$\Sigma_K constraints for Hybrid Systems$

Korovina, Margarita and Kudinov, Oleg

2010

MIMS EPrint: **2010.40**

Manchester Institute for Mathematical Sciences
School of Mathematics

The University of Manchester

Reports available from: http://eprints.maths.manchester.ac.uk/
And by contacting: The MIMS Secretary
School of Mathematics
The University of Manchester
Manchester, M13 9PL, UK

ISSN 1749-9097

Σ_K–constraints for Hybrid Systems*

Margarita Korovina[1] and Oleg Kudinov[2]

[1] Centre for Interdisciplinary Computational and Dynamical Analysis,
The University of Manchester and IIS SB RAS Novosibirsk
Margarita.Korovina@manchester.ac.uk
[2] Sobolev Institute of Mathematics, Novosibirsk
kud@math.nsc.ru

Abstract. In this paper we introduce and study computational aspects of Σ_K-constraints which are powerful enough to represent computable continuous data, but also simple enough to be an approach to approximate constraint solving for a large class of quantified continuous constraints. We illustrate how Σ_K-constraints can be used for reasoning about hybrid systems.

1 Introduction

A continuous constraint is a logical formalism which is used extensively in modeling, formal analysis and synthesis of control of hybrid systems [1,5,23,25]. From a mathematical point of view a *continuous constraint* is an expression (well formed formula) in an appropriate language over the reals involving constants, variables (ranging over continuous data i.e. the real numbers, functions), operations, relations, logical connectives and quantifiers. Since continuous constraints involve continuous data such as real numbers, functions and sets, solving of such constraints is already a challenging research problem.

This has resulted in various different approaches to continuous constraint solving. There are at least two main nonequivalent models of continuous constraint solving. The first one is related to model theory and real algebraic geometry (e.g. [4,6,7,24]) where real numbers are considered as basic entities which can be added, multiplied, divided or compared in a single step. Here most of methods for continuous constraint solving are exact and based on quantifier elimination and cylindrical cell decomposition. However, this approach is restricted to special cases such as quantified polynomial constraints.

The second model is closely related to numerical and computable analysis (e.g. [2,22,26]), where continuous data (real numbers, real-valued functions) are given by appropriate representations and computations of the solution sets of continuous constraints are infinite processes which produce inner or outer approximations to the results.

This model conforms to our intuition of reals based on rational approximations to a real number, but depends on representations of continuous data [26]

* This research was partially supported by EPSRC grant EP/E050441/1, DFG-RFBR (grant No 436 RUS 113/1002/01, grant No 09-01-91334).

A. Pnueli, I. Virbitskaite, and A. Voronkov (Eds.): PSI 2009, LNCS 5947, pp. 230–241, 2010.

and particular validated numerical techniques [5]. In this paper we introduce and study Σ_K–constraints which formalise problems involving logical quantifiers bounded by computable compact sets, computable real numbers and computable real-valued functions. A key idea of our approach to approximate solving of Σ_K–constraints is based on a procedure which given a Σ_K–constraint produces an effective sequence of quantifier free formulas (inequality polynomial constraints) defining the solution set. On one hand proposed approach agrees with the second model mentioned above, on the other hand it does not depend on the particular representation of real numbers. We illustrate how Σ_K-constraints can be used for studying reachability problems of switched controlled systems.

The paper is structured as follows. In Section 2 we give the main definitions and notions. We recall properties of Σ-definability and computability over the real numbers. In Section 3 we introduce the notion of Σ_K–constraints and propose an approach to approximate Σ_K–constraint solving. In Section 4 we recall slightly modified **SHS**-specifications of switched controlled systems introduced in [18] and show that under natural assumptions the behaviour of a hybrid system is computable. We illustrate how Σ_K-constraints can be used for studying reachability problems.

2 Basic Notions and Definitions

In order to introduce Σ_K-constraints we propose a basic model, recall the notions and properties of Σ-definability and computability over the reals.

2.1 Basic Model

Our approach to continuous constraints is based on the notion of definability [14,15], where continuous objects and computational processes involving these objects can be defined using finite formulas in a suitable structure. Definability has been a very successful framework for generalised computability theory, descriptive complexity and databases. One of the most interesting and practically important types of definability is Σ-definability, which generalises recursive enumerability over the natural numbers [3,8]. However, the most developed part of definability theory deals with abstract structures with equality (i.e., the natural numbers, trees, automata, etc.). In the case of continuous data, such as real numbers, real-valued functions and functionals, it is reasonable to consider the corresponding structures without equality. This is motivated by the following natural reason. In all effective approaches to exact real number computation via concrete representations [22,26], the equality test is undecidable. In order to do any kind of computation or to develop a computability theory, one has to work within a structure rich enough for information to be coded and stored. For this purpose we extend the structure $\mathbb{R}$ by the set of hereditarily finite sets $\mathrm{HF}(\mathbb{R})$. The idea that the hereditarily finite sets over a structure form a natural domain for computation is discussed in [3,8]. Note that such or very similar extensions of structures are used in the theory of abstract state machines [11], in query languages for hierarchic databases [20].

