers, $k \in \mathbb{N}$, $A_1, \ldots, A_k$ are real matrices with corresponding dimensions, and $u_1, \ldots, u_k$ are *polynomial-exponential-trigonometric functions*. Intuitively, an n-dimensional dynamical system is solvable if its state variables can be partitioned into k groups such that the derivatives of the variables in the ith group are linear in themselves, yet potentially non-linear in the variables from the preceding groups.

[Gan et al. 2018] identified three families of solvable vector fields and proved the decidability of reachability problems thereof: (1") $A_1, \ldots, A_k$ in Eq. (5) are nilpotentand each component of u_i is a polynomial. (2") Each A_i is diagonalizable with real eigenvalues, and each component of u_i is of the form $\sum_{j=1}^{m_i} c_{ij}\mathrm{e}^{\lambda_{ij}t}$, where λ_{ij}s are reals and c_{ij}s are subject to semi-algebraic constraints, with $i = 1, \ldots, k$. (3") Each A_i is diagonalizable with purely imaginary eigenvalues, whose imaginary parts are reals, and each component of u_i is of the form $\sum_{j=1}^{m_i} c_{ij}\sin(\lambda_{ij}t) + d_{ij}\cos(\lambda_{ij}t)$, where λ_{ij}s are reals and c_{ij}s and d_{ij}s are subject to semi-algebraic constraints. The known decidable classes of solvable non-linear systems are thus subject to equivalent constraints as the known decidable classes of linear systems, despite the added expensiveness due to hierarchical polynomial-exponential-trigonometric interdependencies.

o-minimal hybrid systems. Similar results as the above have been obtained on hybrid systems rather than just vector fields. Restricting the permitted differential dynamics to (mode-wise) linear vector fields $f(m, \vec{x}) = A_m\vec{x}$, where A is an $n \times n$-matrix of rationals and requiring transitions to always *re-initialize continuous state* while permitting polynomial guard conditions, [Lafferriere et al. 1999] have shown that the reachability problem is decidable if the matrices A_m are of one of the forms (1^*) A is nilpotent, or (2^*) A is diagonalizable with rational eigenvalues, or (3^*) A has diagonal real Jordan form and purely imaginary eigenvalues. These results can, however, not be carried over to an un-initialized setting where continuous variables maintain their values across transitions.

4.2. Decidable bounded reachability

Given the almost universal undecidability of the unbounded reachability problem, the problem of bounded reachability, though less informative due to its step- or time-bounded horizon, becomes an attractive alternative verification target. By saving a fixed-point computation, bounded problems are generally more amenable to reduction into finite constraint formulae in decidable logic fragments.

Bounded model checking using decidable logic.. Following the success of SAT-based bounded model checking (BMC) for finite-state systems [Groote et al. 1995; Biere et al. 1999], it has been observed that similar encodings of step-bounded reachability properties would be possible for linear hybrid automata (LHA) via a reduction to mixed-integer linear programming [Bemporad and Morari 1999] or LinSAT, i.e. the quantifier theory of linear arithmetic as solved by SAT-modulo-theory (SMT) solvers [Audemard et al. 2005]. As LinSAT is decidable and the reduction is exact, this provides a decision procedure for step-bounded reachability in LHA and consequently, by checking successively larger step bounds, also a semi-decision procedure for unbounded reachability

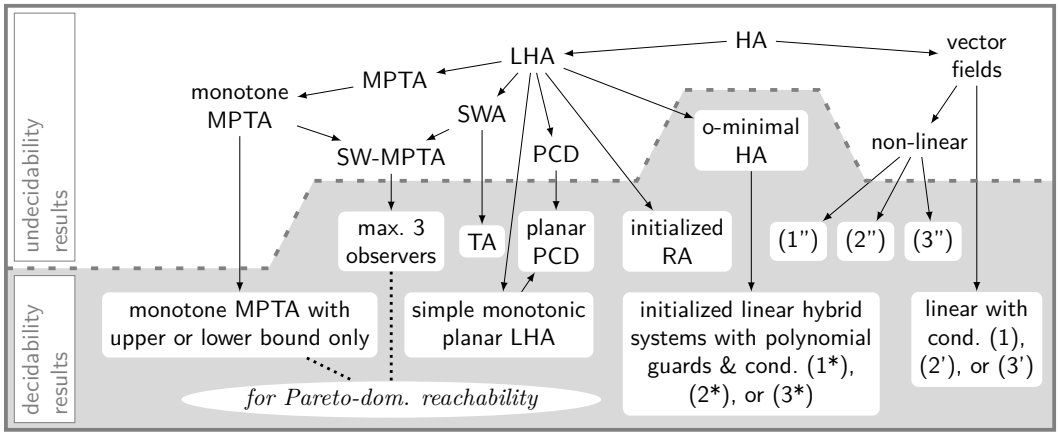

Fig. 2: Decidability of the reachability problem for subclasses of hybrid automata

in LHA. Using finite unions of polytopes for representing finitely reachable state sets instead of a reduction to arithmetic constraint formulae, a related procedure has earlier been implemented in [Henzinger et al. 1992]. All these procedures do obviously also apply to any subclass of LHA, like MPTA or PCD.

A generalization to more general hybrid automata than LHA is in principle straightforward as long as one sticks to a decidable fragment of arithmetic. The classes (1) to (3) of linear differential equations exposed above paired with polynomial guards and updates could therefore in principle be tackled, though corresponding procedures have not been implemented yet.[3]

5. APPROXIMATION IN REACHABILITY ANALYSIS

As has become apparent from the discussion in the previous sections, the hope for exact procedures for deciding reachability remains elusive when reasonably expressive subclasses of hybrid automata are to be dealt with. Whenever dynamics beyond clocks or initialization comes paired with non-trivial dimensionality of the problem at hand, the best we can hope for are approximation procedures. Several such procedures have been suggested over the years, with the logical properties of the individual procedures not being straightforward to delineate, as their underlying techniques as well as the model classes addressed are surprisingly diverse. In Table I we are giving a coarse overview of major types of approximation algorithms, which we describe in some more detail subsequently.

5.1. Trace generation based on shooting methods

For finding numerically approximate runs to the target state-set, multiple shooting methods can be used. Such methods generate numerous short segments of numerical simulation runs based on randomized variation of initial states of the simulation and then concatenate segments that are approximately matching in that end-points of one segment are metrically close to start-points of the successor segment [Zutshi et al. 2014]. The attraction of such shooting schemes is that they can deal with very general hybrid dynamics, as their analysis of system dynamics is based on numerical

[3]The situation is unclear w.r.t. hybrid systems featuring continuous dynamics satisfying constraints (2") and (3") or even (2') and (3'), as their BMC problems would involve solving quantified formulae over polynomial-exponential functions with multiple exponential variables. Decidability of the corresponding problem is open.

