Discretizing Affine Hybrid Automata with Uncertainty*,**

Thao Dang[1] and Thomas Martin Gawlitza[1]

VERIMAG
{Thao.Dang,Thomas.Gawlitza}@imag.fr

Abstract. Over-approximating the set of all reachable states of a given system is an important task for the verification of *safety properties*. Such an unbounded time verification is in particular challenging for *hybrid systems*. We recently developed an algorithm that over-approximates the set of all reachable states of a given *affine hybrid automata* by performing linear template-based abstract interpretation [4]. In this article we extend the previous results by adding *uncertainty* to the model of affine hybrid automata. Uncertainty can be used for abstracting the behavior of non-linear hybrid systems. We adapt our techniques to this model and show that, w.r.t. given linear templates, the abstract reachability problem is still in coNP by reducing abstract reachability for affine hybrid automata with uncertainty to abstract reachability for affine programs (affine hybrid automata where only discrete transitions are allowed). We thus provide a new connection between a continuous time model and a purely discrete model.

1 Introduction

Hybrid systems have been widely recognized as a mathematical model appropriate for describing and reasoning about the interactions of software, modeled by discrete systems such as automata, with the physical world, described by continuous systems such as differential equations. Cyber-physical systems are recent applications involving such interactions. In addition, many applications of cyber-physical systems must be reliable and safe, not only for economic reasons but also for human safety. Automated verification technologies are thus indispensable for the efficiency of their design. Uncertainty is an important feature of cyber-physical systems. Indeed, accurate models of some of their components may not be available or reliability of interoperation of their heterogeneous subsystems may not be guantanteed. Moreover, modelling complex cyber-physical systems with reasonable accuracy is a very challenging task; therefore uncertainty in their models is often unavoidable. While uncertainty can result from imprecision in modelling, it can also result from the abstraction and approximation procedures frequenty used in systems design. Indeed, the dynamics of real-life systems are often non-linear, for which most common analysis techniques involve some "linearization" step, since the resulting linear approximation can be treated using well-developed numerical and symbolic methods.

* This work was partially funded by the ANR project VEDECY.

** VERIMAG is a joint laboratory of CNRS, Université Joseph Fourier and Grenoble INP.

T. Bultan and P.-A. Hsiung (Eds.): ATVA 2011, LNCS 6996, pp. 473–481, 2011.

In this article, we study affine hybrid automata with uncertainty and propose a method for computing invariants of such systems. Such an invariant, being a conservative approximation of the reachable set, can be used to verify safety proterties.

Hybrid automata with linear continuous dynamics have been a focus in hybrid systems verification, and a number of tools for verifying such systems have been developed [1, 2, 5, 8, 9]. The state-of-the-art reachability computation techniques can efficiently handle *continuous systems* described by linear differential equations with uncertain inputs of up to a few hundreds of variables [7]. However, their extension to handle *hybrid systems* is still limited. *Unbounded time* reachability analysis of hybrid systems with linear continuous dynamics remains a challenge.

The novelty of our approach lies in its ability to efficiently handle *unbounded time verification*. Indeed, by exporting abstract interpretation techniques in hybrid systems verification, we avoid the complexity of the step-by-step approximations of reachable sets in the continuous phase. Our work is close in spirit to the works on barrier certificates [10], polynomial invariants [14] and, in particular polyhedral invariants [12]. Computationally, an important advantage of our approach is the application of efficient techniques for computing invariants and abstract semantics, initially developed for program analysis, to verify hybrid systems.

2 Affine Hybrid Automata with Uncertainty

The set of real numbers is denoted by $\mathbb{R}$. The complete linearly ordered set $\mathbb{R} \cup \{-\infty, \infty\}$ is denoted by $\overline{\mathbb{R}}$. The transpose of a matrix A is denoted by $A^\top$. We denote the i-th row (resp. the j-th column) of a matrix A by $A_{i\cdot}$ (resp. $A_{\cdot j}$). Accordingly, $A_{i\cdot j}$ denotes the component in the i-th row and the j-th column. We also use this notation for vectors and functions $f : X \to Y^k$, i.e., $f_{i\cdot}(x) = (f(x))_{i\cdot}$ for all $x \in X$ and all $i \in \{1, \ldots, k\}$. For $x, y \in \overline{\mathbb{R}}^n$, we write $x \leq y$ iff $x_{i\cdot} \leq y_{i\cdot}$ for all $i \in \{1, \ldots, n\}$. The complete lattice $\overline{\mathbb{R}}^n$ is partially ordered by $\leq$. We write $x < y$ iff $x \leq y$ and $x \neq y$. The elements x and y are called *comparable* iff $x \leq y$ or $y \leq x$. Let $\mathbb{D}$ be a partially ordered set. We denote the *least upper bound* and the *greatest lower bound* of a set $X \subseteq \mathbb{D}$ by $\bigvee X$ and $\bigwedge X$, respectively, provided that they exist. Their existence is in particular guaranteed if $\mathbb{D}$ is a *complete lattice*. The least element $\bigvee \emptyset$ (resp. the greatest element $\bigwedge \emptyset$) is denoted by $\bot$ (resp. $\top$), provided that it exists. We define the binary operators $\vee$ and $\wedge$ by $x \vee y := \bigvee\{x, y\}$ and $x \wedge y := \bigwedge\{x, y\}$ for all $x, y \in \mathbb{D}$, respectively. If $\mathbb{D}$ is a *linearly ordered set* (for instance $\mathbb{R}$ or $\overline{\mathbb{R}}$), then $\vee$ is the *maximum* operator and $\wedge$ the *minimum* operator. A function $f : \mathbb{D}_1 \to \mathbb{D}_2$, where $\mathbb{D}_1$ and $\mathbb{D}_2$ are partially ordered sets, is called *monotone* iff $x \leq y$ implies $f(x) \leq f(y)$ for all $x, y \in \mathbb{D}_1$. The fixpoint theorem of Knaster/Tarski [13] states that any monotone self-map $f : \mathbb{D} \to \mathbb{D}$ on a complete lattice $\mathbb{D}$ has a least fixpoint $\mu f = \bigwedge\{x \in \mathbb{D} \mid x \geq f(x)\}$.