According to this motivation we consider the ordered structure of the real numbers in *the finite predicate language*, $\langle \mathbb{R}, \sigma_P, < \rangle = \langle \mathbb{R}, \sigma_< \rangle$, with $\sigma_P \supseteq \{\mathcal{M}^*_E, \mathcal{M}^*_H, \mathcal{A}^+_E, \mathcal{A}^+_H, \}$, where $\mathcal{M}^*_E, \mathcal{M}^*_H$ are interpreted as an open epigraph and an open hypograph of multiplication respectively, and $\mathcal{A}^+_E, \mathcal{A}^+_H$ are interpreted as an open epigraph and an open hypograph of addition respectively. *We don't assume that the language σ_P contains equality.*

We extend the real numbers by the set of hereditarily finite sets $\mathrm{HF}(\mathbb{R})$ which is rich enough for information to be coded and stored. We construct the set of hereditarily finite sets, $\mathrm{HF}(\mathbb{R})$ over the reals, as follows:

1. $\mathrm{HF}_0(\mathbb{R}) \rightleftharpoons \mathbb{R}$,
2. $\mathrm{HF}_{n+1}(\mathbb{R}) \rightleftharpoons \mathcal{P}_\omega(\mathrm{HF}_n(\mathbb{R})) \cup \mathrm{HF}_n(\mathbb{R})$, where $n \in \omega$ and for every set B, $\mathcal{P}_\omega(B)$ is the set of all finite subsets of B.
3. $\mathrm{HF}(\mathbb{R}) = \bigcup_{m \in \omega} \mathrm{HF}_m(\mathbb{R})$.

We define $\mathbf{HF}(\mathbb{R})$ as the following model: $\mathbf{HF}(\mathbb{R}) \rightleftharpoons (\mathrm{HF}(\mathbb{R}), U, \sigma_<, \emptyset, \in) \rightleftharpoons (\mathrm{HF}(\mathbb{R}), \sigma)$, where the constant $\emptyset$ stands for the empty set and the binary predicate symbol $\in$ has the set-theoretic interpretation. We also add a 1-ary predicate symbol U naming the set of urelements (the real numbers). The natural numbers $0, 1, \ldots$ are identified with the (finite) ordinals in $\mathrm{HF}(\mathbb{R})$ i.e. $\emptyset, \{\emptyset, \{\emptyset\}\}, \ldots$, so in particular, $n+1 = n \cup \{n\}$ and the set ω is a subset of $\mathrm{HF}(\mathbb{R})$.

The atomic formulas include $U(x)$, $\neg U(x)$, $x < y$, $x \in s$, $x \notin s$ where s ranges over sets, and also, for every $Q_i \in \sigma_P$ with the arity n_i, $Q_i(x_1, \ldots, x_{n_i})$ which has the following interpretation:

$$\mathbf{HF}(\mathbb{R}) \models Q_i(x_1, \ldots, x_{n_i}) \text{ if and only if}$$
$$\mathbb{R} \models Q_i(x_1, \ldots, x_{n_i}) \text{ and, for every } 1 \leq j \leq n_i,\ x_j \in \mathbb{R}.$$

The set of Δ_0-*formulas* is the closure of the set of atomic formulas under $\wedge, \vee$, bounded quantifiers $(\exists x \in y)$ and $(\forall x \in y)$, where $(\exists x \in y)\ \Psi$ means the same as $\exists x (x \in y \wedge \Psi)$ and $(\forall x \in y)\ \Psi$ as $\forall x (x \in y \to \Psi)$ where y ranges over sets. The set of Σ-*formulas* is the closure of the set of Δ_0-formulas under $\wedge, \vee$, $(\exists x \in y)$, $(\forall x \in y)$ and $\exists$, where y ranges over sets.

Remark 1. It is worth noting that all predicates $Q_i \in \sigma_P$ and $<$ occur only positively in Σ-formulas. Hence, if σ_P does not contain equality then in Σ-formulas we don't allow equality on the urelements (elements from $\mathbb{R}$).

Definition 1. *1. A relation $B \subseteq \mathrm{HF}(\mathbb{R})^n$ is Σ-definable if there exists a Σ-formula $\Phi(\bar{a})$ such that $\bar{b} \in B \leftrightarrow \mathbf{HF}(\mathbb{R}) \models \Phi(\bar{b})$.*

2.2 Basic Properties Σ-definability over the Reals

In this subsection we recall the basic principles for Σ-definability which allow to make effective reasoning about continuous constraints using Σ-formulas.

2.3 Gandy's Theorem and Inductive Definitions

Let us recall Gandy's Theorem for $\mathbf{HF}(\mathbb{R})$ which shows that continuous objects and computational processes involving these objects can be defined using Σ-formulas. Let $\Phi(a_1, \ldots, a_n, P)$ be a Σ-formula, where P occurs positively in Φ and the arity of Φ is equal to n. We think of Φ as defining an *effective operator* $\Gamma : \mathcal{P}(\mathrm{HF}(\mathbb{R})^n) \to \mathcal{P}(\mathrm{HF}(\mathbb{R})^n)$ given by $\Gamma(Q) = \{\bar{a} | (\mathbf{HF}(\mathbb{R}), Q) \models \Phi(\bar{a}, P)\}$. Since the predicate symbol P occurs only positively the corresponding operator Γ is monotone, i.e., from $B \subseteq C$ implies $\Gamma(B) \subseteq \Gamma(C)$. By monotonicity, the operator Γ has a least (w.r.t. inclusion) fixed point which can be described as follows. We start from the empty set and apply operator Γ until we reach the fixed point: $\Gamma^0 = \emptyset$, $\Gamma^{n+1} = \Gamma(\Gamma^n)$, $\Gamma^\gamma = \cup_{n<\gamma}\Gamma^n$, where γ is a limit ordinal.

