Reach set computations using real quantifier elimination

Hirokazu Anai
Computer System Laboratories
Fujitsu Laboratories Ltd
Kawasaki, Kanagawa 211-8588, Japan
e-mail: anai@flab.fujitsu.co.jp

Volker Weispfenning
Fakultät für Mathematik und Informatik
Universität Passau
D-94030 Passau, Germany
e-mail: weispfen@uni-passau.de

MIP-0012
October, 2000

Abstract

Reach set computations are of fundamental importance in control theory. We consider the reach set problem for open-loop systems described by parametric inhomogeneous linear differential systems and use real quantifier elimination methods to get exact and approximate solutions. For simple elementary functions we give an exact calculation of the cases where exact semialgebraic transcendental implicitization is possible. For the negative cases we provide approximate alternating using discrete point checking or safe estimations of reach sets and control parameter sets. The method employs a reduction of forward and backward reach set and control parameter set problem to the transcendental implicitization problem for the components of special solutions of simpler non-parametric systems. Numerous examples are computed using the REDLOG and QEPCAD packages.

1 Introduction

Today integrated systems which combine physical process with information systems (*i.e.* digital programs) are great in demand. In fact complex systems which have been designed recently incorporate both differential equations to model the continuous behavior and discrete event systems to model instantaneous state changes in response to events. Systems of this kind appear in many ongoing research projects such as the development of automated highways, automated air-traffic controllers, chemical plants, analog-digital mixed circuits and multimedia protocols. Systems that are finite state machine with differential equation at each discrete state, in other words, combination of discrete event systems with differential equations are called *Hybrid Systems*.

Hence the analysis and design of such systems are very important problems. In order to analyze and design hybrid systems, their modeling formalisms should have both continuous and discrete models. A lot of research effort has been devoted to develop mathematical models, specification formalisms, analysis/design/control methods (and tools) to help control engineers in building such systems. (See [GNRR93, AKNS95, AHS96, AKNS97, HS98, Mal97, VvS99].)

For example, large scale concurrent systems like automated highway systems are *safety critical.* Most of the applications of hybrid systems are also safety critical. Safety is usually encoded as avoidance of an undesirable region of the state space. Consequently, among the most important problems for analyzing hybrid systems are verification problems; these are essentially reachability problems, that ask whether trajectories of the hybrid systems reach certain undesirable (unsafe) regions from an initial regions.

Computing reach set of hybrid systems is difficult because the hybrid systems have an infinite state space. Even for purely continuous systems, the reachability problem is known to be a difficult problem with a very few exceptions: Optimal control [TPS98] and game theoretic [Var97] approaches have been used to calculate reach sets for some systems. In case of purely discrete systems the study of reach set computation has come up to the stage of developing various computer-aided verification tools which perform either model checking or automated theorem proving.

The traditional approach to reachability problems attempts to compute the set of reachable states by successive approximation, starting from the set of initial states and repeatedly adding new reachable states. This computation can be automated and is guaranteed to converge for several special cases [KPSY93, AD94, ACH$^+$95, HKPV95, RR96], for which the reachability problem is decidable. However, in general, this approach, so-called *reachability construction*, may not be automatable or may not converge.

Due to the difficulty of computing the reach set for systems of differential equations, formal verification methods and tools for hybrid systems have been developed. For example, for hybrid automata which is one of several models representing hybrid systems [HNSY94, ACH$^+$95], model checking tools (see [DOTY96],[HHWT95]) have been developed based on the decidability results in [AD94, HKPV95] : The key tool for obtaining decidable classes of hybrid systems is the concept of *bisimulation*. Briefly speaking, bisimulations are simply reachability preserving quotient systems. If an infinite state hybrid system has a finite state bisimulation, then checking reachability can be equivalently achieved on the finite, discrete, quotient graph. Since quotient graph is finite, the algorithm terminates in finitely many steps. The first attempt to apply this approach was proposed in [AD94] for the model of timed automata. Then it was extended to multirate [ACH$^+$95] and rectangular [HKPV95, PV94] automata. However these methods and tools can deal with only very simple continuous models as, e.g. $\dot{x} = 1$, $A\dot{x} = b$. What is actually required is to handle hybrid systems with more complicated continuous parts.

Decidability of reachability problem for hybrid system with linear differential equation of the form $\dot{y} = Ay + Bu$ is discussed in [LPY99a, LPY99b]. This is a significant class of linear differential equations that is really widely used in linear control theory. These results are based on the notion of "o-minimality" [Dri98] from model theory and "quantifier elimination"[CJ98]. O-minimality is used to define a class of hybrid systems "o-minimal hybrid systems" and it is shown that all o-minimal hybrid systems admit finite bisimulations in [LPS00]. To make the bisimulation algorithm computationally feasible, they utilize mathematical logic, in particular real quantifier elimination, as main tool to represent and manipulate sets symbolically. Since quantifier elimination, in general, is possible for the theory of reals [Tar48], they have found subclasses of o-minimal hybrid systems that are definable in the theory.

Remark : There are many results that apply real quantifier elimination to control theory. The first attempt was proposed by Anderson *et.al.* [ABJ75] in 1975. They used Tarski's theory for the solution of the static output feedback problems. Since then, in particular recently, several interesting control system design and analysis problems can be reduced to quantifier elimination problems: testing stability of linear systems [HLS97], robust feedback control [DYA96], a robust multi-objective design for linear systems [DWC95], and trajectory tracking of nonlinear control systems [Jir97] and efficient robust control synthesis method [AH00]. Moreover, in [Neš98, Ana99] quantifier elimination is used for verification problems (reachability and observability problems) for discrete-time polynomial systems.

In this paper we study in particular reach set problems for continuous open-loop systems described by parametric systems of linear differential equations [KV97]. Roughly speaking the reach set problems is concerned with the relations between possible values of the state variables at some initial time t_0 and the corresponding values at later points in time. Some more specific problems are the following:

1. Fix a set M of values of the state variables at t_0; what are the possible corresponding values at later points t in time (up to some bound t_1 or ∞). (*Forward reach set*)

2. Fix a set N of "safe" values of the state variables. Find a set M as large as possible of initial values of the state variables at time t_0 that guarantees that the values of the state variables will for all later time points t (up to some bound t_1 or ∞) remain inside N. (*Backward reach set*)

3. Fix a set M of values of the state variables at t_0 and a set N of "safe" values of the state variables. Find a set P as large as possible of the control parameters such that all state variables with initial values at t_0 in M will have values in N for all later time points t (up to some bound t_1 or ∞.) (*Control parameter st*)

In this paper we study these problems for continuous parametric open-loop systems described by a system of parametric linear differential equations with arbitrary coefficients. Our main tool is the method of real quantifier elimination in computer algebra. This approach was introduced into reach set computations in [PY98]. In a series of papers they showed how to get exact solutions of the forward reach set problem for certain homogeneous linear differential systems of special type with constant coefficients [LPY99a] and for associated inhomogeneous systems with very special right hand side [LPY99b]. The exact solutions are always obtained as semialgebraic sets described by a boolean combination of polynomial inequalities.