Scheme	Algorithm	HA classes	Logical properties	Engineering pragmatics
Multiple shooting	numerical simulation with randomized initial states; approximate matching of simulated segments	any	constitutes a numerically approximate search procedure for runs reaching target	constitutes an incomplete approximate falsification procedure for safety
Invariant generation	many: synthesis of Lyapunov functions, barrier certificates, Craig interpolation in k-induction	in principle any	constitutes an incomplete falsification procedure for reachability	constitutes an incomplete verification procedure for safety
Finite-state or predicate abstraction	compute overapproximating finite-state model based on an partitioning of the state-space; decide reachability in the abstraction; possibly refine partition on demand	in principle any	incomplete falsification procedure for reachability; in some cases quasi-decision procedure (s. below)	incomplete verification procedure for safety properties; in some cases quasi-decision procedure (s. below)
Overapproxim. in horizon-bounded reach-set computat.	select computational representation of certain subsets of the $\mathbb{R}^n$; compute overapproximations of (mostly bounded horizon) reach-sets	in principle any	constitutes an incomplete falsification procedure for horizon-bounded reachability	constitutes an incomplete verification procedure for horizon-bounded safety
ε-overapproximation	bound approximation error uniformly or as a function of step number or temporal horizon	primarily differential (in-) equations	constitutes an incomplete falsification procedure for (bounded) reachability	incomplete verification procedure for (bounded) safety; given a safe set with an incorporated safety margin, it is pragmatically complete
Gap domination	compute a bilinear mixed-integer system representing the exact reach-set; solve consistency with Pareto target with controlled error	MPTA with arbitrary number of stopwatch observers	terminating procedure guaranteed to yield exact verdict whenever either the Pareto target cannot be dominated or can be dominated by more than a $\varepsilon > 0$	decides domination of Pareto target "up to ε"
BMC using δ-decidable logic	encode bounded reachability properties as arithmetic constraints, determine either unsatisfiability of the constraint or satisfiability of a δ-relaxed version of the constraint	in principle any	sound but incomplete falsification procedure for bounded reachability	satisfying instances of the relaxation are taken to be close to actual paths reaching the target set; hence pragmatically a complete procedure for bounded *noise-robust* safety
Quasi-deciding robust reachability	iterate an ε-perturbed step relation; check whether bounded reach-set constitutes a pre-fixedpoint of the unperturbed dynamics	HA featuring decidable step relation, e.g. LHA	decides all *robust* cases where validity of the reachability property is not changed by infinitesimally small ε-perturbation	decides all *robust* cases where safety is not changed by infinitesimally small ε-perturbation; *fragile* cases may remain inconclusive
Underapproximation in reach-set computation	select computational representation of certain subsets of the $\mathbb{R}^n$; compute underapproximations of (mostly bounded horizon) reach-sets	primarily differential (in-) equations	sound but incomplete verification procedure for reachability	sound but incomplete falsification procedure for safety properties

Table I: Some approximation schemes for reachability

simulation being available for industrial scale, non-linear hybrid systems. The drawback is that due to numerical approximations in simulation as well as due to the only approximate match of trajectory segments, runs found by shooting do only numerically approximate actual runs of the hybrid system under investigation. The effect of approximate segment matching is alleviated by abstraction refinement reducing permitted gap width between concatenated segments, yet vanishes only in the limit.

5.2. Invariant generation

An obvious way of providing a reliable witness for unreachability is to construct a forward (or backward) invariant set containing the initial states (target, resp.) and to check that it is disjoint with the target (initial states, resp.). For an overview over the lively area of invariant generation, which we cannot cover here, we refer the reader to [Kapur 2017]. The approaches vary, as invariants can be constructed systematically employing reductions (mostly also involving relaxations) to numerical optimization or constraint solving, like in [Prajna et al. 2007; Sankaranarayanan et al. 2004; Liu et al. 2011; Oehlerking and Theel 2009], or can be established using guess-and-verify approaches, where simulations and machine-learning can be used to generate candidates which are subsequently verified (or falsified, which generates further sample points for the learning phase) by constraint solving [Ratschan 2017].

5.3. Finite-state abstractions

Another line of attack is to leverage existing techniques for finite-state model-checking by computing an existential finite-state abstraction of the hybrid dynamics that overapproximates location reachability. Such approaches rely on partitioning the continuous state-space of the hybrid automaton into finitely many cells, either by regular gridding or via the Boolean combination of a finite number of predicates. As this finite partition P together with the discrete modes V spans a finite state-space, the abstraction is obtained by computing a transition relation based on an (exact or necessary) condition for partition pairs containing points being connected by a step of the hybrid automaton: $(v_1, p_1) \rightarrow_{\text{abstract}} (v_2, p_2)$ in the abstraction, where $p_{1,2,} \in P$, if $\exists \vec{x}_1 \in p_1, \vec{x}_2 \in p_2 : (v_1, \vec{x}_1) \rightarrow_{\text{concrete}} (v_2, \vec{x}_2)$, where $\rightarrow_{\text{concrete}}$ denotes the union of all jumps and continuous evolutions. Abstraction thus involves a constraint-solving problem, as the above necessary condition for an abstract transition resides in an existential fragment of arithmetic. Reachability procedures exploiting such schemes are numerous, among them HSolver [Ratschan and She 2005], which implements a quasi-decision procedure for reachability (see Sect. 5.8) by applying counter-example guided partition refinement. Such refinement sequences need not terminate due to undecidability of the reachability problem, but rarely do so in practice.

5.4. Guaranteed over- and underapproximation in reach-set computation

A straightforward algorithmic decomposition of the reachability problem is to first compute the set of reachable states and then decide emptiness of its intersection with the target. The undecidability of reachability for sufficiently expressive hybrid automata implies that we have to accept that reach-set computation and intersection test in general cannot both be exact at the same time. A common scheme, known as safe approximation in reach-set computation, decides for a computational representation of subsets of the $\mathbb{R}^n$ that permits an exact intersection test at the price of having to approximate the (generally horizon-bounded) reach-set. In order for the approximation to be indicative for the reachability problem at hand, one computes a guaranteed overapproximation (or, more rarely, a guaranteed underapproximation) of the reachable state set such that a necessary (sufficient, resp.) condition for the reachability property at hand is provided. This implies that negative reachability verdicts obtained

on an overapproximation (positive verdicts obtained on an underapproximation, resp.) are reliable, while the opposite results are inconclusive w.r.t. the original reachability question. As the error in computing the approximation is in general not controlled, logical inferences possible from such failed analysis attempts, i.e. from positive reachability verdicts obtained on an overapproximation (negative reachability verdicts obtained on an underapproximation, resp.) are limited, unless further analysis is applied to the witnesses generated in those verification attempts.