One can easily check that the sets Γ^n form an increasing chain of sets: $\Gamma^0 \subseteq \Gamma^1 \subseteq \ldots$. By set-theoretical reasons, there exists the least ordinal γ such that $\Gamma(\Gamma^\gamma) = \Gamma^\gamma$. This Γ^γ is the least fixed point of the given operator Γ.

Theorem 1. *[15][Gandy's Theorem for* $\mathbf{HF}(\mathbb{R})$*]*
Let $\Gamma : \mathcal{P}(\mathrm{HF}(\mathbb{R})^n) \to \mathcal{P}(\mathrm{HF}(\mathbb{R})^n)$ *be an effective operator. Then the least fixed-point of* Γ *is* Σ*-definable and the least ordinal such that* $\Gamma(\Gamma^\gamma) = \Gamma^\gamma$ *is less or equal to* ω*.*

Definition 2. *A relation* $B \subset \mathbb{R}^n$ *is called* Σ*-inductive if it is the least-fixed point of an effective operator.*

Corollary 1. *Every* Σ*-inductive relation is* Σ*-definable.*

2.4 Universal Σ-predicate

The following result shows that we can effectively check validity of a Σ-formula on $\mathbf{HF}(\mathbb{R})$. As a corollary there exists a universal Σ-predicate for Σ-formulas over this model.

Theorem 2. *[14] There exists a binary* Σ*-definable predicate* Tr *such that for any* $n \in \omega$ *and* $A \in \mathrm{HF}(\mathbb{R})$ *we have that* $(n, A) \in Tr$ *if and only if* n *is the Gödel number of a* Σ*-formula* Φ*,* γ_A *is a correct interpretation for free variables of* Φ *and* $\mathbf{HF}(\mathbb{R}) \models \Phi[\gamma_A]$*.*

2.5 Semantic Characterisation of Σ-definability

The following theorem reveals algorithmic properties of Σ-formulas over $\mathbf{HF}(\mathbb{R})$.

Theorem 3. *[14][Semantic Characterisation of* Σ*-definability]*
A set $B \subseteq \mathbb{R}^n$ *is* Σ*-definable if and only if there exists an effective sequence of quantifier free formulas in the language* $\sigma_<$*,* $\{\Phi_s(x_1, \ldots, x_n)\}_{s\in\omega}$*, such that*

$$(x_1, \ldots, x_n) \in B \leftrightarrow \mathbb{R} \models \bigvee_{s\in\omega} \Phi_s(x_1, \ldots, x_n).$$

The proof of this theorem is based on Gandy's theorem and existence of Σ-universal predicate. It is worth noting that both of the directions of this characterisation are important. The right direction gives us an effective procedure which generates quantifier free formulas approximating Σ-relations. The converse direction provides tools for descriptions of the results of effective infinite approximating processes by finite formulas.

2.6 Computability and Σ-definability over $\mathbb{R}$

In order uniformly characterise computability of different continuous data in logical terms, we consider an arbitrary structure $\mathcal{A} = \langle A, \sigma_P, \neq\rangle = \langle A, \sigma_A\rangle$, where A contains more than one element, and σ_P is a finite set of basic predicates. We assume that the existential theory of $\mathcal{A}$ is computably enumerable. For the structure $\mathcal{A}$, we introduce a topology τ_Σ, with the base consisting of the subsets defined by existential formulas with positive occurrences of basic predicates and $\neq$. As examples we can consider the real numbers without equality $\mathbb{R}_< = \langle \mathbb{R}, \sigma_<\rangle$, the real numbers with equality $\mathbb{R}_= = \langle \mathbb{R}, +, *, \leq\rangle$, the real-valued continues functions $C(\mathbb{R}) = \langle C(\mathbb{R}), P_1, \ldots, P_{12}, \neq\rangle$ [12]. We denote Σ-definability in the language σ as Σ-definability in σ. For the definitions of computable real numbers, computable functions, and computable compact sets we refer to [12,19,22,26]. The following theorems connect computable continuous data with validity of Σ–formulas.

Proposition 1. *[17] A real number is computable if and only if the left Dedekind cut and the right Dedekind cut are Σ-definable in $\sigma_<$.*

Theorem 4. *If $f \in C[0,1]$ and its epigrap and hypograph are Σ-definable in $\sigma_=$ then f is computable.*

Proposition 2. *[17] A total function $F : \mathbb{R} \to \mathbb{R}$ is computable if and only if its epigraph and hypograph are Σ-definable in $\sigma_<$.*

Definition 3. *A total continuous function $F : A \times \mathbb{R} \to \mathbb{R}$ is called weakly computable if there exist effective infinite sequences $\{\langle \psi_m^-(x), \phi_m^-(y,z)\rangle\}_{m\in\omega}$ and $\{\langle \psi_m^+(x), \phi_m^+(y,z)\rangle\}_{m\in\omega}$ of Σ-formulas, where $\psi_m^-(x)$ and $\psi_m^+(x)$ are Σ-formulas in σ_A, $\phi_m^-(y,z)$ and $\phi_m^+(y,z)$ are Σ-formulas in $\sigma_=$ such that*

$$F(x,y) < z \leftrightarrow \bigvee_{m\in\omega} (\psi_m^-(x) \wedge \phi_m^-(y,z)) \text{ and}$$
$$F(x,y) > z \leftrightarrow \bigvee_{m\in\omega} (\psi_m^+(x) \wedge \phi_m^+(y,z)).$$

It is worth noting that the computable functions is a proper subclass of the weakly computable functions.