Here we extend this ad hoc approach for special types of differential systems to a systematic study of the type of results obtainable by an approach via real quantifier elimination. By reducing the approach to its bare essentials, we obtain a much wider systematic framework applicable to a considerably larger class of systems. The main observation is that all the problems mentioned above can be reduced by exact symbolic algorithms to an implicitization problem for certain basic transcendental functions associated with the given system. Exact solutions for implicitization problems with rational parametrizations are well-known [BWK98, MC92]. Here we deal with the corresponding problem for transcendental parametrizations that has been studied only for very special cases e.g. in robotics [Cra86, Kov94].

Our main results are as follows: We associate with every parametric linear system of differential equations $\dot{y} = A(t)y + b(t, \underline{b})$ a finite system F of basic functions. Then for semialgebraic sets M, N all three problems can be solved exactly by real quantifier elimination relative to the implicitization problem for the components of the functions in F. Moreover the discrete point version of these problems require only finitely many evaluations of functions in F.

We prove a theorem that determines the exact classes of vector-valued functions of the kind arising in linear differential systems with constant coefficients, where exact semialgebraic implicitization is possible. As a corollary we obtain the exact limitations of the approach of [LPY99a, LPY99b] for linear differential systems with constant coefficients and special right hand sides.

We propose several ways to overcome these limitations by approximate computations: One way is to compute exact reach sets at a finite selection of discrete time points. This is always possible and practically quite efficient, but may lead to underestimation of the true reach set, depending on the selection of time points. Another approach separates the common time variable into different time variables in different components of the basic functions or different factors within the basic functions. This leads to an overestimation in the implicitization problem resulting in an overestimation of the forward reach set and an underestimation of the backward reach set and the control parameter set: So all three approximations are on the safe side.

We illustrate the different problems and solution methods by examples computed in the REDLOG-package of REDUCE [DS97] and QEPCAD [CH91].

Our results can be extended to the hybrid systems with linear continuous parts.

2 Reach sets and the transcendental implicitization problem

2.1 Problem statement

We consider parametric inhomogeneous systems S of linear differential systems of the form $\dot{y} = A(t)y + b(t)$ with an $n \times n$ matrix $A(t)$ of real continuous functions $a_{ij}(t)$ and a vector-valued real continuous function $b(t)$ defined on some interval I. The inhomogeneous part is assumed to be a linear combination $b(t) = \sum_{i=1}^{k} r_i g_i(t)$ with continuously differentiable functions $g_i : I \longrightarrow \mathbb{R}^n$, and real parameters r_i. Such a system can be viewed as an continuous open-loop control system with control parameters $r = (r_1, \ldots, r_k)$. Let M be some subset of $\mathbb{R}^n$ and fix an initial time point $t_0 \in I$: Then we denote the set of all solution functions $f : I \longrightarrow \mathbb{R}^n$ of the given system with parameters $r = (r_1, \ldots, r_k)$, by F_r, and the set of all solution functions $f \in F_r$ with initial value $f(t_0) \in M$ by $F = F_{M,r}$.

We consider the following *forward reach set problems*:

pointwise reach set Compute for a given time $t_1 \in I$ the set $\{f(t_1) \mid f \in F_{M,r}\}$.

discrete reach sets Compute for finitely many time points $t_1 < \ldots < t_k$ in I the union of the sets $\{f(t_i) \mid f \in F_{M,r}\}$.

bounded reach set Compute for a given time $t_1 > t_0$ in I the set $\{f(t) \mid f \in F_{M,r}, t_0 \leq t \leq t_1\}$.

unbounded reach set Suppose $I \supseteq [t_0, \infty)$, and compute the set $\{f(t) \mid f \in F_{M,r}, t_0 \leq t\}$.

unbounded reach set closure Suppose $I \supseteq [t_0, \infty)$, and compute the closure of the set $\{f(t) \mid f \in F_{M,r}, t_0 \leq t\}$.

All computations should be performed in explicit dependence on the control parameters r.

Of course any solution of the pointwise reach set problem yields a solution of the discrete reach sets problem by iteration. Any solution of the discrete reach sets problem yields an lower estimate for the sets to be computed in the bounded and unbounded reach set problems. Finally solution of the unbounded reach set closure problem yields a good upper estimate for the set to be computed in the unbounded reach set problems.

All these reach set problems are “forward-type” problems starting with a given set M of initial values. Of equal interest are the corresponding “backward” reach set problems that are a kind of “dual” to the corresponding “forward” problems. Notice that only the point set version is obtains by a simple reversion of the time variable; for the other problems this is not the case.

Some *backward reach set problems*: Let N be a subset of $\mathbb{R}^n$.

backward pointwise reach set Compute for a given time $t_1 \in I$ the set $\{f(t_0) \mid f(t_1) \in N\}$.

backward discrete reach sets Compute for finitely many time points $t_1 < \ldots < t_k$ in I the sets $\{f(t_0) \mid \ f(t_1), \ldots, f(t_k) \in N\}$.

backward bounded reach set Compute for a given time $t_1 > t_0$ in I the set $\{f(t_0) \mid f(t) \in N \text{ for all } t_0 \leq t \leq t_1\}$.

backward unbounded reach set Suppose $I \supseteq [t_0, \infty)$, and compute the set $\{f(t_0) \mid f(t) \in N \text{ for all } t_0 \leq t\}$.