A mapping $V : \mathbb{R}^n \to 2^{\mathbb{R}^n}$ is called a *vector field with uncertainty over* $\mathbb{R}^n$. It assigns a set $V(x) \subseteq \mathbb{R}^n$ of vectors to each state $x \in \mathbb{R}^n$. We denote the set $\{x \in \mathbb{R}^n \mid V(x) \neq \emptyset\}$ by $\mathsf{dom}(V)$. A vector field with uncertainty over $\mathbb{R}^n$ is called *affine* iff there exists some convex polyhedron $P \subseteq \mathbb{R}^{2n}$ such that $V(x) = \{x' \in \mathbb{R}^n \mid (x, x') \in P\}$ for all $x \in \mathbb{R}^n$. The set $\mathsf{dom}(V)$ is a convex polyhedron, whenever V is affine. In the remainder of this article we assume w.l.o.g. that all affine vector fields with uncertainty are

specified by existentially quantified conjunctions of non-strict inequalities and equalities with free variables x and x' that take values from $\mathbb{R}^n$. We say that a continuous differentiable time trajectory $\tau : [0, \delta] \to \mathbb{R}^n$ ($\delta \in \mathbb{R}_{\geq 0}$) *evolves from* $\tau(0)$ *to* $\tau(\delta)$ *according to the vector field with uncertainty* V iff $\dot{\tau}(t) \in V(\tau(t))$ for all $t \in [0, \delta)$.

An *affine hybrid automaton with uncertainty* differs from an affine hybrid automaton on the description of the continuous dynamics. They are now described by affine vector fields with uncertainty instead of ordinary affine vector fields: A *hybrid automaton with uncertainty* $\Psi = (n, \mathbf{L}, \mathcal{T}, \Theta, \mathbf{D}, l_0)$ consists of the following components: n is the number of *continuous variables*. $\mathbf{L}$ is a finite set of *locations*. $l_0 \in \mathbf{L}$ is the *initial location*. $\mathcal{T}$ is a finite set of *discrete transitions*. Each transition $(l_1, \Xi, l_2) \in \mathcal{T}$ consists of a move from the location $l_1 \in \mathbf{L}$ to the location $l_2 \in \mathbf{L}$, and an assertion $\Xi \subseteq (\mathbb{R}^n)^2$. $\Theta \subseteq \mathbb{R}^n$ is the set of possible *initial values* of the continuous variables at l_0. $\mathbf{D}$ is a mapping that maps each $l \in \mathbf{L}$ to a vector field with uncertainty $\mathbf{D}(l)$.

At each location $l \in \mathbf{L}$, the values of the continuous variables evolve according to $\mathbf{D}(l)$. A hybrid automaton with uncertainty $\Psi = (n, \mathbf{L}, \mathcal{T}, \Theta, \mathbf{D}, l_0)$ is called *affine* iff the following statements are fulfilled: (1) The initial condition Θ and all transition relations Ξ are convex polyhedra (we identify $(\mathbb{R}^n)^2$ with $\mathbb{R}^{2n}$). (2) The dynamics $\mathbf{D}(l)$ at each location $l \in \mathbf{L}$ is an affine vector field with uncertainty. In the following we will assume that all convex polyhedra are specified by existentially quantified conjunctions of linear equalities and non-strict linear inequalities.

A computation is a possibly infinite sequence $(l_0, x_0), (l_1, x_1), \ldots$, where $x_0 \in \Theta$ and, for all $i \in \mathbb{N}$, one of the following statements hold: *(Discrete Consecution)* There exists a discrete transition $(l_i, \Xi, l_{i+1}) \in \mathcal{T}$ such that $(x_i, x_{i+1}) \in \Xi$. *(Continuous Consecution)* $l_i = l_{i+1}$ and there exists a $\delta \in \mathbb{R}_{>0}$ and a continuous differentiable time trajectory $\tau : [0, \delta]$ that evolves from x_i to x_{i+1} according to $\mathbf{D}(l_i)$.