Methods for computing overapproximations of differential equations or inclusions can be based on a variety of principles, among them abstracting the non-linear dynamics into piecewise constant, piecewise linear, or piecewise polynomial dynamics [Asarin et al. 2003; Althoff et al. 2008; Dang et al. 2010; Sankaranarayanan 2011; Althoff 2013], bisimulation functions [Fainekos et al. 2006] or discrepancy functions [Duggirala et al. 2013], Taylor series models [Nedialkov et al. 1999; Neher et al. 2007; Ramdani and Nedialkov 2011; Chen et al. 2012], generation of extremal trajectories by exploiting monotonicity properties of systems under consideration [Ramdani et al. 2009; Eggers et al. 2015], and appropriate bloating of the trajectories obtained from numerical simulation samples [Donzé and Maler 2007; Julius et al. 2007; Huang and Mitra 2012]. Applying these methods to hybrid automata additionally requires to be able to overapproximate intersections with the guard conditions whenever such intersections cannot exactly be represented within the set representation at hand. This happens, e.g., already when representing reachable sets by zonotopes while admitting systems of linear inequations as guards [Girard and Le Guernic 2008].

The issue of underapproximation of reach sets of differential equations has attracted comparatively less attention, but approaches exist for linear [Kurzhanski and Varaiya 2000; Girard et al. 2006] and non-linear systems of ordinary differential equations [Xue 2013; Goubault et al. 2014; Chen et al. 2014; Goubault and Putot 2017], as well as algorithms being able to compute both over- and underapproximations based on common principles [Goubault et al. 2018; Li et al. 2018].

5.5. ε-overapproximation in reach-set computation

Arguing with safety margins is standard practice in engineering whenever exact satisfaction of properties cannot be guaranteed. Adoption of such schemes has naturally been suggested for hybrid-automata analysis, where they come in the shape of a bound on the approximation error of a safe overapproximation. Most such schemes apply to differential equations rather than full hybrid automata and there provide a uniform bound $\delta > 0$ on the error in the derivative of the solution to the differential equation such that the Hausdorff distance between the set of states reachable in the overapproximation and in the precise model is bounded by an exponential function $f(h)$ of the length h of the time horizon. Stronger forms of ε-approximation, where the solution rather than its derivative fetaures a uniform error bound ε over an unbounded temporal horizon, can be obtained from variants of proof schemes for Lyapunov stability like bisimulation functions [Fainekos et al. 2006] or discrepancy functions [Duggirala et al. 2013].

ε-approximations with horizon-dependent error can be derived from numerical simulations and their respective error bounds if the initial values are points; generalizations to bounded initial sets can be obtained from safely bounding the dependency on initial values plus appropriately dense sampling [Donzé and Maler 2007; Julius et al. 2007; Huang and Mitra 2012]. [Ren and Kumar 2015] have extended the latter approach to cover hybrid behavior instead of just differential dynamics. Note that for differential dynamics, δ-approximations of the derivative, i.e. the right-hand side of the differential equation, do even permit finite state abstractions if the state set of the differential equation is bounded and the differential equation itself is Lipschitz [Puri

et al. 1996]. Like any abstraction based on state-space gridding, it however is prone to the curse of dimensionality in that its state set is exponential in the dimensionality of the differential system to be abstracted. If such a finite-state abstraction exists, it however permits a very direct overapproximate computation also of the unbounded reach-set, as the full, step-unbounded reach-set of the finite-state abstraction can easily be computed. It should however be noted that the resulting overapproximation of the full reach-set in general has no quantifiable error bound.

The primary engineering use-case of such ε-overapproximations with time-dependent approximation error therefore is bounded analysis of the states reachable within a time horizon h. In these cases, the error bound $f(h)$ permits a metric classification of the amount of incompleteness of the overapproximate analysis: not only is the overapproximation known to always yield a positive reachability verdict whenever the concrete, precise system reaches the target within the bounded time horizon h; vice versa it is also guaranteed that the overapproximation will yield a negative verdict whenever the horizon-bounded reach-set of the precise systems keeps a distance of at least $f(h)$ from the target. In a sense, such a procedure "decides" horizon-bounded reachability up to an accuracy $f(h)$: it is guaranteed to yield accurate verdicts unless the actual horizon-bounded reach-set comes $f(h)$-close to the target without reaching it. Unbounded verdicts can in some cases be derived when the approximation of the horizon-bounded dynamics turns out to be contractive.

5.6. Gap domination

A particular variant of the error-bounded approximation scheme applies to Pareto domination problems, where one may naturally add an indifference region as follows: if a certain Pareto target $\phi = \bigwedge_{i=1}^{n} x_i \sim_i n_i$ cannot be reached over unbounded time horizon then this should be reported, likewise when a strengthening of the Pareto target by a given ε into a Pareto target $\phi' = \bigwedge_{i=1}^{n} x_i \sim_i m_i$ can be reached, where $m_i = n_i - \varepsilon$ if $\sim_i = \leq$ and $m_i = n_i + \varepsilon$ else, then this ought also be reported. Any answer is permitted when ϕ can be reached, but the strengthening ϕ' cannot. This yields a *gap domination problem* where definite answers are only required if the Pareto-optimal solution does not fall into the above ε gap between ϕ and its strengthening ϕ'.

Such gap domination problems take the idea of "deciding" up to a tolerance exposed in the previous section from the bounded-horizon case to the unbounded. As approximation errors would normally accumulate over the duration of the run, settings where a finite approximation error can rigorously be guaranteed in the unbounded are scarce. A notable case here are multi-priced timed automata with stopwatch price observers (stopwatch-MPTA): despite their exact Pareto reachability problems being undecidable, their gap domination problems can be solved effectively based on approximately solving a bilinear mixed-integer system exactly representing the unbounded reach-set [Fränzle et al. 2018]. [Platzer and Clarke 2007] have however shown that such schemes do in general not extend to hybrid automata with a natural notion of an ε-gap to reach targets, where any answer would be allowed when coming ε-close to the reach-set.