From the viewpoint of control theory these problems have still other variants concerning the determination of suitable control parameter values $r = (r_1, \ldots, r_k)$. Let M as before be a subset of $\mathbb{R}^n$, and let N be another subset of $\mathbb{R}^n$. Then we have the following natural *control parameter set problems*:

pointwise control Compute for a given time $t_1 \in I$ the set $\{r \in \mathbb{R}^k \mid f(t_1) \in N \text{ for all } f \in F_{M,r}\}$.

discrete point control Compute for finitely many time points $t_1 < \ldots < t_k$ in I the set $\{r \in \mathbb{R}^k \mid f(t_i) \in N \text{ for all } f \in F_{M,r}, 1 \leq i \leq m\}$.

bounded interval control Compute for a given time $t_1 > t_0$ in I the set $\{r \in \mathbb{R}^k \mid f(t) \in N \text{ for all } f \in F_{M,r}, t_0 \leq t \leq t_1\}$.

unbounded interval control Suppose $I \supseteq [t_0, \infty)$, and compute the set $\{r \in \mathbb{R}^k \mid f(t) \in N \text{ for all } f \in F_{M,r}, t_0 \leq t\}$.

limit control Suppose $I \supseteq [t_0, \infty)$, and compute the set $\{r \in \mathbb{R}^k \mid \text{all accumulation points of } f(t) \text{ for } t \longrightarrow \infty \text{ are in } N \text{ for all } f \in F_{M,r}, t_0 \leq t\}$.

In order to make these problems mathematically precise, we need to specify the way in which the input sets M and N, and the output sets should be described. For an approach using symbolic computations it is natural to consider *semialgebraic* sets as possible inputs. These are subsets of $\mathbb{R}^n$ described by a boolean combination $\varphi(x_1, \ldots, x_n)$ of real polynomial inequalities. If in addition all the polynomials involved in $\varphi(x_1, \ldots, x_n)$ are linear, then the set described by φ is called *semilinear* [Dri98, Wei99].

Our goal is to solve the forward and backward reach set and control parameter set problems for semialgebraic input sets as far as possible with descriptions of semialgebraic sets as outputs. This, however, is not always possible. Hence we consider also the computation of overestimating the forward reach sets and underestimating the backward reach set and the control parameter sets by suitable semialgebraic sets.

Our main tool will be a reduction of reach set and control parameter set computations to corresponding implicitization problems for a fixed finite system of solutions of S, namely a fundamental system $f_1, \ldots, f_n$ for the homogeneous system S_0 associated with S, and special solutions $h_1, \ldots, h_k$ of the parameter-free inhomogeneous system S_i given by $\dot{y} = A(t)y + g_i(t)$. We refer to $\{f_1, \ldots, f_n, h_1, \ldots, h_k\}$ as a system of *basic functions* for S.

Implicitization problems for rational parametrizations of algebraic varieties have been widely considered in computer algebra [BWK98, MC92]. Here we have to study the corresponding problem for the vector-valued functions $f_1, \ldots, f_n, h_1, \ldots, h_k$, arising from the system S. As these functions will in general be transcendental, we refer to these problems as *transcendental implicitization problems.*

More precisely, we consider the following transcendental implicitization problems for given functions $f_i : I \longrightarrow \mathbb{R}^n$ for $1 \leq i \leq k$:

pointwise implicitization Compute for a given time $t_1 \in I$ the value $(f_1(t_1), \ldots f_k(t_1))$, regarded as point in $\mathbb{R}^{nk}$.

discrete points implicitization Compute for finitely many time points $t_1 < \ldots < t_k$ in I the values $(f_1(t_i), \ldots f_k(t_i))$, regarded as points in $\mathbb{R}^{nk}$.

bounded implicitization Compute for a given time $t_1 > t_0$ in I the set $\{(f_1(t), \ldots f_k(t)) \in \mathbb{R}^{nk} \mid t_0 \leq t \leq t_1\}$.

unbounded implicitization Suppose $I \supseteq [t_0, \infty)$, and compute the set $\{(f_1(t), \ldots f_k(t)) \in \mathbb{R}^{nk} \mid t_0 \leq t\}$.

overestimated unbounded implicitization Suppose $I \supseteq [t_0, \infty)$, and compute some semialgebraic or semilinear superset of the set $\{(f_1(t), \ldots f_k(t)) \in \mathbb{R}^{nk} \mid t_0 \leq t\}$

Notice that the unbounded and bounded implicitization problem for a single solution of the differential system S is in fact a special case of the unbounded and bounded reach set problem for S, respectively: Indeed, if f is fixed solution of S with initial value $f(t_0) = c \in \mathbb{R}^n$, put $M := \{c\}$. Then by the uniqueness property of the initial value problem, the set $F_{M,r} = \{f\}$. Hence in this case the unbounded/bounded reach set problem for S coincides with the unbounded/bounded implicitization problem for f.

2.2 Reduction to implicitization problems

Next we show that all reach set computations and control parameter set computations listed above can for semialgebraic input sets M, N be reduced in an exact symbolic way to one of these implicitization problems. All these reductions require real quantifier elimination as fundamental tool. For the case of discrete points forward and backward reach set and control parameter set and semilinear input sets M, N we find moreover that the output sets are also semilinear.

2.2.1 Forward reach set problems

Let $\varphi(x_1, \ldots, x_n)$ and $\psi(x_1, \ldots, x_n)$ be quantifier-free formulas describing the semialgebraic input sets M and N, respectively. Let $\dot{y} = Ay + b(t)$ with $b(t) = \sum r_i g_i(t)$ be a parametric linear system with control parameter r_i. Let f_i be a fundamental system of solutions of $\dot{y} = Ay$. Let h_i be a special solution of the system $\dot{y} = Ay + g_i(t)$. Then by the superposition principle, a special solution of the system S is given by $\sum_{i=1}^{k} r_i h_i$. Note that here r_i's may be regarded as constants or as free parameters.

Then the forward pointwise reach set problem can be described by the following formula:

$$\exists x_1 \ldots \exists x_n (\varphi(\sum_i x_i f_i + \sum_i r_i h_i)(t_0) \wedge \bigwedge_{j=1}^{n} y_j = (\sum_i x_i f_{ij} + \sum_i r_i h_{ij})(t_1)).$$

Hence real quantifier elimination provides an exact algorithmic solution to the problem. Similarly the forward discrete reach set problem can be described by the following formula and hence be solved by real quantifier elimination:

$$\exists x_1 \ldots \exists x_n (\varphi(\sum_i x_i f_i + \sum_i r_i h_i)(t_0) \wedge [\bigwedge_{j=1}^{n} y_j = (\sum_i x_i f_{ij} + \sum_i r_i h_{ij}(t_1))$$

$$\vee \ldots \vee \bigwedge_{j=1}^{n} y_j = (\sum_i x_i f_{ij} + \sum_i r_i h_{ij})(t_k)]).$$