As an abstract domain [3] we use template polyhedra as introduced by Sankaranarayanan et al. [11]. For that we fix a *template constraint matrix* $T \in \mathbb{R}^{m \times n}$, where we w.l.o.g. assume that $T_{i\cdot} \neq (0, \ldots, 0)$ for every $i \in \{1, \ldots, m\}$. Each row of T represents a linear template (a linear function). Each template relates n variables. The *concretization* $\gamma_T : \overline{\mathbb{R}}^m \to 2^{\mathbb{R}^n}$ and the *abstraction* $\alpha_T : 2^{\mathbb{R}^n} \to \overline{\mathbb{R}}^m$ are defined by $\gamma_T(d) := \{x \in \mathbb{R}^n \mid Tx \leq d\}$ for all $d \in \overline{\mathbb{R}}^m$, and $\alpha_T(X) := \min\{d \in \overline{\mathbb{R}}^m \mid \gamma_T(d) \supseteq X\}$ for all $X \subseteq \mathbb{R}^n$. We omit the subscripts T, whenever they are clear from the context. As shown by Sankaranarayanan et al. [11], α and γ form a Galois connection.Hence, $\alpha \circ \gamma$ is a downward closure operator, and $\gamma \circ \alpha$ is an upward closure operator. This in particular implies that $\alpha \circ \gamma$ and $\gamma \circ \alpha$ are monotone. In order to simplify notations, we denote $\alpha \circ \gamma$ by $\mathbf{cl}$. The abstract elements from $\alpha(2^{\mathbb{R}^n}) = \mathbf{cl}(\overline{\mathbb{R}}^m)$ are called *closed*. The convex polyhedra from the set $\gamma(\overline{\mathbb{R}}^m) = \gamma(\alpha(2^{\mathbb{R}^n}))$ are called *template polyhedra*.

For all $X \subseteq \mathbb{R}^n$, we moreover define the operator $\mathbf{cl}^X$ on $\overline{\mathbb{R}}^m$ by $\mathbf{cl}^X(d) := \alpha(\gamma(d) \cap X)$ for all $d \in \overline{\mathbb{R}}^m$. The operator $\mathbf{cl}^X$ is a downward closure operator. Moreover, note that $\mathbf{cl}^{\mathbb{R}^n} = \mathbf{cl}$. Similar to Sankaranarayanan et al. [11] we get

$$\mathbf{cl}^X_{i\cdot}(d) = \sup\{T_{i\cdot}.x \mid x \in X \text{ and } Tx \leq d\} \quad \forall X \subseteq \mathbb{R}^n, i \in \{1, \ldots, m\}, d \in \overline{\mathbb{R}}^m. \quad (1)$$

Let V be a vector field with uncertainty over $\mathbb{R}^n$. A set $X \subseteq \mathbb{R}^n$ is called an *invariant* of V iff every trajectory that starts in X and evolves according to V stays in X. Before going further, we introduce the following notation: For all $d \in \overline{\mathbb{R}}^m$ and all

$R \subseteq \{1,\ldots,m\}$, we define $d|_R \in \overline{\mathbb{R}}^m$ by $(d|_R)_{i\cdot} = d_{i\cdot}$, if $i \in R$, and $(d|_R)_{i\cdot} = \infty$, if $i \notin R$ (for all $i \in \{1,\ldots,m\}$).

Assume now that the vector field with uncertainty V is affine. A template polyhedron $P \in \gamma(\overline{\mathbb{R}}^m)$ is called a *positive invariant* of V iff there exists some $R \subseteq \{1,\ldots,m\}$ such that the following properties are fulfilled: (1) $T_{i\cdot}v \leq 0$ for all $v \in V(x)$ and all $x \in P$ with $T_{i\cdot}x = \alpha_{i\cdot}(P)$ and all $i \in R$. (2) $P \supseteq \gamma(\alpha(P)|_R) \cap \mathsf{dom}(V)$.

Each $i \in \{1,\ldots,m\}$ stands for a face of the template polyhedron P. Condition 1 ensures that there is no point x on the face i such that some vector from $V(x)$ points to the outside. Condition 2 ensures that all faces i that are not from R are implied by the faces from R and the staying condition $\mathsf{dom}(V)$.

We emphasize that our definition of positive invariants differs from the ones we used in [4]. In [4], we assumed that the staying condition is a template polyhedron that is represented by a vector from $\overline{\mathbb{R}}^m$. Our new definition does not require this precondition to be fulfilled. We do so, because the staying condition $\mathsf{dom}(V)$ is obtained from V by projecting out variables. However, we want to avoid this, since it might be costly to compute the templates that are necessary to fulfill that precondition (polynomial-time algorithms for projecting out a set of variables are not known). Hence, we cannot w.l.o.g. assume that $\mathsf{dom}(V)$ is a template polyhedron. The advantage of our new definition is that it does not require such technical preconditions.

We emphasize that every template polyhedron that is positive invariant according to the definition in [4] is also a positive invariant according to the definition in this article, i.e., the above definition gives us additional precision. The two notions coincide, whenever $\mathsf{dom}(V)$ is a template polyhedron.