Theorem 5. *Let $F : A\times\mathbb{R} \to \mathbb{R}$ be a weakly computable continuous function. If there exists a computable function $H : A \times \mathbb{R} \to \mathbb{R}$ such that $|F(x,y)| \leq H(x,y)$ for all $x \in A$ and $y \in \mathbb{R}$ then F is computable.*

Proposition 3. *[2] A compact subset $K \subset \mathbb{R}^n$ is computable if and only if the distance function d_K is computable and there exist rational numbers q_1 and q_2 such that $K \subseteq [q_1, q_2]^n$.*

3 Σ_K-constraints

Now we consider the real numbers $\mathbb{R}$ in an extended language σ. Define $\sigma = \sigma_P \cup \sigma_c \cup \sigma_f \cup \sigma_K = (0, 1, \cdot, +, <, c_1, \ldots, c_k, \ldots, f_1, \ldots, f_n, \ldots, K_1, \ldots, K_m, \ldots)$, where c_i is a computable real number, f_j is a computable function, and K_s is a computable compact subset of $\mathbb{R}^n$.
The atomic Σ_K-constraints include $p(\bar{x}) < q(\bar{y})$, $f_i(\bar{x}) < f_j(\bar{y})$, where $\bar{x}$ and $\bar{y}$ range over the real numbers, p and q are polynomials with computable real coefficients, f_i and f_j are computable real functions.
The set of *Σ_K-constraints* is the closure of the set of atomic Σ_K-constraints under $\wedge, \vee$, existential quantifiers $\exists x$ and bounded quantifiers $(\exists x \in K_s)$ and $(\forall x \in K_m)$, where K_s and K_m are computable compact subset of $\mathbb{R}^n$.

Remark 2. By definition, Σ_K–constraints involve continuous data such as variables ranging over the real numbers, computable real constants, computable real-valued functions, the strict inequality relation $<$, logical connectives $\vee$, $\wedge$ and quantifiers bounded by computable compact sets. It is worth noting that the predicate $<$ occurs only positively in Σ_K–constraints.

Theorem 6. *There is an algorithm which by a Σ_K–constraint φ produces an effective sequence of quantifier free formulas $\{\psi_i\}_{i\in\omega}$ in the language $\sigma_<$ such that*

$$\mathbb{R} \models \varphi(\bar{x}) \leftrightarrow \mathbb{R} \models \bigvee_{i\in\omega} \psi_i(\bar{x}).$$

First we prove the following proposition.

Proposition 4. *For every Σ-formula φ there exists a Σ-formula ψ such that*

$$\mathbf{HF}(\mathbb{R}) \models \forall x \in [a, b]\varphi(x, y_1, \ldots, y_n) \text{ iff } \mathbf{HF}(\mathbb{R}) \models \psi(a, b, y_1, \ldots, y_n),$$

where free variables range over $\mathbb{R}$.

Proof. First we consider the case of $\exists$-formulas in the language $\sigma_{\mathbb{R}}$. Using induction on the structure of a $\exists$-formula φ, we show how to obtain a required formula ψ. Then, based on Theorem 3 we construct a required formula ψ for an arbitrary Σ-formula.
Atomic case. We consider nontrivial subcases.
a) If $\varphi(x, z) \rightleftharpoons x \cdot x > z$ then

$$\psi(a, b, z) \rightleftharpoons z < 0 \vee a > b \vee (a > 0 \wedge b > 0 \wedge a \cdot a > z) \vee (a < 0 \wedge b < 0 \wedge b \cdot b > z)\,.$$

b) If $\varphi(x, z) \rightleftharpoons x \cdot x < z$ then $\psi(a, b, z) \rightleftharpoons a > b \vee (a \cdot a < z \wedge b \cdot b < z)\,.$

c) If $\varphi(x,y) \rightleftharpoons x \cdot y > x$ then

$$\psi(a,b,z) \rightleftharpoons a > b \vee (a > 0 \wedge b > 0 \wedge y > 1) \vee (a < 0 \wedge b < 0 \wedge y < 1)\,.$$