5.7. BMC using δ-decidable logic

A variant of the aforementioned scheme has been facilitated by the development of so-called δ-decision procedures [Gao et al. 2012] and the implementation of practical variants thereof by a tight integration of SAT solving, interval constraint propagation, and safe interval-based enclosure of differential equations [Eggers et al. 2008; Gao et al. 2013]. In δ-decidability, the idea of solving a gap problem has been lifted from the reachability problem (Sect. 5.6: do we get into the target or can we guarantee to stay off the target by a distance of at least ε?) to its representation as a constraint formula. Taking a formula ϕ encoding a bounded reachability problem in an undecid-

able fragment of arithmetic comprising arithmetic expressions as well as differential equations composed from linear, polynomial, and transcendental functions, the question here is, given a user-specified bound $\delta > 0$: is ϕ unsatisfiable or is a δ-relaxation ϕ' satisfiable? The δ-relaxation ϕ' is obtained by first rewriting all equations from ϕ to conjunctions of inequations, then normalizing all inequations to the form $e \leq 0$, and subsequently relaxing them to $e \leq \delta$?

Note that for conjunctive path conditions ϕ, satisfiability of ϕ' is closely related to the concept of ε-approximate reach-set computation, as a path formula conjoins equations expressing initialization, consistency with continuous dynamics, and consecution (cf. Sect. 2), each of which is relaxed by δ in ϕ'. This close correspondence unfortunately breaks down when disjunctions are allowed to express the branching behavior of a transition system, potentially giving rise to spurious paths.

5.8. Robust reachability

As mentioned in Sect. 3, pronounced criticism has been expressed against undecidability results encoding integers within the finite-dimensional continuous state-space of a hybrid automaton. This criticism is based on the observation that such encodings have to rely on unbounded range of continuous variables or have to feature at least one accumulation point in the $\mathbb{R}^n$, both of which are not representative of practical applications combining compact operational ranges with the ubiquity of noise. The corresponding undecidability results are consequently suspected to be artifacts of an overly idealized model rather than inherent to the problem domain. In the hope of avoiding undecidability, various relaxations of the hybrid automaton model have been suggested that set out to robustify reachability verdicts against infinitesimal noise. The two basic set-ups here are to either eliminate topologically isolated runs of the hybrid system and likewise add missing runs featuring a neighborhood of runs [Gupta et al. 1997] or to perturb the step relation (where a step is a continuous flow or a transition and perturbation may apply to just one of the two categories) to an ε-neighborhood and to consider all those states as reachable which are reachable within *all* such $\varepsilon > 0$-perturbations [Puri 1998; Fränzle 1999]. An observation made in [Fränzle 1999] was that a pre-fixedpoint check whether the unperturbed step relation is contracting on a step-bounded reach-set of the perturbed one is bound to eventually —in the sense of after finitely many steps of unraveling the transition relation— succeed if (1.) the step relations (both unperturbed and perturbed) can be expressed in a decidable logic, (2.) the state invariants or the complement of the target form a bounded subset of the $\mathbb{R}^n$, and (3.) the perturbed dynamics does not reach the target. The consequence is that a procedure can be built that is guaranteed to terminate with one of the two following verdicts: the perturbed dynamics is shown to reach the target or the unperturbed one is proven to avoid the target. If one iterates this procedure for successively smaller $\varepsilon \to 0$ until a witness for unreachability of the target in the unperturbed is obtained and runs the resulting procedure in parallel with the straightforward semi-decision procedure for reachability of the target that is based on bounded model-checking of a decidable step relation, then this procedure terminates on all but the non-robust cases as follows: If the unperturbed dynamics reaches the target set then BMC will confirm this. If the unperturbed dynamics avoids the target and if there exists an (arbitrarily small) positive disturbance $\varepsilon > 0$ such that the perturbed dynamics also avoids the target then the pre-fixedpoint test will confirm unreachability in the unperturbed. The procedure may fail to terminate only if the unperturbed dynamics itself cannot reach the target, yet any infinitesimally small disturbance does so.

Ratschan [Damm et al. 2005; Ratschan 2014] has coined the term *quasi-decidability* for this property that all reachability properties that are robust against infinitesimal perturbation can be decided and that only the fragile cases may remain undecided.

Thereby a system is called *robust* for a property iff the truth value of the property agrees between the unperturbed system and a certain (maybe small) range of perturbations; conversely it is called *fragile* if the truth value disagrees between the unperturbed system and any small perturbatio thereof.

[Asarin and Bouajjani 2001] have investigated such perturbations widening the step relation on several models of computation and concluded that robust avoidance of the target is semi-decidable for LHA (and a number of other models), in contrast to the unperturbed case where reachability of the target is semi-decidable. Quasi-decidability exploits the combination of these two properties in that it conceptually runs both semi-decision procedures in parallel and thus is able to ensure termination in all robust cases. An immediate consequence of the general undecidability of reachability in linear hybrid automata then is that it is i.g. undecidable whether an LHA (or a related computational structure) is robust w.r.t. a reachability property. From an engineering standpoint, this seems to constitute a minor impediment only, as fragile satisfaction of a design requirement may be considered bad engineering practice.

6. CONCLUSIONS

A good quarter of a century after their inception as a formal model seamlessly integrating continuous dynamics described by ordinary differential equations and discrete switching behavior, hybrid automata [Maler et al. 1992; Alur et al. 1993] and their reachability analysis still are a lively area of research. Major breakthroughs concerning scalability of automatic analysis techniques have been achieved both with approaches building on reach-set computation, like [Henzinger et al. 1997; Althaus et al. 2017; Frehse et al. 2011; Althoff et al. 2018; Chen et al. 2015], and with such exploiting reductions to arithmetic constraint solving, like [Bemporad and Morari 1999; Audemard et al. 2005; Eggers et al. 2008; Gao et al. 2013]. It is interesting to see that many of the more recent variants adopt approximations not only because they are necessary due to undecidability results pertaining to exact reachability, but argue that the use of safe approximations is inherent to the engineering pragmatics in the domain, and thus warranted. In a sense, these approaches do thus consider the model of hybrid automata as a mathematically elegant, yet overidealized abstraction. Despite the plethora of deep and elegant results as well as of useful algorithms obtained on the existing model, the quest for alternative models remains pronounced. We anticipate that it will spark further interdisciplinary research between the areas of computer science, control theory, and metrology, perhaps more deeply embedding the perspectives of state estimation and control under uncertainty into the computational model.

ACKNOWLEDGMENTS

We dedicate this note to our dear colleague Oded Maler (1957–2018), whose untimely death at an age of 61 came to a shock to all of us. Oded has been instrumental to the development of this very field of research in so many ways: as a researcher preparing a thorough ground for the others by contributing rigorous and elegant formalizations of the problem domain and numerous fundamental results, as an academic teacher and colleague guiding and supporting others in their scientific endeavors, as a dedicated scientist devoted to scientific truth and therefore always discussing with verve and reviewing with exceptional scrutiny, and last not least as an organizer and manager crucial to the launch and stabilization of key conferences in the field.