Our goal is to compute the abstract semantics for affine hybrid automata with uncertainty w.r.t. given linear templates. The *abstract semantics* for the affine hybrid automaton with uncertainty $\Psi = (n, \mathbf{L}, \mathcal{T}, \Theta, \mathbf{D}, l_0)$ (w.r.t. given linear templates that are specified by T) is the point-wise minimal mapping $V^\sharp_\square$ that maps every location $l \in \mathbf{L}$ to a template polyhedron $V^\sharp_\square[l] \in \gamma(\overline{\mathbb{R}}^m)$ and fulfills the following constraints: (1) $V^\sharp_\square[l_0] \supseteq \Theta$. (2) $V^\sharp_\square[l]$ is a positive invariant of $\mathbf{D}(l)$ for every location $l \in \mathbf{L}$. (3) $x' \in V^\sharp_\square[l']$ for all $(l, \Xi, l') \in \mathcal{T}$ and all $(x, x') \in \Xi$ with $x \in V^\sharp_\square[l]$. The existence of such a point-wise minimal mapping will be ensured by our findings.

In order to verify safety properties, a problem one is interested in is *abstract reachability*, which is the following decision problem: Decide whether or not, for a given template constraint matrix $T \in \mathbb{R}^{m\times n}$, a given affine hybrid automaton with uncertainty $\Psi = (n, \mathbf{L}, \mathcal{T}, \Theta, \mathbf{D}, l_0)$, and a given location $l \in L$, the statement $V^\sharp_\square[l] \neq \emptyset$ holds. The location l may represent an unsafe state. The decision problem then answers the question, whether or not the unsafe state can be reached within the abstraction. The system is safe, whenever this is not the case. If the unsafe state can be reached within the abstraction, then either the system is unsafe or the abstraction is too coarse.

It is important to note that most existing hybrid systems verification techniques were developed first for purely continuous systems (defined by ordinary differential equations) and were then adapted with some loss of precision to handle staying conditions in hybrid automata. Our approach, in contrast, can handle in a unified manner differential equations and differential algebraic inequalities (i.e. inequalities involving differential and algebraic variables).

3 From Affine Hybrid Automata to Affine Programs

The Time Elapse Operation We will firstly prepare our main result by studying the time elapse operation. We will basically extend the results of Dang and Gawlitza [4] by allowing *uncertainty*. Let V be an affine vector field with uncertainty. Firstly, we define the operator Δ^V on $\overline{\mathbb{R}}^m$ by $\Delta^V_{k\cdot}(d) := \sup\{T_{k\cdot}v \mid x \in \mathbb{R}^n,\ Tx \leq d,\ T_{k\cdot}x \geq d_{k\cdot},\ v \in V(x)\}$ for all $k \in \{1,\dots,m\}$ and all $d \in \overline{\mathbb{R}}^m$ with $d_{k\cdot} < \infty$. Note that $\Delta^V_{k\cdot}(d) = -\infty$, whenever $\{v \in \mathbb{R}^n \mid x \in \mathbb{R}^n, Tx \leq d, T_{k\cdot}x \geq d_{k\cdot}, v \in V(x)\} = \emptyset$. This is in particular fulfilled, if there exists some $i \in \{1,\dots,m\}$ with $d_{i\cdot} = -\infty$. Moreover, we set $\Delta^V_{k\cdot}(d) := 0$ for all $k \in \{1,\dots,m\}$ and $d \in \overline{\mathbb{R}}^m$ with $d_{k\cdot} = \infty$. Intuitively, $\Delta^V_{k\cdot}(d) > 0$ iff there exists some point x on the face $\mathcal{F} := \{x \in \mathbb{R}^n \mid Tx \leq d, T_{k\cdot}x \geq d_{k\cdot}\}$ such that some vector $v \in V(x)$ points to the outside. For all $\epsilon \in \mathbb{R}^m_{>0}$, we define the operator $f^{V,\epsilon}$ on $\overline{\mathbb{R}}^m$ by $f^{V,\epsilon}(d) := d + \epsilon^\top \Delta^V(d)$ for all $d \in \overline{\mathbb{R}}^m$. An application of the operator $f^{V,\epsilon}$ corrects the bounds to the templates according to the vector field with uncertainty V. Note that the staying condition (a.k.a. location invariant) $\mathsf{dom}(V)$ is not completely taken into account so far. More precisely, we have not taken care of the second requirement of the definition of positive invariants. This will be done through the operator $\mathbf{cl}^{\mathsf{dom}(V)}$. Similarly to Dang and Gawlitza [4], we get:

Lemma 1. *Let $\epsilon \in \mathbb{R}^m_{>0}$ and $d \in \overline{\mathbb{R}}^m$. The template polyhedron $\gamma(d)$ is a positive invariant of V iff $d \geq \mathbf{cl}^{\mathsf{dom}(V)}(\mathbf{cl}(d) \vee f^{V,\epsilon}(\mathbf{cl}(d)))$.* □