d) If $\varphi(x) \rightleftharpoons x \cdot x > x$ then $\psi(a,b) \rightleftharpoons a > b \vee (a > 1 \wedge b > 1) \vee (a < 0 \wedge b < 0)$.
e) If $y \cdot z < x$ then $\psi(a,b,y,z) \rightleftharpoons y \cdot z < a \vee b < a$. Other atomic subcases can be considered by analogy.
Conjunction.
If $\varphi \rightleftharpoons \varphi_1 \wedge \varphi_2$ and ψ_1, ψ_2 are already constructed for φ_1, φ_2 then $\psi \rightleftharpoons \psi_1 \wedge \psi_2$.
Disjunction.
Suppose $\varphi \rightleftharpoons \varphi_1 \vee \varphi_2$ and ψ_1, ψ_2 are already constructed. Since $[a,b]$ is compact, validity of the formula $\forall x \in [a,b]\,(\varphi_1 \vee \varphi_2)$ is equivalent to existence of a finite family of open intervals $\{(\alpha_i,\beta_i)\}_{i=1,\ldots,r+s}$ such that $[a,b] \subseteq \bigcup_{i=1}^{r}(\alpha_i,\beta_i)$, for $i = 1,\ldots,r$ $\mathbb{R} \models \varphi_1$ and for $i = r+1,\ldots,s$ $\mathbb{R} \models \varphi_2$. Since φ_1 and φ_2 define open sets, this is equivalent to existence of a finite family of closed intervals $\{[\alpha_i',\beta_i']\}_{i=1,\ldots,r+s}$ such that $[a,b] \subseteq \bigcup_{i=1}^{r}[\alpha_i',\beta_i']$, for $i = 1,\ldots,r$ $\mathbb{R} \models \varphi_1$ and for $i = r+1,\ldots,s$ $\mathbb{R} \models \varphi_2$. It is represented by the following formula.

$$\bigvee_{r\in\omega}\bigvee_{r\in\omega} \exists\alpha_1'\ldots\exists\alpha_{s+1}'\exists\beta_1'\ldots\exists\beta_{s+1}'\left(\bigwedge_{i=1}^{r}\forall x\in[\alpha_i',\beta_i']\varphi_1 \wedge \bigwedge_{j=r+1}^{s}\forall x\in[\alpha_j',\beta_j']\varphi_2\right).$$

By induction hypothesis and Theorem 3, this formula is equivalent to a Σ-formula ψ.
Existential case.
Suppose $\varphi \rightleftharpoons \exists z\varphi_1(z,x_1,\ldots,x_n)$. As $[a,b]$ is compact and

$$\{\{x_1|\mathbb{R} \models \varphi_1(z,x_1,\ldots,x_n)\}\}_{z\in\mathbb{R}} = \{V_z\}_{z\in\mathbb{R}}$$

is an open cover, there exists a finite set $J = \{z_1,\ldots,z_s\} \subset \mathbb{R}$ such that $[a,b] \subseteq \bigcup_{z\in J} V_z$. So, validity of the formula $\forall x_1 \in [a,b]\exists z\varphi_1(z,x_1,\ldots,x_n)$ is equivalent to existence of the finite set $J = \{z_1,\ldots,z_s\}$ such that

$$\mathbb{R} \models \forall x_1 \in [a,b]\exists z\varphi_1(z,x_1,\ldots,x_n) \leftrightarrow \mathbb{R} \models \forall x_1 \in [a,b]\varphi^s(z_1,\ldots,z_s,x_1,\ldots,x_n),$$

where $\varphi^s(z_1,\ldots,z_s,x_1,\ldots,x_n) \rightleftharpoons \varphi_1(z_1,x_1,\ldots,x_n) \vee \cdots \vee \varphi_1(z_s,x_1,\ldots,x_n)$. By induction hypotheses, for every $J = \{z_1,\ldots,z_s\}$ there exists a Σ-formula $\psi^s(z_1,\ldots,z_s,a,b,x_2,\ldots,x_n)$ in the language $\sigma \cup \{P_\lambda'|\lambda : \{1,\ldots,n\} \to \{1,\ldots,n\}\}$ which is equivalent to $\forall x_1 \in [a,b]\varphi^s(z_1,\ldots,z_s,x_1,\ldots,x_n)$. Finally,

$$\begin{aligned}&\mathbb{R} \models \forall x_1 \in [a,b]\exists z\varphi_1(z,x_1,\ldots,x_n) \leftrightarrow\\ &\mathbf{HF}(\mathbb{R}) \models \textstyle\bigvee_{s\in\omega} \exists z_1\ldots\exists z_s\,(\psi^s(z_1,\ldots,z_s,a,b,x_2,\ldots,x_n))\,.\end{aligned}$$

A required Σ-formula ψ can be constructed using Theorem 3.

Now we are ready to construct a required formula ψ for a given Σ-formula φ. By Theorem 3, there exists an effective sequence of quantifier free formulas $\{\varphi_i\}_{i\in\omega}$ such that $\mathbf{HF}(\mathbb{R}) \models \varphi \leftrightarrow \mathbf{HF}(\mathbb{R}) \models \bigvee_{i\in\omega}\varphi_i$. As $[a,b]$ is compact and

$\{\{x_1|\mathbb{R} \models \varphi_i(x_1,\dots,x_n)\}\}_{i\in\omega} = \{U_i\}_{i\in\omega}$ is its cover, there exist $k \in \omega$ and a finite family $\{U_i\}_{i\le k}$ such that $[a,b] \subseteq \bigcup_{i\le k} U_i$. So,

$$\mathbb{R} \models \forall x_1 \in [a,b]\varphi(x_1,\dots,x_n) \leftrightarrow$$
$$\mathbf{HF}(\mathbb{R}) \models \bigvee_{k\in\omega} \forall x_1 \in [a,b] \bigvee_{i\le k} \varphi_i(x_1,\dots,x_n).$$

By induction hypotheses, for every $k \in \omega$ there exists a Σ–formula $\psi_k(a,b,\dots)$ which is equivalent to $\forall x_1 \in [a,b] \bigvee_{i\le k} \varphi_i(x_1,\dots,x_n)$. A required Σ-formula ψ can be constructed using Theorem 3.