Suppose we have a quantifier-free formula

$$\mu(y_{11}, \ldots, y_{1n}, \ldots, y_{n1}, \ldots, y_{nn}, z_{11}, \ldots, z_{1n}, \ldots, z_{k1}, \ldots, z_{kn})$$

describing the combined range of $(f_1, \ldots, f_n, h_1, \ldots, h_k)$ on the interval $[t_0, \infty)$ or $[t_0, t_1]$. So $\mu(y_{11}, \ldots, z_{kn})$ holds for $n(k+n)$-tuple in $\mathbb{R}^{n(k+n)}$ if and only if this tuple is in the combined range of $(f_1, \ldots, f_n, h_1, \ldots, h_k)$ on the given interval. Then the forward bounded and unbounded reach set problem, respectively, can be described by the following formula and hence solved by real quantifier elimination:a

$$\exists x_1 \dots \exists x_n [\varphi(\sum_i x_i f_i + \sum_i r_i h_i)(t_0) \wedge$$

$$\exists y_{11} \dots \exists y_{nn} \exists z_{11} \dots \exists z_{kn} (\mu(y_{11}, \dots, z_{kn}) \wedge \bigwedge_{j=1}^{n} y_j = (\sum_i x_i y_{ij} + \sum_i r_i z_{ij}))].$$

Next let $\mu'(y_{11}, \dots, y_{1n}, \dots, y_{n1}, \dots, y_{nn}, z_{11}, \dots, z_{1n}, \dots, z_{k1}, \dots, z_{kn})$ be a quantifier-free formula describing the set of all accumulation points of $(f_1(t), \dots, f_n(t), h_1(t), \dots, h_k(t))$ on $[t_0, \infty)$. So $\mu'(y_{11}, \dots, z_{kn})$ holds for $n(k+n)$-tuple in $\mathbb{R}^{n(k+n)}$ iff this tuple is in the set of all accumulation points of $(f_1, \dots, f_n, h_1, \dots, h_k)$ on the given interval. Then the forward unbounded reach set closure problem can be described by the following formula and hence solved by real quantifier elimination:

$$\exists x_1 \dots \exists x_n [\varphi(\sum_i x_i f_i + \sum_i r_i h_i)(t_0) \wedge \forall \epsilon (\epsilon > 0 \rightarrow$$

$$\exists y_1 \dots \exists y_{nn} \exists z_1 \dots \exists z_{kn} (\mu(y_{11}, \dots, z_{kn}) \wedge \bigwedge_{j=1}^{n} -\epsilon < y_j - (\sum_i x_i y_{ij} + \sum_i r_i z_{ij}) < \epsilon))]$$

2.2.2 Backward reach set problems

We use the same notations as above. Note that here again r_i's may be regarded as constants or as free parameters.

Then the backward pointwise reach set problem can be described by the following formula:

$$\exists x_1 \dots \exists x_n (\bigwedge_{j=1}^{n} y_j = (\sum_i x_i f_{ij} + \sum_i r_i h_{ij})(t_0) \wedge \psi(\sum_i x_i f_i + \sum_i r_i h_i)(t_1)).$$

Hence real quantifier elimination provides an exact algorithmic solution to the problem. Similarly the backward discrete reach set problem can be described by the following formula and hence be solved by real quantifier elimination:

$$\exists x_1 \dots \exists x_n (\bigwedge_{j=1}^{n} y_j = (\sum_i x_i f_{ij} + \sum_i r_i h_{ij})(t_0) \wedge$$

$$\psi(\sum_i x_i f_i + \sum_i r_i h_i)(t_1) \wedge \dots \wedge \psi(\sum_i x_i f_i + \sum_i r_i h_i)(t_k)).$$

Suppose we have a quantifier-free formula

$$\mu(y_{11},\ldots,y_{1n},\ldots,y_{n1},\ldots,y_{nn},z_{11},\ldots,z_{1n},\ldots,z_{k1},\ldots,z_{kn})$$

describing the combined range of $(f_1,\ldots,f_n,h_1,\ldots,h_k)$ on the interval $[t_0,\infty)$ or $[t_0,t_1]$, respectively. Then the backward bounded and unbounded reach set problem, respectively, can be described by the following formula and hence solved by real quantifier elimination:

$$\exists x_1\ldots\exists x_n[\bigwedge_{j=1}^{n} y_j = (\sum_i x_i f_{ij} + \sum_i r_i h_{ij})(t_0))\wedge$$

$$\forall y_{11}\ldots\forall y_{nn}\forall z_{11}\ldots\forall z_{kn}(\mu(y_{11},\ldots,z_{kn}) \rightarrow \psi(\sum_i x_i \underline{y_i} + \sum_i r_i \underline{z_i})(t))].$$

2.2.3 Control parameter set problems

We use the same notations as above. Note that here r_i's are regarded as free parameters.

Then the pointwise control problem can be described by the following formula:

$$\exists x_1\ldots\exists x_n(\varphi(\sum_i x_i f_i + \sum_i r_i h_i)(t_0)\wedge\psi(\sum_i x_i f_i + \sum_i r_i h_i)(t_1)).$$

Reduction of the discrete point control problem is immediately obtained from above as follows:

$$\exists x_1\ldots\exists x_n(\varphi(\sum_i x_i f_i + \sum_i r_i h_i)(t_0)\wedge\psi(\sum_i x_i f_i + \sum_i r_i h_i)(t_1)$$

$$\wedge\psi(\sum_i x_i f_i + \sum_i r_i h_i)(t_2)\wedge\ldots\wedge\psi(\sum_i x_i f_i + \sum_i r_i h_i)(t_k)).$$

Hence real quantifier elimination provides an exact algorithmic solution to the problem.