In order to use the above lemma within a monotone framework, we have to ensure that $f^{V,\epsilon} \circ \mathbf{cl}$ is monotone. Then $f^{V,\epsilon} \circ \mathbf{cl}^{\mathsf{dom}(V)}$ and $\mathcal{F} := \mathbf{cl}^{\mathsf{dom}(V)} \circ (\mathbf{cl} \vee f^{V,\epsilon} \circ \mathbf{cl})$ are monotone, too, and the fixpoint theorem of Knaster/Tarski [13] can be applied.[1] The operator $f^{V,\epsilon} \circ \mathbf{cl}$ is monotone on $\overline{\mathbb{R}}^m$, whenever the operator $f^{V,\epsilon}$ is monotone on $\mathbf{cl}(\overline{\mathbb{R}}^m)$ (It is not always possible to choose an ϵ such that $f^{V,\epsilon}$ is monotone on $\overline{\mathbb{R}}^m$). Analogously to Dang and Gawlitza [4], we get:

Lemma 2 (Monotonicity of $f^{V,\epsilon}$). *In polynomial time we can compute an $\epsilon^{(0)} \in \mathbb{R}^m_{>0}$ such that $f^{V,\epsilon}$ is monotone on $\mathbf{cl}(\overline{\mathbb{R}}^m)$, whenever $\epsilon \leq \epsilon^{(0)}$.* □

Because of Lemma 2, we from now on assume that we have chosen an $\epsilon \in \mathbb{R}^m_{>0}$ such that $f^{V,\epsilon} \circ \mathbf{cl}$ and thus finally $\mathbf{cl}^{\mathsf{dom}(V)} \circ (\mathbf{cl} \vee f^{V,\epsilon} \circ \mathbf{cl}) = \mathbf{cl}^{\mathsf{dom}(V)} \circ (\mathsf{id} \vee f^{V,\epsilon}) \circ \mathbf{cl}$ is monotone. Therefore, for all sets $\Theta \subseteq \mathbb{R}^n$ of values, there exists a least positive invariant P of V which is a superset of Θ. It is given by $\gamma(\mu(\alpha(\Theta) \vee \mathbf{cl}^{\mathsf{dom}(V)} \circ (\mathbf{cl} \vee f^{V,\epsilon} \circ \mathbf{cl})))$. However, we want to have a simpler formulation that allows to perform time elapse operations in polynomial time. In order to obtain such a simpler formulation, we observe that $\mu(\theta \vee \mathbf{cl}^{\mathsf{dom}(V)} \circ (\mathbf{cl} \vee f^{V,\epsilon} \circ \mathbf{cl})) = \mathbf{cl}^{\mathsf{dom}(V)}(\mu(\theta \vee f^{V,\epsilon} \circ \mathbf{cl}^{\mathsf{dom}(V)}))$ for all $\theta \in \mathbf{cl}^{\mathsf{dom}(V)}(\overline{\mathbb{R}}^m)$. Here, θ denotes the function that returns θ for every argument. Putting everything together, we obtain our main result for the time elapse operation:

Theorem 1 (The Time Elapse Operation). *Let V be an affine vector field with uncertainty over $\mathbb{R}^n$, and $\Theta \subseteq \mathbb{R}^n$. Assume that $\epsilon \in \mathbb{R}^m_{>0}$ is chosen such that $f^{V,\epsilon} \circ \mathbf{cl}$ is monotone. The template polyhedron $\gamma(\alpha(\Theta \cup \gamma(\mu(\alpha(\Theta \cap \mathsf{dom}(V)) \vee f^{V,\epsilon} \circ \mathbf{cl}^{\mathsf{dom}(V)}))))$ is the least positive invariant of V which is a superset of Θ.* □

[1] For mappings $f, g : X \to \mathbb{D}$, $f \vee g$ is defined by $(f \vee g)(x) := f(x) \vee g(x)$ for all $x \in X$.

The Abstract Semantic Inequalities We will now set up a system of inequalities over $\overline{\mathbb{R}}^m$ whose least solution corresponds to the abstract semantics of the affine hybrid automaton with uncertainty Ψ. In the next subsection, we will construct an affine program whose abstract semantics gives us the solution of this system of inequalities.

So far, we have ignored the discrete transitions. In order to take them into account, we define an abstract semantics for discrete transitions $(l, \Xi, l') \in \mathcal{T}$. Recall that the assertion $\Xi \subseteq \mathbb{R}^{2n}$ is a convex polyhedron (represented by an existentially quantified conjunction of inequalities with free variables x and x' that take values from $\mathbb{R}^n$). The collecting semantics $[\![\Xi]\!]$ of Ξ is defined by $[\![\Xi]\!](X) := \{y \in \mathbb{R}^n \mid \exists x \in X\,.\,(x, y) \in \Xi\}$ for all $X \subseteq \mathbb{R}^n$. The abstract semantics $[\![\Xi]\!]^\sharp$ of Ξ is defined by $[\![\Xi]\!]^\sharp := \alpha \circ [\![\Xi]\!] \circ \gamma$. The abstract semantics $[\![\Xi]\!]^\sharp$ safely over-approximates the collecting semantics $[\![\Xi]\!]$ and the concrete semantics.