Proof (Theorem 6).

We proceed by induction on the structure of the Σ_K-constraint φ.

Atomic Σ_K–constraint case. Suppose $\varphi(\bar{x}) \rightleftharpoons f_1(\bar{x}) < f_2(\bar{x})$, where $f_1(\bar{x}), f_2(\bar{x})$ are computable real-valued functions. It is easy to note that

$$\mathbb{R} \models \varphi(\bar{x}) \text{ iff } \mathbb{R} \models \exists a_1\exists a_2\exists b_1\exists b_2 \forall y_1 \in [a_1,b_1]\forall y_2 \in [a_2,b_2] \left(\bigwedge_{1\le i\le 2} (f_i(\bar{x}) < b_i \wedge f_i(\bar{x}) > a_i) \wedge (y_1 < y_2)\right).$$

By Proposition 2, f_i is computable if and only if $f_i(\bar{x}) < z$ and $f_j(\bar{x}) > z$ are Σ-definable. So, we can construct a required sequence of quantifier free formulas $\{\psi\}_{i\in\omega}$ using Proposition 4 and Theorem 3.

Conjunction, Disjunction and *Existential quantifier* cases are straightforward from Theorem 3.

Bounded Existential quantifier case. Suppose $\varphi(\bar{x}) \rightleftharpoons \exists y \in K\phi(y,\bar{x})$, where K is a computable compact subset of $\mathbb{R}^n$. Since ϕ defines effectively open set, the formula φ is equivalent to the formula

$$\exists y'\exists \epsilon > 0 \left(\phi(y',\bar{x}) \wedge d_K(y') < \epsilon \wedge \forall z \in \bar{B}(y',\epsilon)\phi(z,\bar{x})\right),$$

where $\bar{B}(y',\epsilon)$ is a closed ball. By properties of computable compact sets, the distance function d_K is computable [2], and, as a corollary, the set $\{(y',\epsilon)|d_K(y') < \epsilon\}$ is Σ-definable. By Proposition 4 and Theorem 3, there exists a required sequence of quantifier free formulas $\{\psi\}_{i\in\omega}$.

Bounded Universal quantifier case. Suppose $\varphi(\bar{x}) \rightleftharpoons \forall y \in K\phi(y,\bar{x})$, where K is a computable compact subset of $\mathbb{R}^n$. It is easy to see that φ is equivalent to the formula

$$\forall y \in [-q,q]^n \, (y \not\in K \vee \varphi(y,\bar{x}))$$

for some rational q which can be find effectively by K. By properties of computable closed sets, the distance function d_K is computable [2], and, as a corollary, $\{y|y \not\in K\} = \{y|d_K(y) > 0\}$ is Σ-definable. By Proposition 4 and Theorem 3, there exists a required sequence of quantifier free formulas $\{\psi\}_{i\in\omega}$.

Remark 3. It is worth noting that Theorem 6 provides an effective procedure which generates quantifier free formulas approximating the solution set of Σ_K–constraints.

4 Σ_K–constraints for Hybrid Systems

In this section we reconsider reachability problems in terms of Σ_K–constraints for a large class of hybrid systems, where continuous dynamics are represented by computable real-valued functions or functionals. In contrast to special types of hybrid systems such as timed automata or linear hybrid systems, for the considered class of hybrid systems difficulties arise from the fact that we can not exactly compute flow successors, but can only effectively approximate.

4.1 SHS-Specifications of Hybrid Systems

We consider the models of hybrid systems proposed by Nerode, Kohn in [21], called switched controlled systems. A hybrid system is a system which consists of a continuous plant that is disturbed by the external world and controlled by a program implemented on a sequential automaton. In the Nerode–Kohn model a hybrid system is represented by a continuous device given by a collection of dynamical systems parameterised by a control set along with a control automaton for switching among them.

The control automaton has input data (the set of sensor measurements) and the output data (the set of control laws).

The control automaton is modeled by three units. The first unit is a converter which converts each measurement into input symbols of the internal control automaton. The internal control automaton, in practice, is a finite state automaton with finite input and output alphabets. The second unit is the internal control automaton, which has a symbolic representation of a measurement as input and produces a symbolic representation of the next control law to be imposed on the plant as output. The third unit is a converter which converts these output symbols representing control laws into the actual control laws imposed on the plant. The plant interacts with the control automata at discrete times t_i, where the time sequence $\{t_i\}_{i\in\omega}$ satisfies realizability requirements. At time t_i the control automaton gets sensor data, computes the next control law, and imposes it on the plant. The plant will continue using this control law until the next interaction at time t_{i+1}.