As in the previous section let $\mu(y_{11},\ldots,z_{kn})$ be a quantifier-free formula describing the combined range of $(f_1,\ldots,f_n,h_1,\ldots,h_k)$ on the interval $[t_0,\infty)$ or $[t_0,t_1]$. Then the bounded interval control problem and the unbounded interval control problem, respectively, can be described by the following formula and hence solved by real quantifier elimination:

$$\exists x_1\ldots\exists x_n[\varphi(\sum_i x_i f_i + \sum_i r_i h_i)(t_0)\wedge$$

$$\forall y_{11} \dots \forall y_{nn} \forall z_{11} \dots \forall z_{kn} (\mu(y_{11}, \dots, z_{kn}) \to \psi(\sum_i x_i \underline{y_i} + \sum_i r_i \underline{z_i})(t))].$$

Again as in the previous section let $\mu'(y_{11}, \dots, z_{kn})$ be a quantifier-free formula describing the set of all accumulation points of $(f_1(t), \dots, f_n(t), h_1(t), \dots, h_k(t))$ on $[t_0, \infty)$. Then the limit control problem can be described by the following formula and hence solved by real quantifier elimination:

$$\exists x_1 \dots \exists x_n [\varphi(\sum_i x_i f_i + \sum_i r_i h_i)(t_0) \wedge$$

$$\forall y_{11} \dots \forall y_{nn} \forall z_{11} \dots \forall z_{kn} (\mu'(y_{11}, \dots, z_{kn}) \to \psi(\sum_i x_i \underline{y_i} + \sum_i r_i \underline{z_i})(t))].$$

3 Exact transcendental implicitization

Here we consider cases, where the unbounded and bounded transcendental implicitization problem for given functions $f_i : I \longrightarrow \mathbb{R}^n$ $1 \leq i \leq k$: has an exact solution. Notice that the transcendental implicitization problem refers only to the component functions $f_{ij}(t)$ of $f_i(t)$; the grouping of these component functions into vector-valued functions is irrelevant here. So we may assume w.l.o.g. that $k = 1$ and that we deal with a single vector-valued function $f(t) := (f_1(t), \dots, f_n(t))$.

Then the exact transcendental implicitization problem is to determine the range of $f(t)$ on an unbounded interval $[t_0, \infty)$, or a compact interval $[t_0, t_1]$ contained in I. Since the f is continuous, this range is always a connected subset of $\mathbb{R}^n$.

In particular for $n = 1$ the range is a real interval J; moreover J is compact for the bounded implicitization case. In the unbounded implicitization case J is compact iff f is bounded on $[t_0, \infty)$, otherwise it is a closed semiinfinite interval or all of $\mathbb{R}$. In particular J is always a semialgebraic set that can computed explicitly from upper and lower bounds for f. In other words the unbounded and the bounded transcendental implicitization problem always has a positive solution for $n = 1$.

For $n = 2$ there are two well-known cases, where exact unbounded and bounded implicitization is possible, namely the $\sin - \cos$-pair and the $\sinh - \cosh$-pair: If f has components $f_1 := \cos(p(x)), f_2 := \sin(p(x))$, where $p(x)$ is a real polynomial of positive degree, then the range of $p(x)$ on $[t_0, \infty)$ includes an unbounded interval; consequently the range of f on $[t_0, \infty)$ is exactly the unit circle $\{(x_1, x_2) \mid x_1^2 + x_2^2 = 1\}$. On a bounded interval $[t_0, t_1]$, the range of $p(x)$ is again a compact interval, and so the range of f is a connected subset of the circle that can be easily computed as semialgebraic set from the range of $p(x)$.

For the hyperbolic case, where $f_1 := \cosh(p(x)), f_2 := \sinh(p(x))$, the situation is analogous, except that the role of the circle is replaces by the hyperbola $\{(x_1, x_2) \mid x_1^2 - x_2^2 = 1\}$.

The next theorem shows that exact transcendental implicitization is preserved under composition of functions in a very general sense:

Theorem 3.1 *Let $f(t) := (f_1(t), \ldots, f_k(t))$ be a vector valued function such that the range of f on every compact or upper unbounded closed interval I is a semialgebraic set described by a quantifier-free formula $\varphi_I(x_1, \ldots, x_k)$. Let g be a continuous real function defined on some compact or upper semiinfinite closed interval I'. Let h_i $(1 \leq i \leq n)$ be semialgebraic real functions defined on some subset of $\mathbb{R}^n$ extending the range of f. Let $\rho_i(x_1, \ldots, x_n, y)$ be quantifier-free formulas defining the graph $\{(x_1, \ldots, x_n, y) \mid y = h_i(x_1, \ldots, x_n)\}$ of h_i. Then the vector-valued function $f^*(t) := (f_1^*(t), \ldots, f_n^*(t))$ with components $f_i^*(t) := h_i(f_1(g(t)), \ldots, f_n(g(t)))$ for $1 \leq i \leq n$ has a semialgebraic range described by the formula $\psi(x_1, \ldots, x_n) :=$*

$$\exists y_1 \ldots \exists y_n (\varphi_J(y_1, \ldots, y_n) \wedge \bigwedge_{i=1}^{n} \rho_i(y_1, \ldots, y_n, x_i)),$$

where J is the range of $g(t)$ on I'.

The proof is obvious. Notice that the algorithmic quantifier elimination for the ordered field of real numbers this formula is required in order to transform the formula ψ into an equivalent quantifier-free formula that describes the range of f^* as a semialgebraic set. Typical instances of g and h_i are real polynomials or real rational functions.

The method can in particular be applied to the situation, where f consists of a $\sin - \cos$-pair or a $\sinh - \cosh$-pair as described above.

Other interesting examples are pairs $(\wp, \wp')$, where $\wp(t)$ is a Weierstrass $\wp$-function [All81, Kob84]. Then the range of $(\wp, \wp')$, on a large enough interval is a real elliptic curve

$$\{(x, y) \mid y^2 = 4x^3 - g_2 x - g_3\}$$

Further examples are functions $f : I \longrightarrow \mathbb{R}^n$ with components $f_i(t) := e^{p_i t / q_i}$ for integers p_i, q_i with $q_i \neq 0$. Let $g(t) := e^{t/q}$ for the least common multiple $q = q_i' q_i$. Then $f_i(t) = g(t)^{p_i q_i'}$. So the range of f can now be computed as semialgebraic set from the interval range of $g(t)$.

Example 3.2 *Here we show several concrete examples taken from [Kam77] of linear systems that have basic functions with components of type $e^{f(t)}$, $e^{f(t)}\sin(g(t))$, $e^{f(t)}\cos(g(t))$ and polynomials in t.*

1. *([Kam77] p609, No.8·11) Consider the inhomogeneous system $\dot{y} = Ay + b$ with*
$$A := \begin{pmatrix} 0 & -2 \\ 2 & 0 \end{pmatrix}, \quad b := \begin{pmatrix} 3t \\ 4 \end{pmatrix}.$$
Then all solutions are of the form
$$\begin{aligned} y_1 &= -\tfrac{5}{4} + C_1\cos(2t) - C_2\sin(2t), \\ y_2 &= \tfrac{3}{2}t + C_1\sin(2t) + C_2\cos(2t) \end{aligned}$$
for some real numbers C_1, C_2. So
$$\begin{pmatrix} \cos(2t) \\ \sin(2t) \end{pmatrix}, \begin{pmatrix} -\sin(2t) \\ \cos(2t) \end{pmatrix}$$
are fundamental solutions of the homogeneous system and
$$\begin{pmatrix} -\frac{5}{4} \\ \frac{3}{2}t \end{pmatrix}$$
is a special solution.