We are now going to define an abstract semantics $V^\sharp$ for an *affine* hybrid automaton $\Psi = (n, \mathbf{L}, \mathcal{T}, \Theta, \mathbf{D}, l_0)$ with uncertainty that corresponds to the abstract semantics $V^\sharp_\square$ of Ψ. The abstract semantics $V^\sharp$ of Ψ is the least solution to the following constraints:

$$
\begin{array}{lll}
\mathbf{A}^\sharp[l_0] \geq \alpha(\Theta) & \mathbf{A}^\sharp[l'] \geq [\![\Xi]\!]^\sharp(\mathbf{V}^\sharp[l]) & \forall (l, \Xi, l') \in \mathcal{T} \\
\mathbf{B}^\sharp[l] \geq \mathbf{cl}^{\mathsf{dom}(\mathbf{D}(l))}(\mathbf{A}^\sharp[l])) & \mathbf{B}^\sharp[l] \geq f^{\mathbf{D}(l),\epsilon(l)}(\mathbf{cl}^{\mathsf{dom}(\mathbf{D}(l))}(\mathbf{B}^\sharp[l])) & \forall l \in \mathbf{L} \\
\mathbf{V}^\sharp[l] \geq \mathbf{A}^\sharp[l] & \mathbf{V}^\sharp[l] \geq \mathbf{cl}^{\mathsf{dom}(\mathbf{D}(l))}(\mathbf{B}^\sharp[l])) & \forall l \in \mathbf{L}
\end{array}
$$

The variables $\mathbf{A}^\sharp[l]$, $\mathbf{B}^\sharp[l]$, and $\mathbf{V}^\sharp[l]$ (for $l \in \mathbf{L}$) take values from $\overline{\mathbb{R}}^m$. $\mathbf{A}^\sharp[l]$ and $\mathbf{B}^\sharp[l]$ are just auxiliary variables. The existence of the least solution is ensured by the fixpoint theorem of Knaster/Tarski, since we assume that, for all locations $l \in \mathbf{L}$, $\epsilon(l) \in \mathbb{R}^m_{>0}$ is chosen such that $f^{\mathbf{D}(l),\epsilon(l)} \circ \mathbf{cl}$ and thus $f^{\mathbf{D}(l),\epsilon(l)} \circ \mathbf{cl}^{\mathsf{dom}(\mathbf{D}(l))}$ are monotone. The existence of such an $\epsilon(l)$ is again ensured by Lemma 2.

The first constraint takes all possible initial values of the continuous variables at the initial location l_0 into account. The second constraint ensures that the template polyhedron $\gamma(V^\sharp[l'])$ contains at least all values that can come through the discrete transition (l, Ξ, l'). The remaining constraint ensure that the template polyhedron $\gamma(V^\sharp[l])$ is a positive invariant of $\mathbf{D}(l)$ (cf. Theorem 1). By construction, we get $V^\sharp_\square[l] = \gamma(V^\sharp[l])$ for all locations $l \in \mathbf{L}$.

The Reduction. We are now going to reduce the problem of computing abstract semantics of affine hybrid automata w.r.t. template polyhedra to the problem of computing abstract semantics of affine programs w.r.t. template polyhedra. An *affine program* is an affine hybrid automaton with uncertainty $\Psi = (n, \mathbf{L}, \mathcal{T}, \Theta, \mathbf{D}, l_0)$, where $\mathbf{D}(l) = \emptyset$ for every location $l \in \mathbf{L}$. That is, only discrete transitions are allowed.

Abstract reachability for affine programs is in coNP (see e.g. Dang and Gawlitza [4]). Moreover, it is known to be at least as hard as computing the winning regions of mean-payoff games (cf. Gawlitza [6]). The latter problem is known to be in $\mathsf{UP} \cap \mathsf{coUP}$, but not known to be in P. It is an open question whether or not abstract reachability for affine programs is $\mathsf{coNP}-\mathsf{hard}$. Hence, it makes sense to ask the question, whether or not abstract reachability for affine hybrid automata with uncertainty is more difficult than abstract reachability for affine programs. In this section, we show that this is not the case by providing a polynomial-time reduction from abstract reachability for affine hybrid automata with uncertainty to abstract reachability for affine programs. Hence,

any efficient algorithm for affine programs gives us an efficient algorithm for affine hybrid automata with uncertainty.