Work on this note has been supported through grants by Deutsche Forschungsgemeinschaft within the Research Training Group DFG GRK 1765 SCARE and by NSFC under grant No. 61625206 and 61732001, by "973 Program" under grant No. 2014CB340701, and by the CAS/SAFEA International Partnership Program for Creative Research Teams.

REFERENCES

Ernst Althaus, Björn Beber, Werner Damm, Stefan Disch, Willem Hagemann, Astrid Rakow, Christoph Scholl, Uwe Waldmann, and Boris Wirtz. 2017. Verification of linear hybrid systems with large discrete state spaces using counterexample-guided abstraction refinement. *Sci. Comput. Program.* 148 (2017), 123–160.

Matthias Althoff. 2013. Reachability analysis of nonlinear systems using conservative polynomialization and non-convex sets. In *Proceedings of the 16th International Conference on Hybrid systems: Computation and Control, HSCC 2013, April 8-11, 2013, Philadelphia, PA, USA*, Calin Belta and Franjo Ivancic (Eds.). ACM, 173–182.

Matthias Althoff, Dmitry Grebenyuk, and Niklas Kochdumper. 2018. Implementation of Taylor models in CORA 2018. In *ARCH18. 5th International Workshop on Applied Verification of Continuous and Hybrid Systems, ARCH@ADHS 2018, Oxford, UK, July 13, 2018 (EPiC Series in Computing)*, Goran Frehse, Matthias Althoff, Sergiy Bogomolov, and Taylor T. Johnson (Eds.), Vol. 54. EasyChair, 145–173.

Matthias Althoff, Olaf Stursberg, and Martin Buss. 2008. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In *Proceedings of the 47th IEEE Conference on Decision and Control, CDC 2008, December 9-11, 2008, Cancún, Mexico*. IEEE, 4042–4048.

Rajeev Alur, Costas Courcoubetis, Thomas A. Henzinger, and Pei-Hsin Ho. 1993. Hybrid automata: an algorithmic approach to the specification and verification of hybrid systems. In *Hybrid Systems*, Robert L. Grossman, Anil Nerode, Anders P. Ravn, and Hans Rischel (Eds.). Lecture Notes in Computer Science, Vol. 736. Springer Berlin Heidelberg, 209–229.

Rajeev Alur and David L. Dill. 1990. Automata for modeling real-time systems. In *Automata, Languages and Programming, 17th International Colloquium, ICALP90, Warwick University, England, UK, July 16-20, 1990, Proceedings (Lecture Notes in Computer Science)*, Mike Paterson (Ed.), Vol. 443. Springer, 322–335.

Hirokazu Anai and Volker Weispfenning. 2001. Reach set computations using real quantifier elimination. In *Hybrid Systems: Computation and Control, 4th International Workshop, HSCC 2001, Rome, Italy, March 28-30, 2001, Proceedings*. Springer Berlin Heidelberg, Berlin, Heidelberg, 63–76.

Eugene Asarin and Ahmed Bouajjani. 2001. Perturbed turing machines and hybrid systems. In *16th Annual IEEE Symposium on Logic in Computer Science, Boston, Massachusetts, USA, June 16-19, 2001, Proceedings*. IEEE Computer Society, 269–278.

Eugene Asarin, Thao Dang, and Antoine Girard. 2003. Reachability analysis of nonlinear systems using conservative approximation. In *Hybrid Systems: Computation and Control, 6th International Workshop, HSCC 2003 Prague, Czech Republic, April 3-5, 2003, Proceedings (Lecture Notes in Computer Science)*, Oded Maler and Amir Pnueli (Eds.), Vol. 2623. Springer, 20–35.

Eugene Asarin, Oded Maler, and Amir Pnueli. 1995. Reachability analysis of dynamical systems having piecewise-constant derivatives. *Theoretical Computer Science* 138, 1 (1995), 35 – 65. Hybrid Systems.

Gilles Audemard, Marco Bozzano, Alessandro Cimatti, and Roberto Sebastiani. 2005. Verifying industrial hybrid systems with MathSAT. *Electronic Notes in Theoretical Computer Science* 119, 2 (2005), 17 – 32. Proceedings of the 2nd International Workshop on Bounded Model Checking (BMC 2004).

Thomas Becker, Volker Weispfenning, and Heinz Kredel. 1993. *Gröbner Bases: A Computational Approach to Commutative Algebra*. Springer-Verlag.

Gerd Behrmann, Alexandre David, Kim G. Larsen, Paul Pettersson, and Wang Yi. 2011. Developing UPPAAL over 15 years. *Softw., Pract. Exper.* 41, 2 (2011), 133–142.

Alberto Bemporad and Manfred Morari. 1999. Verification of hybrid systems via mathematical programming *(Lecture Notes in Computer Science)*, Frits W. Vaandrager and Jan H. van Schuppen (Eds.), Vol. 1569. Springer, 31–45.

Armin Biere, Alessandro Cimatti, Edmund M. Clarke, and Yunshan Zhu. 1999. Symbolic model checking without BDDs. In *Tools and Algorithms for Construction and Analysis of Systems, 5th International Conference, TACAS '99, Held as Part of the European Joint Conferences on the Theory and Practice of Software, ETAPS'99, Amsterdam, The Netherlands, March 22-28, 1999, Proceedings (Lecture Notes in Computer Science)*, Rance Cleaveland (Ed.), Vol. 1579. Springer, 193–207.

Xin Chen, Erika Ábrahám, and Sriram Sankaranarayanan. 2012. Taylor model flowpipe construction for non-linear hybrid systems. In *Proceedings of the 33rd IEEE Real-Time Systems Symposium, RTSS 2012, San Juan, PR, USA, December 4-7, 2012*. IEEE Computer Society, 183–192.

Xin Chen, Sriram Sankaranarayanan, and Erika Ábrahám. 2014. Under-approximate flowpipes for non-linear continuous systems. In *Formal Methods in Computer-Aided Design, FMCAD 2014, Lausanne, Switzerland, October 21-24, 2014*. IEEE, 59–66.

Xin Chen, Sriram Sankaranarayanan, and Erika Ábrahám. 2015. Flow* 1.2: more effective to play with hybrid systems. In *1st and 2nd International Workshop on Applied Verification for Continuous and Hybrid Systems, ARCH@CPSWeek 2014, Berlin, Germany, April 14, 2014 / ARCH@CPSWeek 2015, Seattle, WA, USA, April 13, 2015. (EPiC Series in Computing)*, Goran Frehse and Matthias Althoff (Eds.), Vol. 34. EasyChair, 152–159.