2. *([Kam77] p611, No.8·19) Consider the homogeneous system $\dot{y} = Ay$ with*
$$A := \begin{pmatrix} f(t) & g(t) \\ -g(t) & f(t) \end{pmatrix}$$
and continuous real functions f, g. Then all solutions are of the form
$$\begin{aligned} y_1 &= (C_1\cos G + C_2\sin G)F, \\ y_2 &= (-C_1\sin G + C_2\cos G)F \end{aligned}$$
for some real numbers C_1, C_2 and with
$$F = exp\int f(t)dt, \quad G = \int g(t)dt.$$
So
$$\begin{pmatrix} \cos(G(t)) \cdot F(t) \\ -\sin(G(t)) \cdot F(t) \end{pmatrix}, \begin{pmatrix} \sin(G(t)) \cdot F(t) \\ \cos(G(t)) \cdot F(t) \end{pmatrix}$$
are fundamental solutions.

3. *([Kam77] p618, No.8·49) Consider the homogeneous system* $\dot{y} = Ay$ *with*
$$A := \begin{pmatrix} 0 & c & -b \\ -c & 0 & a \\ b & -a & 0 \end{pmatrix}.$$
Then all solutions are of the form
$$\begin{aligned} y_1 &= aC_0 + rC_1\cos(rt) + (cC_2 - bC_3)\sin(rt), \\ y_2 &= bC_0 + rC_2\cos(rt) + (aC_3 - cC_1)\sin(rt), \\ y_3 &= cC_0 + rC_3\cos(rt) + (bC_1 - aC_2)\sin(rt) \end{aligned}$$
for some real numbers C_1, C_2, C_3 *which satisfy* $aC_1 + bC_2 + cC_3 = 0$ *and with* $r^2 = a^2 + b^2 + c^2$ *So*
$$\begin{pmatrix} a \\ b \\ c \end{pmatrix}, \begin{pmatrix} r\cos(rt) + \frac{ab}{c}\sin(rt) \\ -(c + \frac{a^2}{c})\sin(rt) \\ b\sin(rt) - \frac{ra}{c}\cos(rt) \end{pmatrix}, \begin{pmatrix} (c + \frac{b^2}{c})\sin(rt) \\ r\cos(rt) - \frac{ab}{c}\sin(rt) \\ -(a\sin(rt) + \frac{rb}{c}\cos(rt)) \end{pmatrix}$$
are fundamental solutions.

4. *([Kam77] p619, No.8·51) Consider the homogeneous system* $\dot{y} = Ay$ *with*
$$A := \begin{pmatrix} 1 & 1 & -1 \\ -1 & 1 & 1 \\ 1 & -1 & 1 \end{pmatrix}.$$
Then all solutions are of the form
$$\begin{aligned} y_1 &= e^t(C_0 + C_1\sqrt{3}\cos(t\sqrt{3}) + (C_2 - C_3)\sin(t\sqrt{3})), \\ y_2 &= e^t(C_0 + C_2\sqrt{3}\cos(t\sqrt{3}) + (C_3 - C_1)\sin(t\sqrt{3})), \\ y_1 &= e^t(C_0 + C_3\sqrt{3}\cos(t\sqrt{3}) + (C_1 - C_2)\sin(t\sqrt{3})) \end{aligned}$$
For some real number C_1, C_2, C_3 *which satisfy* $C_1 + C_2 + C_3 = 0$. *So*
$$\begin{pmatrix} e^t \\ e^t \\ e^t \end{pmatrix}, \begin{pmatrix} e^t(\sqrt{3}\cos(t\sqrt{3}) + \sin(t\sqrt{3})) \\ -2e^t\sin(t\sqrt{3}) \\ -e^t(\sqrt{3}\cos(t\sqrt{3}) - \sin(t\sqrt{3})) \end{pmatrix}, \begin{pmatrix} 2e^t\sin(t\sqrt{3}) \\ e^t(\sqrt{3}\cos(t\sqrt{3}) - \sin(t\sqrt{3})) \\ -e^t(\sqrt{3}\cos(t\sqrt{3}) + \sin(t\sqrt{3})) \end{pmatrix}$$
are fundamental solutions.

5. *([Kam77] p620, No.8·57) Consider the homogeneous system* $\dot{y} = Ay$ *with*
$$A := \begin{pmatrix} 0 & a & b\cos(ct) & b\sin(ct) \\ -a & 0 & b\sin(ct) & -b\cos(ct) \\ -b\cos(ct) & -b\sin(ct) & 0 & a \\ -b\sin(ct) & b\cos(ct) & -a & 0 \end{pmatrix}.$$

Then all solutions are of the form

$$\begin{aligned} y_1 &= C_1\cos(\alpha t) + C_2\sin(\alpha t) + C_3\cos(\beta t) + C_4\sin(\beta t),\\ y_2 &= C_1\sin(\alpha t) - C_2\cos(\alpha t) + C_3\sin(\beta t) - C_4\cos(\beta t),\\ y_3 &= \gamma C_1\sin(\beta t) + \gamma C_2\cos(\beta t) + \delta C_3\sin(\alpha t) + \delta C_4\cos(\alpha t),\\ y_4 &= -\gamma C_1\cos(\beta t) + \gamma C_2\sin(\beta t) - \delta C_3\cos(\alpha t) + \delta C_4\sin(\alpha t) \end{aligned}$$

for some real number C_1, C_2, C_3, C_4 *and with*

$$\alpha, \beta = \frac{1}{2}c \pm \frac{1}{2}\sqrt{(2a+c)^2 + 4b^2}, \quad \gamma = \frac{1}{b}(a+\alpha), \quad \delta = \frac{1}{b}(\alpha+\beta).$$

The fundamental solutions are the followings:

$$\begin{pmatrix} \cos(\alpha t) \\ \sin(\alpha t) \\ \gamma\sin(\beta t) \\ -\gamma\cos(\beta t) \end{pmatrix}, \begin{pmatrix} \sin(\alpha t) \\ -\cos(\alpha t) \\ \gamma\cos(\beta t) \\ \gamma\sin(\beta t) \end{pmatrix}, \begin{pmatrix} \cos(\beta t) \\ \sin(\beta t) \\ \gamma\sin(\alpha t) \\ -\gamma\cos(\alpha t) \end{pmatrix}, \begin{pmatrix} \sin(\beta t) \\ -\cos(\beta t) \\ \gamma\cos(\alpha t) \\ \gamma\sin(\alpha t) \end{pmatrix}.$$