Let $\Psi = (n, \mathbf{L}, \mathcal{T}, \Theta, \mathbf{D}, \mathbf{st})$ be an affine hybrid automaton with uncertainty and $T \in \mathbb{R}^{m \times n}$ be a template constraint matrix. We construct an affine program $\Psi' = (m, \mathbf{L}', \mathcal{T}', \Theta', \mathbf{D}', \mathbf{st}')$ such that we can read off the abstract semantics of Ψ from the abstract semantics of Ψ'. Here, we consider the abstract semantics of Ψ' w.r.t. the template constraint matrix T' that is simply the identity matrix of size m, i.e., we restrict our considerations to upper bounds. We set $\mathbf{L}' := \{l, l_{\mathbf{A}}, l_{\mathbf{B}} \mid l \in \mathbf{L}\}$, i.e., we replace each location of Ψ by three locations. We will use the location l for the variable $\mathbf{V}^\sharp[l]$, the location $l_{\mathbf{A}}$ for the variable $\mathbf{A}^\sharp[l]$, and the location $l_{\mathbf{B}}$ for the variable $\mathbf{B}^\sharp[l]$.

The initial location $\mathbf{st}'$ is the location $\mathbf{st}_{\mathbf{A}}$. The set Θ' of initial states of the affine program Ψ' is given by $\Theta' := \{x \in \mathbb{R}^m \mid x \leq \alpha_T(\Theta)\}$. Hence, $\alpha_{T'}(\Theta') = \alpha_T(\Theta)$. These definitions correspond to the first inequality.

Moreover, we set $\mathbf{D}'(l) := \emptyset$ for all locations $l \in \mathbf{L}$, i.e., we are actually constructing an affine program. The set $\mathcal{T}'$ of discrete transitions is the smallest set that fulfills the following constraints:

1. If $(l, \Xi, l') \in \mathcal{T}$, then $(l, \Xi', l'_{\mathbf{A}}) \in \mathcal{T}'$, where
$$\Xi' := \left\{ \big(d, d'\big) \in (\mathbb{R}^m)^2 \;\middle|\; \exists x, x' \in \mathbb{R}^n \,.\, Tx \leq d,\ \big(x, x'\big) \in \Xi,\ d' \leq Tx' \right\}$$
Recall that Ξ is a convex polyhedron. Therefore, Ξ' is a convex polyhedron. By the construction, we get $\alpha_T([\![\Xi]\!](\gamma_T(d))) = \alpha_{T'}([\![\Xi']\!](\gamma_{T'}(d)))$ for all $d \in \overline{\mathbb{R}}^m$. This discrete transition corresponds to the second inequality.
2. For every location $l \in \mathbf{L}$, we have to add additional discrete transitions in order to deal with the time elapse operation. For simplicity, let $V := \mathbf{D}(l)$. Assume further that $\epsilon \in \mathbb{R}^m_{\geq 0}$ is chosen such that $f^{V,\epsilon} \circ \mathbf{cl}$ is monotone. In order to apply $\mathbf{cl}^{\mathsf{dom}(V)}$, we define the polyhedron $\Xi_{\mathbf{cl}} := \{\big(d, d'\big) \in (\mathbb{R}^m)^2 \mid \exists x \in \mathsf{dom}(V) \,.\, d' \leq Tx,\ Tx \leq d\}$. By construction, we have $\alpha_{T'}([\![\Xi_{\mathbf{cl}}]\!](\gamma_{T'}(d))) = \mathbf{cl}^{\mathsf{dom}(V)}(d)$ for all $d \in \overline{\mathbb{R}}^m$ (see (1)). Hence, we add the discrete transitions $(l_{\mathbf{A}}, \Xi_{\mathbf{cl}}, l_{\mathbf{B}})$ and $(l_{\mathbf{B}}, \Xi_{\mathbf{cl}}, l)$ for the 3rd and the 6th inequality, respectively. For the 5th inequality, we add the discrete transition $(l_{\mathbf{A}}, \Xi_{\mathsf{id}}, l)$, where $\Xi_{\mathsf{id}} := \left\{ \big(d, d'\big) \in (\mathbb{R}^m)^2 \mid d' = d \right\}$. For the 4th inequality, we finally add the discrete transition $(l_{\mathbf{B}}, \Xi, l_{\mathbf{B}})$, where
$$\begin{aligned}
&\Xi := \left\{ \big(d, d'\big) \in (\mathbb{R}^m)^2 \;\middle|\; d' \leq f^{V,\epsilon}(\mathbf{cl}^{\mathsf{dom}(V)}(d)) \right\} \\
&= \left\{ \big(d, d'\big) \in (\mathbb{R}^m)^2 \;\middle|\; \forall k \in \{1, \ldots, m\} \,. \right. \\
&\qquad \left. \exists x \in \mathbb{R}^n,\ v \in V(x) \,.\, d'_{k\cdot} \leq d_{k\cdot} + \epsilon_{k\cdot} T_{k\cdot} v,\ Tx \leq d,\ T_{k\cdot} x \geq d_{k\cdot} \right\} \\
&= \left\{ \big(d, d'\big) \in (\mathbb{R}^m)^2 \;\middle|\; \exists x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^n, v^{(1)} \in V(x^{(1)}), \ldots, v^{(m)} \in V(x^{(m)}) \,. \right. \\
&\qquad \left. \forall k \in \{1, \ldots, m\} \,.\, d'_{k\cdot} \leq d_{k\cdot} + \epsilon_{k\cdot} T_{k\cdot} v^{(k)},\ Tx^{(k)} \leq d,\ T_{k\cdot} x^{(k)} \geq d_{k\cdot} \right\}
\end{aligned}$$
Ξ is a convex polyhedron, and $\alpha_{T'}([\![\Xi]\!](\gamma_{T'}(d))) = f^{V,\epsilon}(\mathbf{cl}^{\mathsf{dom}(V)}(d))\ \forall d \in \overline{\mathbb{R}}^m$.