The specification $\mathbf{SHS} = \langle TS, \mathbb{X}, \mathbb{U}, \mathbb{D}, Init, \mathbf{F}, Conv1, A, Conv2\rangle$ of a hybrid system consists of:

- $TS = \{t_i\}_{i\in\omega}$ is an effective sequence of rational numbers which encodes the times of communication of the external world, the plant and the control automata and satisfies realizability requirements.
- $\mathbb{X} \subseteq \mathbb{R}^n$ is a plant state space.
- $\mathbb{U} \subseteq \mathbb{R}^k$ is a set of control parameters.
- $\mathbb{D} \subseteq C(\mathbb{R})$ is a set of acceptable disturbances.
- $\mathbf{F} : \mathbb{D} \times \mathbb{U} \times \mathbb{X} \times \mathbb{R}^+ \to \mathbb{X}$ is a total computable function modeling the behaviour of the plant.
- $Conv1 : \mathbb{D} \times \mathbb{X} \to \omega$ is a weakly computable function. At the time of communication this function converts measurements, presented by $\mathbf{F}$, and the

representation of external world f into finite words which are input words of the internal control automata.
- $A : \omega \to \omega$ is a Σ-definable function. The internal control automata, in practice, is a finite state automata with finite input and finite output alphabets. So, it is naturally modeled by Σ-definable function which has a symbolic representation of measurements as input and produces a symbolic representation of the next control law as output.
- $Conv2 : \omega \to \mathbb{U}$ is a computable function. This function converts finite words representing control laws into control laws imposed on the plant.
- $Init = Init_{\mathbb{U}} \times Init_{\mathbb{X}}$ is a computable compact set of initial conditions.

Definition 4. *The behaviour of a hybrid system is defined by a function* $H : \mathbb{D} \times \mathbb{X} \times \mathbb{R}^+ \to \mathbb{X}$ *if for any external disturbance* $f \in \mathbb{D}$ *and initial states* $x \in Init_{\mathbb{X}}$ *the function* $H(f, x, \cdot) : \mathbb{R}^+ \to \mathbb{X}$ *defines the trajectory of the hybrid system.*

In order to investigate the behaviour of a hybrid system we consider the spaces $\mathbb{X}$, $\mathbb{U}$ and $\mathbb{D}$ and their products as structures in appropriate languages with induced τ_Σ topologies (see Subsection 2.6).

Theorem 7. *Suppose a hybrid system is specified as above. If the behaviour of the hybrid system is defined by a continuous function* $H : \mathbb{D} \times \mathbb{X} \times \mathbb{R}^+ \to \mathbb{X}$ *and there exists a computable function* $G : \mathbb{D} \times \mathbb{X} \times \mathbb{R}^+ \to \mathbb{R}^n$ *such that* $||H(f, x, t)|| \leq G((f, x, t))$ *for all* $f \in \mathbb{D}$, $x \in \mathbb{X}$ *and* $t \in \mathbb{R}^+$ *then* H *is computable.*

4.2 Σ_K-constraints and Reachability Problems

In this section we illustrate how Σ_K-constraints can be used for reasoning about hybrid systems. Suppose a hybrid system is formalised by

$$\mathbf{SHS} = \langle TS, \mathbb{X}, \mathbb{U}, \mathbb{D}, Init, \mathbf{F}, Conv1, A, Conv2 \rangle$$

which satisfies the conditions of Theorem 7.

Theorem 8. *The set of* Σ*-definable sets of* $\mathbb{X}$ *which are reachable by the hybrid system is computably enumerable.*

Proof. Let A be Σ-definable set. The reachability problem can be formalised as follows: $\psi \rightleftharpoons (\exists x \in Init_{\mathbb{X}})\, \exists f \exists t H(f, x, t) \in A$. Since the set of polynomials with rational coefficients is dense in $C(\mathbb{R})$ with the compact open topology, Theorem 6 and Theorem 7 imply the equivalence of ψ and a Σ-formula. So, for every Σ-definable set we can effectively check reachability.

Let $\mathcal{D} = \{f_i\}_{i \in \omega}$ be a computable family of acceptable computable disturbances.

Theorem 9. *The set*

$$\{< i, j > | A_i \text{ is reachable by the hybrid system under a disturbance } f_i, \text{ where } A_i \text{ is } \Sigma\text{-definable and } f_j \in \mathcal{D}\}$$

is computably enumerable.

Theorem 10. *The set*

$$\{< i, j > | K_i \text{ is unreachable by the hybrid system in bounded time under a disturbance } f_i, \text{ where } K_i \text{ is a co-semicomputable compact set and } f_j \in \mathcal{D}\}$$

is computably enumerable.

Proof. Let K be co-semicomputable compact set and time bounded by N and f a computable disturbance. The unreachability problem can be formalised as follows: $\varphi \rightleftharpoons \forall a \in Init_{\mathbb{X}} \forall t \in [0, N] H(f, a, t) \notin K$. By properties of co-semicomputable compact sets, the distance function d_K is lower semicomputable [2], and, as a corollary, $\{x | x \not\in K\} = \{x | d_K(x) > 0\}$ is Σ-definable. By Theorem 6 and Proposition 2, φ is equivalent to a Σ-formula. So, for every co-semicomputable compact set we can effectively check unreachability.

Now let us fix Σ-definable set A and co-semicomputable compact set K. Let I_r denote a subset of $Init_{\mathbb{X}}$ from which the set A is reachable and I_u denote a subset of $Init_{\mathbb{X}}$ from which the set K is unreachable in bounded time.

Theorem 11. *The sets I_r and I_u are Σ-definable.*

5 Conclusion

We present a methodology that enables the algorithmic analysis of Σ_K-constraints via translation to effective sequences of quantifier free formulas. We hope that proposed results and existing numerical constraint satisfaction techniques (e.g. [5,23]) will lead to new algorithms for effective continuous constraint solving.