Hubert Comon and Yan Jurski. 1999. Timed automata and the theory of real numbers. In *CONCUR '99: Concurrency Theory, 10th International Conference, Eindhoven, The Netherlands, August 24-27, 1999, Proceedings (Lecture Notes in Computer Science)*, Jos C. M. Baeten and Sjouke Mauw (Eds.), Vol. 1664. Springer, 242–257.

Werner Damm, Guilherme Pinto, and Stefan Ratschan. 2005. Guaranteed termination in the verification of LTL properties of non-linear robust discrete time hybrid systems. In *Automated Technology for Verification and Analysis, Third International Symposium, ATVA 2005, Taipei, Taiwan, October 4-7, 2005, Proceedings (Lecture Notes in Computer Science)*, Doron A. Peled and Yih-Kuen Tsay (Eds.), Vol. 3707. Springer, 99–113.

Thao Dang, Oded Maler, and Romain Testylier. 2010. Accurate hybridization of nonlinear systems. In *Proceedings of the 13th ACM International Conference on Hybrid Systems: Computation and Control, HSCC 2010, Stockholm, Sweden, April 12-15, 2010*, Karl Henrik Johansson and Wang Yi (Eds.). ACM, 11–20.

Alexandre Donzé and Oded Maler. 2007. Systematic simulation using sensitivity analysis. In *Hybrid Systems: Computation and Control, 10th International Workshop, HSCC 2007, Pisa, Italy, April 3-5, 2007, Proceedings (Lecture Notes in Computer Science)*, Alberto Bemporad, Antonio Bicchi, and Giorgio C. Buttazzo (Eds.), Vol. 4416. Springer, 174–189.

Parasara S. Duggirala, Sayan Mitra, and Mahesh Viswanathan. 2013. Verification of annotated models from executions. In *Proceedings of the International Conference on Embedded Software, EMSOFT 2013, Montreal, QC, Canada, September 29 - Oct. 4, 2013*. IEEE, 26:1–26:10.

Andreas Eggers, Martin Fränzle, and Christian Herde. 2008. SAT modulo ODE: A direct SAT approach to hybrid systems. In *Automated Technology for Verification and Analysis, 6th International Symposium, ATVA 2008, Seoul, Korea, October 20-23, 2008. Proceedings (Lecture Notes in Computer Science)*, Sung Deok Cha, Jin-Young Choi, Moonzoo Kim, Insup Lee, and Mahesh Viswanathan (Eds.), Vol. 5311. Springer, 171–185.

Andreas Eggers, Nacim Ramdani, Nedialko S. Nedialkov, and Martin Fränzle. 2015. Improving the SAT modulo ODE approach to hybrid systems analysis by combining different enclosure methods. *Software and System Modeling* 14, 1 (2015), 121–148.

Georgios E. Fainekos, Antoine Girard, and George J. Pappas. 2006. Temporal logic verification using simulation. In *Formal Modeling and Analysis of Timed Systems, 4th International Conference, FORMATS 2006, Paris, France, September 25-27, 2006, Proceedings (Lecture Notes in Computer Science)*, Eugene Asarin and Patricia Bouyer (Eds.), Vol. 4202. Springer, 171–186.

Martin Fränzle. 1999. Analysis of hybrid systems: an ounce of realism can save an infinity of states. In *Computer Science Logic: 13th International Workshop, CSL'99 8th Annual Conference of the EACSL Madrid, Spain, September 20–25, 1999 Proceedings*, Jörg Flum and Mario Rodriguez-Artalejo (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 126–139.

Martin Fränzle, Mahsa Shirmohammadi, Mani Swaminathan, and James Worrell. 2018. Costs and rewards in priced timed automata. In *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic (LIPIcs)*, Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella (Eds.), Vol. 107. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 125:1–125:14.

Goran Frehse, Colas Le Guernic, Alexandre Donzé, Scott Cotton, Rajarshi Ray, Olivier Lebeltel, Rodolfo Ripado, Antoine Girard, Thao Dang, and Oded Maler. 2011. SpaceEx: Scalable verification of hybrid systems. In *Computer Aided Verification - 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings (Lecture Notes in Computer Science)*, Ganesh Gopalakrishnan and Shaz Qadeer (Eds.), Vol. 6806. Springer, 379–395.

Ting Gan, Mingshuai Chen, Liyun Dai, Bican Xia, and Naijun Zhan. 2015. Decidability of the reachability for a family of linear vector fields. In *Automated Technology for Verification and Analysis - 13th International Symposium, ATVA 2015, Shanghai, China, October 12-15, 2015, Proceedings*. Springer International Publishing, Cham, 482–499.

Ting Gan, Mingshuai Chen, Yangjia Li, Bican Xia, and Naijun Zhan. 2016. Computing reachable sets of linear vector fields revisited. In *2016 European Control Conference, ECC 2016, Aalborg, Denmark, June 29 - July 1, 2016*. IEEE, 419–426.

Ting Gan, Mingshuai Chen, Yangjia Li, Bican Xia, and Naijun Zhan. 2018. Reachability analysis for solvable dynamical systems. *IEEE Trans. Automat. Contr.* 63, 7 (2018), 2003–2018.

Sicun Gao, Jeremy Avigad, and Edmund M. Clarke. 2012. δ-complete decision procedures for satisfiability over the reals. In *Automated Reasoning - 6th International Joint Conference, IJCAR 2012, Manchester, UK, June 26-29, 2012. Proceedings (Lecture Notes in Computer Science)*, Bernhard Gramlich, Dale Miller, and Uli Sattler (Eds.), Vol. 7364. Springer, 286–300.

Sicun Gao, Soonho Kong, and Edmund M. Clarke. 2013. Satisfiability modulo ODEs. In *Formal Methods in Computer-Aided Design, FMCAD 2013, Portland, OR, USA, October 20-23, 2013*. IEEE, 105–112.

Antoine Girard, Colas Le Guernic, and Oded Maler. 2006. Efficient computation of reachable sets of linear time-invariant systems with inputs. In *Hybrid Systems: Computation and Control, 9th International Workshop, HSCC 2006, Santa Barbara, CA, USA, March 29-31, 2006, Proceedings (Lecture Notes in Computer Science)*, João P. Hespanha and Ashish Tiwari (Eds.), Vol. 3927. Springer, 257–271.

Antoine Girard and Colas Le Guernic. 2008. Zonotope/hyperplane intersection for hybrid systems reachability analysis. In *Hybrid Systems: Computation and Control*, Magnus Egerstedt and Bud Mishra (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 215–228.