We finally get: Let $V^\sharp_\square$ denote the abstract semantics of Ψ w.r.t. the template constraint matrix T, and $V^{\sharp\,\prime}_\square$ denote the abstract semantics of Ψ' w.r.t. the template constraint matrix T'. Then $\alpha_T(V^\sharp_\square[l]) = \alpha_{T'}(V^{\sharp\,\prime}_\square[l])$ for all locations $l \in \mathbf{L}$.

The construction contains existential quantifications. This does not cause any problems, since the existential quantifications can be eliminated by introducing at most polynomially many auxiliary program variables (We cannot simply project out the existentially quantified variables, since this could not be carried out in polynomial time). Since the above construction can be carried out in polynomial time, we obtain:

Theorem 2. *Abstract reachability w.r.t. template polyhedra for affine hybrid automata with uncertainty is polynomial-time equivalent to abstract reachability w.r.t. template polyhedra for affine programs.* □

4 Conclusion

In this article, we studied the problem of template-based unbounded time verification of safety properties for affine hybrid automata with uncertainty. This model is used to safely over-approximate non-linear behavior. We showed that, w.r.t. template polyhedra, abstract reachability for affine hybrid automata with uncertainty is polynomial-time reducible to abstract reachability for affine programs. That is, these problems are polynomial-time equivalent. The reduction replaces every time elapse operation by a bunch of discrete transitions forming a loop.

References

[1] Asarin, E., Bournez, O., Dang, T., Maler, O.: Approximate reachability analysis of piecewise linear dynamical systems. In: Lynch, N.A., Krogh, B.H. (eds.) HSCC 2000. LNCS, vol. 1790, pp. 20–31. Springer, Heidelberg (2000)

[2] Chutinan, A., Krogh, B.: Computational techniques for hybrid system verification. IEEE Trans. on Automatic Control (48), 64–75 (2003)

[3] Cousot, P., Cousot, R.: Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In: POPL (1977)

[4] Dang, T., Gawlitza, T.M.: Template-based unbounded time verification of affine hybrid automata. Technical report, VERIMAG (2011)

[5] Frehse, G., Guernic, C.L., Donzé, A., Cotton, S., Ray, R., Lebeltel, O., Ripado, R., Girard, A., Dang, T., Maler, O.: Spaceex: Scalable verification of hybrid systems. In: Gopalakrishnan, G., Qadeer, S. (eds.) CAV 2011. LNCS, vol. 6806, pp. 379–395. Springer, Heidelberg (2011)

[6] Gawlitza, T.M.: Strategieverbesserungsalgorithmen für exakte Programmanalysen, Ph.D. Thesis. Dr. Hut Verlag, München, Munich, Germany (October 2009)

[7] Girard, A., Guernic, C.L., Maler, O.: Efficient computation of reachable sets of linear time-invariant systems with inputs. In: Hespanha, J.P., Tiwari, A. (eds.) HSCC 2006. LNCS, vol. 3927, pp. 257–271. Springer, Heidelberg (2006)

[8] Kurzhanskiy, A., Varaiya, P.: Ellipsoidal techniques for reachability analysis of discrete-time linear systems. IEEE Trans. Automatic Control (52), 26–38 (2007)

[9] Kvasnica, M., Grieder, P., Baotić, M., Morari, M.: Multi-parametric toolbox (mpt). In: Alur, R., Pappas, G.J. (eds.) HSCC 2004. LNCS, vol. 2993, pp. 448–462. Springer, Heidelberg (2004)

[10] Prajna, S., Jadbabaie, A.: Safety verification of hybrid systems using barrier certificates. In: Alur, R., Pappas, G.J. (eds.) HSCC 2004. LNCS, vol. 2993, pp. 477–492. Springer, Heidelberg (2004)

[11] Sankaranarayanan, S., Sipma, H.B., Manna, Z.: Scalable analysis of linear systems using mathematical programming. In: Cousot, R. (ed.) VMCAI 2005. LNCS, vol. 3385, pp. 25–41. Springer, Heidelberg (2005)
[12] Sankaranarayanan, S., Dang, T., Ivančić, F.: A policy iteration technique for time elapse over template polyhedra. In: Egerstedt, M., Mishra, B. (eds.) HSCC 2008. LNCS, vol. 4981, pp. 654–657. Springer, Heidelberg (2008)
[13] Tarski, A.: A lattice-theoretical fixpoint theorem and its appications. Pac. J. Math. 5, 285–309 (1955)
[14] Tiwari, A., Khanna, G.: Nonlinear systems: Approximating reach sets. In: Alur, R., Pappas, G.J. (eds.) HSCC 2004. LNCS, vol. 2993, pp. 600–614. Springer, Heidelberg (2004)