Example 3.3 *Consider the inhomogeneous system* $\dot{y} = Ay + b$ *with*

$$A := \begin{pmatrix} 0 & \dot{p}(t) \\ -\dot{p}(t) & 0 \end{pmatrix}, \quad b := \begin{pmatrix} \dot{p}(t)\cos(p(t)) \\ \dot{p}(t)\sin(p(t)) \end{pmatrix}.$$

Note that A *has polynomial entries. Then all solutions are of the form*

$$\begin{aligned} y_1 &= (1+C_1)\sin(p(t)) + C_2\cos(p(t)),\\ y_2 &= -C_2\sin(p(t)) + C_1\cos(p(t)) \end{aligned}$$

for some real numbers C_1, C_2*. So*

$$\begin{pmatrix} \sin(p(t)) \\ \cos(p(t)) \end{pmatrix}, \begin{pmatrix} \cos(p(t)) \\ -\sin(p(t)) \end{pmatrix}$$

are fundamental solutions of the homogeneous system and

$$\begin{pmatrix} \sin(p(t)) \\ 0 \end{pmatrix}$$

is a special solution of the inhomogeneous system.

Next consider the parametric inhomogeneous system $\dot{y} = Ay + b$ *with the same* A *as above and*

$$b := r_1 \begin{pmatrix} \dot{p}(t)\cos(p(t)) \\ \dot{p}(t)\sin(p(t)) \end{pmatrix} + r_2 \begin{pmatrix} -\dot{p}(t)\sin(p(t)) \\ \dot{p}(t)\cos(p(t)) \end{pmatrix}.$$

Then all solutions are of the form

$$\begin{array}{rcl} y_1 & = & r_1 \sin(p(t)) + C_1 \sin(p(t)) + C_2 \cos(p(t)), \\ y_2 & = & r_2 \sin(p(t)) + C_1 \cos(p(t)) - C_2 \sin(p(t)) \end{array}$$

for some real numbers C_1, C_2, r_1, r_2. *So*

$$\begin{pmatrix} \sin(p(t)) \\ \cos(p(t)) \end{pmatrix}, \begin{pmatrix} \cos(p(t)) \\ -\sin(p(t)) \end{pmatrix}$$

are fundamental solutions of the homogeneous systems and

$$\begin{pmatrix} \sin(p(t)) \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \sin(p(t)) \end{pmatrix}$$

are further basic solutions, by superposing these solutions we have a special solution

$$\begin{pmatrix} r_1 \sin(p(t)) \\ r_2 \sin(p(t)) \end{pmatrix}.$$

Example 3.4 ([Bra83] p.586 example 3.13) *Consider the inhomogeneous system* $\dot{y} = Ay + b$ *with*

$$A := \begin{pmatrix} 1 & -3 \\ -2 & 2 \end{pmatrix}, \quad b := r_1 \begin{pmatrix} e^{2t} \\ 0 \end{pmatrix} + r_2 \begin{pmatrix} 0 \\ e^t \end{pmatrix}.$$

Then all solutions are of the form

$$\begin{array}{rcl} y_1 & = & C_1 e^{-t} - C_2 e^{4t} + \frac{1}{2} r_2 e^t \\ y_2 & = & \frac{2}{3} C_1 e^{-t} + C_2 e^{4t} + \frac{1}{3} r_1 e^{2t} \end{array}$$

for some real numbers C_1, C_2, r_1, r_2. *So*

$$\begin{pmatrix} e^{-t} \\ \frac{2}{3} e^{-t} \end{pmatrix}, \begin{pmatrix} -e^{4t} \\ e^{4t} \end{pmatrix},$$

are fundamental solutions of the homogeneous systems and

$$\begin{pmatrix} 0 \\ \frac{1}{3} e^{2t} \end{pmatrix}, \begin{pmatrix} \frac{1}{2} \\ 0 \end{pmatrix}$$

are special solutions of $\dot{y} = Ay + \begin{pmatrix} e^{2t} \\ 0 \end{pmatrix}$ *and* $\dot{y} = Ay + \begin{pmatrix} 0 \\ e^{2t} \end{pmatrix}$, *respectively. By superposing these solutions we have a special solution*

$$\begin{pmatrix} \frac{1}{2} r_2 e^t \\ \frac{1}{3} r_1 e^{2t} \end{pmatrix}.$$

4 Semialgebraic implicitization for simple elementary functions

In this section we characterize those cases of linear differential systems S with constant coefficients and "simple right hand side", where an exact implicitization of the system of basic functions for S is possible. The condition on the right hand side $b(t)$ of the system is as follows: All components $b_i(t)$ of $b(t)$ are $\mathbb{R}$-linear combinations of functions of the form $t^{d_i}e^{a_i t}\cos(\omega_i t)$, $t^{d_i}e^{a_i t}\sin(\omega_i t)$, where a_i, ω_i, α_i are real numbers. Then it is well known [Cro94] that a special solution of the inhomogeneous system and the fundamental solutions of the homogeneous system are again real linear combinations of functions of this kind. We call linear systems of this form *regular* and functions of type $t^d e^{at}\cos(\omega t)$, $t^d e^{at}\sin(\omega t)$, with a, ω, α real numbers *simple elementary functions.*

In some special cases of regular systems, it has been shown in [LPY99a, LPY99b, LPS00] how to solve the reach set problem by an implicit semialgebraic implicitization of functions of the following type:

- real polynomials $p_i(t)$
- exponential functions $e^{a_i t}$ with rational values of a_i
- trigonometric functions $\cos(\omega_i t)$, $\sin(\omega_i t)$, for rational ω_i.

In the following we show that for simple elementary functions there are only few more cases which allow unbounded exact semialgebraic implicitization; all these cases are covered by theorem 4.1 of the last section. In most of the remaining cases the exact semialgebraic implicitization problem is unsolvable. In fact we provide a complete characterization of those cases, where unbounded semialgebraic implicitization is possible.