Eric Goubault, Olivier Mullier, Sylvie Putot, and Michel Kieffer. 2014. Inner approximated reachability analysis. In *17th International Conference on Hybrid Systems: Computation and Control (part of CPS Week), HSCC'14, Berlin, Germany, April 15-17, 2014*, Martin Fränzle and John Lygeros (Eds.). ACM, 163–172.

Eric Goubault and Sylvie Putot. 2017. Forward inner-approximated reachability of non-linear continuous systems. In *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control, HSCC 2017, Pittsburgh, PA, USA, April 18-20, 2017*, Goran Frehse and Sayan Mitra (Eds.). ACM, 1–10.

Eric Goubault, Sylvie Putot, and Lorenz Sahlmann. 2018. Inner and outer approximating flowpipes for delay differential equations. In *Computer Aided Verification - 30th International Conference, CAV 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford, UK, July 14-17, 2018, Proceedings, Part II (Lecture Notes in Computer Science)*, Hana Chockler and Georg Weissenbacher (Eds.), Vol. 10982. Springer, 523–541.

Jan F. Groote, J. W. C. Koorn, and Sebastiaan F. M. van Vlijmen. 1995. The safety guaranteeing system at station Hoorn-Kersenboogerd. In *Compass '95: 10th Annual Conference on Computer Assurance*. National Institute of Standards and Technology, Gaithersburg, Maryland, 57–68.

Vineet Gupta, Thomas A. Henzinger, and Radha Jagadeesan. 1997. Robust timed automata *(Lecture Notes in Computer Science)*, Oded Maler (Ed.), Vol. 1201. Springer, 331–345.

Godfrey H. Hardy, Edward M. Wright, and E.W. Wright. 1979. *An Introduction to the Theory of Numbers*. Clarendon Press.

Thomas A. Henzinger, Pei-Hsin Ho, and Howard Wong-Toi. 1997. HYTECH: A model checker for hybrid systems. *STTT* 1, 1-2 (1997), 110–122.

Thomas A. Henzinger, Peter W. Kopke, Anuj Puri, and Pravin Varaiya. 1998. What's decidable about hybrid automata? *J. Comput. System Sci.* 57, 1 (1998), 94 – 124.

Thomas A. Henzinger, Xavier Nicollin, Joseph Sifakis, and Sergio Yovine. 1992. Symbolic model checking for real-time systems. In *Proceedings of the Seventh Annual Symposium on Logic in Computer Science (LICS '92), Santa Cruz, California, USA, June 22-25, 1992*. IEEE Computer Society, 394–406.

Zhenqi Huang and Sayan Mitra. 2012. Computing bounded reach sets from sampled simulation traces. In *Hybrid Systems: Computation and Control (part of CPS Week 2012), HSCC'12, Beijing, China, April 17-19, 2012*, Thao Dang and Ian M. Mitchell (Eds.). ACM, 291–294.

A. Agung Julius, Georgios E. Fainekos, Madhukar Anand, Insup Lee, and George J. Pappas. 2007. Robust test generation and coverage for hybrid systems. In *Hybrid Systems: Computation and Control, 10th International Workshop, HSCC 2007, Pisa, Italy, April 3-5, 2007, Proceedings (Lecture Notes in Computer Science)*, Alberto Bemporad, Antonio Bicchi, and Giorgio C. Buttazzo (Eds.), Vol. 4416. Springer, 329–342.

Deepak Kapur. 2017. Nonlinear polynomials, interpolants and invariant generation for system analysis. In *Proceedings of the 2nd International Workshop on Satisfiability Checking and Symbolic Computation co-located with the 42nd International Symposium on Symbolic and Algebraic Computation (ISSAC 2017), Kaiserslautern, Germany, July 29, 2017. (CEUR Workshop Proceedings)*, Matthew England and Vijay Ganesh (Eds.), Vol. 1974. CEUR-WS.org.

Alexander B. Kurzhanski and Pravin Varaiya. 2000. Ellipsoidal techniques for reachability analysis: internal approximation. *Systems & Control Letters* 41, 3 (2000), 201 – 211.

Gerardo Lafferriere, George J. Pappas, and Sergio Yovine. 1999. A new class of decidable hybrid systems. In *Hybrid Systems: Computation and Control*, Frits W. Vaandrager and Jan H. van Schuppen (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 137–151.

Gerardo Lafferriere, George J. Pappas, and Sergio Yovine. 2001. Symbolic reachability computation for families of linear vector fields. *J. Symb. Comput.* 32, 3 (2001), 231–253.

Kim G. Larsen, Paul Pettersson, and Wang Yi. 1995. Compositional and symbolic model-checking of real-time systems. In *Proceedings 16th IEEE Real-Time Systems Symposium(RTSS)*. IEEE Computer Society, 76–87.

Kim G. Larsen and Jacob I. Rasmussen. 2008. Optimal reachability for multi-priced timed automata. *Theoretical Computer Science* 390, 2 (2008), 197 – 213. Foundations of Software Science and Computational Structures.

Meilun Li, Peter Nazier Mosaad, Martin Fränzle, Zhikun She, and Bai Xue. 2018. Safe over- and under-approximation of reachable sets for autonomous dynamical systems. In *Formal Modeling and Analysis of Timed Systems - 16th International Conference, FORMATS 2018, Beijing, China, September 4-6, 2018, Proceedings (Lecture Notes in Computer Science)*, David N. Jansen and Pavithra Prabhakar (Eds.), Vol. 11022. Springer, 252–270.

Jiang Liu, Naijun Zhan, and Hengjun Zhao. 2011. Computing semi-algebraic invariants for polynomial dynamical systems. In *Proceedings of the 11th International Conference on Embedded Software, EMSOFT 2011, part of the Seventh Embedded Systems Week, ESWeek 2011, Taipei, Taiwan, October 9-14, 2011*, Samarjit Chakraborty, Ahmed Jerraya, Sanjoy K. Baruah, and Sebastian Fischmeister (Eds.). ACM, 97–106.

Oded Maler, Zohar Manna, and Amir Pnueli. 1992. From timed to hybrid systems. In *Real-Time: Theory in Practice, REX Workshop, Mook, The Netherlands, June 3-7, 1991, Proceedings (Lecture Notes in Computer Science)*, J. W. de Bakker, Cornelis Huizing, Willem P. de Roever, and Grzegorz Rozenberg (Eds.), Vol. 600. Springer, 447–484.

Dinesh Manocha and John F. Canny. 1992. Algorithm for implicitizing rational parametric surfaces. *Comput. Aided Geom. Des.* 9, 1 (May 1992), 25–50.