References

1. Anai, H., Weispfenning, V.: Reach set computation using real quantifier elimination. In: Di Benedetto, M.D., Sangiovanni-Vincentelli, A.L. (eds.) HSCC 2001. LNCS, vol. 2034, p. 63. Springer, Heidelberg (2001)
2. Brattka, V., Weihrauch, K.: Computability on subsets of euclidean space I: Closed and compact sets. TCS 219, 65–93 (1999)
3. Barwise, J.: Admissible sets and Structures. Springer, Berlin (1975)
4. Basu, S., Pollack, R., Roy, M.-F.: Algorithms in Real Algebraic Geometry. Springer, Heidelberg (2003)
5. Benhamou, F., Goualard, F., Languénou, E., Christie, M.: Interval constraint solving for camera control and motion planning. ACM Trans. Comput. Log. 5(4), 732–767 (2004)
6. Caviness, B.F., Johnson, J.R. (eds.): Quantifier Elimination and Cylindrical Algebraic Decomposition. Springer, Wien (1998)
7. Collins, G.E.: Hauptvortrag: Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In: Brakhage, H. (ed.) GI-Fachtagung 1975. LNCS, vol. 33, pp. 134–183. Springer, Heidelberg (1975)
8. Ershov, Y.L.: Definability and computability. Plenum, New York (1996)

9. Henzinger, T.A., Rusu, V.: Reachability Verification for Hybrid Automata. In: Henzinger, T.A., Sastry, S.S. (eds.) HSCC 1998. LNCS, vol. 1386, pp. 190–205. Springer, Heidelberg (1998)
10. Immerman, N.: Descriptive Complexity. Springer, Heidelberg (1999)
11. Blass, A., Gurevich, Y.: Background, reserve and Gandy machines. In: Clote, P.G., Schwichtenberg, H. (eds.) CSL 2000. LNCS, vol. 1862, pp. 1–17. Springer, Heidelberg (2000)
12. Korovina, M.V., Kudinov, O.V.: Towards Computability over Effectively Enumerable Topological Spaces. Electr. Notes Theor. Comput. Sci. 202, 305–313 (2008)
13. Korovina, M.V., Kudinov, O.V.: Towards computability of higher type continuous data. In: Cooper, S.B., Löwe, B., Torenvliet, L. (eds.) CiE 2005. LNCS, vol. 3526, pp. 235–241. Springer, Heidelberg (2005)
14. Korovina, M.V.: Computational aspects of Σ-definability over the real numbers without the equality test. In: Baaz, M., Makowsky, J.A. (eds.) CSL 2003. LNCS, vol. 2803, pp. 330–344. Springer, Heidelberg (2003)
15. Korovina, M.V.: Gandy's theorem for abstract structures without the equality test. In: Vardi, M.Y., Voronkov, A. (eds.) LPAR 2003. LNCS, vol. 2850, pp. 290–301. Springer, Heidelberg (2003)
16. Korovina, M.V., Kudinov, O.V.: Semantic characterisations of second-order computability over the real numbers. In: Fribourg, L. (ed.) CSL 2001 and EACSL 2001. LNCS, vol. 2142, pp. 160–172. Springer, Heidelberg (2001)
17. Korovina, M.V., Kudinov, O.V.: Formalisation of Computability of Operators and Real-Valued Functionals via Domain Theory. In: Blank, J., Brattka, V., Hertling, P. (eds.) CCA 2000. LNCS, vol. 2064, pp. 146–168. Springer, Heidelberg (2001)
18. Korovina, M.V., Kudinov, O.V.: Generalised Computability and Applications to Hybrid Systems. In: Bjørner, D., Broy, M., Zamulin, A.V. (eds.) PSI 2001. LNCS, vol. 2244, pp. 494–499. Springer, Heidelberg (2001)
19. Korovina, M.V., Kudinov, O.V.: Characteristic properties of majorant-computability over the reals. In: Gottlob, G., Grandjean, E., Seyr, K. (eds.) CSL 1998. LNCS, vol. 1584, pp. 188–203. Springer, Heidelberg (1999)
20. Dahlhaus, E., Makowsky, J.A.: Query languages for hierarchic databases. Information and Computation 101, 1–32 (1992)
21. Nerode, A., Kohn, W.: Models for Hybrid Systems: Automata, Topologies, Controllability, Observability. In: Grossman, R.L., Ravn, A.P., Rischel, H., Nerode, A. (eds.) HS 1991 and HS 1992. LNCS, vol. 736, pp. 317–357. Springer, Heidelberg (1993)
22. Pour-El, M.B., Richards, J.I.: Computability in Analysis and Physics. Springer, Heidelberg (1988)
23. Ratschan, S., She, Z.: Constraints for Continuous Reachability in the Verification of Hybrid Systems. In: Calmet, J., Ida, T., Wang, D. (eds.) AISC 2006. LNCS (LNAI), vol. 4120, pp. 196–210. Springer, Heidelberg (2006)
24. Tarski, A.: A Decidion Method in Algebra and Geometry. University of California Press, Berkeley (1951)
25. Tiwari, A.: Abstractions for hybrid systems. Formal Methods in System Design 32(1), 57–83 (2008)
26. Weihrauch, K.: Computable Analysis. Springer, Berlin (2000)