Nedialko S. Nedialkov, Kenneth R. Jackson, and George F. Corliss. 1999. Validated solutions of initial value problems for ordinary differential equations. *Appl. Math. Comput.* 105, 1 (1999), 21–68.

Markus Neher, Kenneth R. Jackson, and Nedialko S. Nedialkov. 2007. On Taylor model based integration of ODEs. *SIAM J. Numerical Analysis* 45, 1 (2007), 236–262.

Jens Oehlerking and Oliver E. Theel. 2009. A decompositional proof scheme for automated convergence proofs of stochastic hybrid systems. In *Automated Technology for Verification and Analysis, 7th International Symposium, ATVA 2009, Macao, China, October 14-16, 2009. Proceedings (Lecture Notes in Computer Science)*, Zhiming Liu and Anders P. Ravn (Eds.), Vol. 5799. Springer, 151–165.

André Platzer and Edmund M. Clarke. 2007. The image computation problem in hybrid systems model checking. In *Hybrid Systems: Computation and Control, 10th International Workshop, HSCC 2007, Pisa, Italy, April 3-5, 2007, Proceedings (Lecture Notes in Computer Science)*, Alberto Bemporad, Antonio Bicchi, and Giorgio C. Buttazzo (Eds.), Vol. 4416. Springer, 473–486.

Pavithra Prabhakar, Vladimeros Vladimerou, Mahesh Viswanathan, and Geir Dullerud. 2015. A decidable class of planar linear hybrid systems. *Theoretical Computer Science* 574, Supplement C (2015), 1 – 17.

Stephen Prajna, Ali Jadbabaie, and George J. Pappas. 2007. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Trans. Automat. Contr.* 52, 8 (2007), 1415–1428.

Anuj Puri. 1998. Dynamical properties of timed automata *(Lecture Notes in Computer Science)*, Anders P. Ravn and Hans Rischel (Eds.), Vol. 1486. Springer, 210–227.

Anuj Puri, Vivek Borkar, and Pravin Varaiya. 1996. ϵ-Approximation of differential inclusions. In *Hybrid Systems III*, Rajeev Alur, Thomas A. Henzinger, and Eduardo D. Sontag (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 362–376.

Karin Quaas, Mahsa Shirmohammadi, and James Worrell. 2017. Revisiting reachability in timed automata. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*. IEEE Computer Society, 1–12.

Nacim Ramdani, Nacim Meslem, and Yves Candau. 2009. A hybrid bounding method for computing an over-approximation for the reachable set of uncertain nonlinear systems. *IEEE Trans. Automat. Contr.* 54, 10 (2009), 2352–2364.

Nacim Ramdani and Nedialko S. Nedialkov. 2011. Computing reachable sets for uncertain nonlinear hybrid systems using interval constraint-propagation techniques. *Nonlinear Analysis: Hybrid Systems* 5, 2 (2011), 149 – 162. Special Issue related to IFAC Conference on Analysis and Design of Hybrid Systems (ADHS09).

Stefan Ratschan. 2014. Safety verification of non-linear hybrid systems is quasi-decidable. *Formal Methods in System Design* 44, 1 (2014), 71–90. DOI:http://dx.doi.org/10.1007/s10703-013-0196-2

Stefan Ratschan. 2017. Simulation based computation of certificates for safety of dynamical systems. In *Formal Modeling and Analysis of Timed Systems - 15th International Conference, FORMATS 2017,*

Berlin, Germany, September 5-7, 2017, Proceedings (Lecture Notes in Computer Science), Alessandro Abate and Gilles Geeraerts (Eds.), Vol. 10419. Springer, 303–317.

Stefan Ratschan and Zhikun She. 2005. Safety verification of hybrid systems by constraint propagation based abstraction refinement. In *Hybrid Systems: Computation and Control, 8th International Workshop, HSCC 2005, Zurich, Switzerland, March 9-11, 2005, Proceedings (Lecture Notes in Computer Science)*, Manfred Morari and Lothar Thiele (Eds.), Vol. 3414. Springer, 573–589.

Hao Ren and Ratnesh Kumar. 2015. Step simulation/overapproximation-based verification of nonlinear deterministic hybrid system with inputs. In *5th IFAC Conference on Analysis and Design of Hybrid Systems, ADHS 2015, Atlanta, GA, USA, October 14-16, 2015 (IFAC-PapersOnLine)*, Magnus Egerstedt and Yorai Wardi (Eds.), Vol. 48. Elsevier, 21–26.

Enric Rodríguez-Carbonell and Deepak Kapur. 2007. Generating all polynomial invariants in simple loops. *J. Symb. Comput.* 42, 4 (2007), 443–476.

Sriram Sankaranarayanan. 2011. Automatic abstraction of non-linear systems using change of bases transformations. In *Proceedings of the 14th ACM International Conference on Hybrid Systems: Computation and Control, HSCC 2011, Chicago, IL, USA, April 12-14, 2011*, Marco Caccamo, Emilio Frazzoli, and Radu Grosu (Eds.). ACM, 143–152.

Sriram Sankaranarayanan, Henny Sipma, and Zohar Manna. 2004. Constructing invariants for hybrid systems. In *Hybrid Systems: Computation and Control, 7th International Workshop, HSCC 2004, Philadelphia, PA, USA, March 25-27, 2004, Proceedings (Lecture Notes in Computer Science)*, Rajeev Alur and George J. Pappas (Eds.), Vol. 2993. Springer, 539–554.

Warren D. Smith. 2006. Church's thesis meets the N-body problem. *Appl. Math. Comput.* 178, 1 (2006), 154–183.

Alfred Tarski. 1951. *A Decision Method for Elementary Algebra and Geometry*. University of California Press, Berkeley.

Thomas Wilke. 1994. *Automaten und Logiken zur Beschreibung zeitabhängiger Systeme*. Ph.D. Dissertation. University of Kiel, Germany.

Ming Xu, Jiaqi Zhu, and Zhi-bin Li. 2013. Some decidable results on reachability of solvable systems. *Int. J. General Systems* 42, 4 (2013), 405–425.

Bai Xue. 2013. *Computing Rigor Quadratic Lyapunov Functions and Under-approximate Reachable Sets for Ordinary Differential Equations*. Ph.D. Dissertation. Beihang University.

Aditya Zutshi, Jyotirmoy V. Deshmukh, Sriram Sankaranarayanan, and James Kapinski. 2014. Multiple shooting, CEGAR-based falsification for hybrid systems. In *2014 International Conference on Embedded Software, EMSOFT 2014, New Delhi, India, October 12-17, 2014*, Tulika Mitra and Jan Reineke (Eds.). ACM, 5:1–5:10.