Let $f(t) := (f_1(t), \ldots, f_n(t))$ with non-constant, pairwise different component functions $f_i(t) := t^{d_i}e^{a_i t}\cos(\omega_i t)$, or $f_i(t) := t^{d_i}e^{a_i t}\sin(\omega_i t)$, where d_i are non-negative integers and a_i, ω_i are real numbers. Moreover we assume that the functions f_i appear in $\cos - \sin$-pairs, whenever $\omega_i \neq 0$.

Theorem 4.1 *Let $f : [t_0, \infty) \longrightarrow \mathbb{R}^n$ be as above and let $n \geq 2$. Then the range of f is a semialgebraic set iff one of the following holds:*

1. *For all $1 \leq i \leq n$, $f_i(t) := t^{d_i}$.*
2. *For all $1 \leq i \leq n$, $d_i = 0$, $f_i(t) := e^{a_i t}$ and $dim_{\mathbb{Q}}(span(a_1, \ldots, a_n)) \leq 1$.*
3. *For all $1 \leq i \leq n$, $d_i \neq 0$, $a_i \neq 0$, $f_i(t) := t^{d_i}e^{a_i t}$, and $dim_{\mathbb{Q}}(span(a_1, \ldots, a_n)) \leq 1$, and $\frac{d_i}{d_1} = \frac{a_i}{a_1}$.*

4. *For all* $1 \le i \le n$, $f_i(t) := \cos(\omega_i t)$, *or* $f_i(t) := \sin(\omega_i t)$, *and* $dim_{\mathbb{Q}}(span(\omega_1, \ldots, \omega_n)) \le 1$.

Moreover in these positive cases a quantifier-free formula describing the range of f *can be computed algorithmically over the reals.*

Proof. In the cases mentioned above the unbounded semialgebraic implicitization is always achieved by the methods of the previous section, in particular theorem 4.1. Case 1 is obvious. In case 2 each $f_i(t)$ is a power of a single function $g(t) := e^{rp^{-1}t}$ for some real r and some positive integer p. In case 3 each $f_i(t)$ is a power of a single function $g(t) := t^d e^{rp^{-1}t}$ for some real r and positive integer d, p. In case 4 we can apply the addition theorem of cos and sin in order to expand functions of type $\cos(\omega t)$ or $\sin(\omega t)$ for integer ω into a real homogeneous sin-cos-polynomial of degree ω. Consequently each $f_i(t)$ is a homogeneous bivariate real polynomial in only two functions $g_1(t) := \cos(\alpha p^{-1}t)$ and $g_2(t) := \sin(\alpha p^{-1}t)$ for some real α and some positive integer p.

It remains to show that in all other cases the range of f is not a semialgebraic set. This requires a case distinction. In all remaining cases one of the conditions is violated already by two of the component functions of f, say by f_1, and f_2. Assume for a contradiction that the range of f, and hence the range of (f_1, f_2) on a semiinfinite interval $I = [t_0, \infty)$ is described by a quantifier-free formula $\varphi(x, y)$. We may assume without loss of generality that $t_0 = 0$.

Case 1: $f_i(t) := t^{d_i} e^{a_i t}$.

Subcase 1.1: $d_1 \neq 0$, $a_1 = 0$, $d_2 = 0$, $a_2 \neq 0$. Then the formula

$$\exists z (x \ge 0 \ \wedge \ x^{d_1} = z \ \wedge \ \varphi(z, y))$$

defines the graph $\{(x, y) \mid y = e^{a_2 x}\}$ of an exponential function on the non-negative reals. By its rate of growth, this is known to be a non-semialgebraic set ([BCR87], proposition 2.6.2), a contradiction. For other semiinfinite intervals I one argues similarly.

Subcase 1.2: $d_1 \neq 0$, $a_1 = 0$, $d_2 \neq 0$, $a_2 \neq 0$. Then again the formula

$$\exists z \exists u (x \ge 0 \ \wedge \ x^{d_1} = z \ \wedge \ x^{d_2} = u \ \wedge \ \varphi(z, uy))$$

defines the graph $\{(x, y) \mid y = e^{a_2 x}\}$, and so we get a contradiction as in subcase 1.1.

Subcase 1.3 $d_1 \neq 0$, $a_1 \neq 0$, $d_2 = 0$, $a_2 \neq 0$ and a_1, a_2 are linearly dependent over $\mathbb{Q}$, say $a_i = cb_i$ with real c and non-zero rational b_i. Then the formula

$$\exists z \exists u (x \geq 0 \ \wedge \ x^{d_1 b_2} = z \ \wedge \ u^{b_2} = z y^{b_1} \ \wedge \ \varphi(u, y))$$

defines the graph $\{(x, y) \mid y = e^{a_2 x}\}$ of an exponential function on the non-negative reals, and so we get a contradiction as in subcase 1.1.

Subcase 1.4: $d_1 \neq 0$, $a_1 \neq 0$, $d_2 \neq 0$, $a_2 \neq 0$ and a_1, a_2 are linearly dependent over $\mathbb{Q}$, say $a_i = cb_i$ with real c and non-zero rational b_i, and $d_2/d_1 \neq a_2/a_1$. Then the formula

$$\exists r \exists s \exists u \exists v (x \geq 0 \ \wedge \ x^{d_1 b_2} = r \ \wedge \ x^{d_2 b_1} = s \ \wedge$$

$$y^{b_1} s = v^{b_1} \ \wedge \ u^{b_2} s = v^{b_1} r \ \wedge \ \varphi(u, v))$$

defines the graph $\{(x, y) \mid y = e^{a_2 x}\}$ of an exponential function on the non-negative reals, and so we get a contradiction as in subcase 1.1.

Subcase 1.5: $a_1 \neq 0$ and $a_2 \neq 0$, and a_1, a_2 are linearly independent over $\mathbb{Q}$. Then by our assumption we can find a formula $\psi(x, y)$ such that $y = x^b$ iff $\psi(x, y)$ holds for all real x, y in some semiinfinite interval I' and some irrational constant b. Then by an analysis of this formula, one sees that for some sequence x_n going to infinity, all the values $y_n = x_n^b$ must satisfy a nontrivial polynomial equation with coefficients polynomials in x_n. Expanding this polynomial this would show that the sequence y_n has a rate of growth that coincides - up to some multiplicative constants - with the rate of growth of a rational power $z_n = x_n^q$ with q rational. This is a contradiction, since all powers x^c have different rate of growth for different values of c.

Case 2: $\omega_1 \neq 0, \omega_2 = 0$ and ($d_2 \neq 0$ or $a_2 \neq 0$).

Suppose to begin with that $a_2 \geq 0$. Then f_1 has infinitely many zeros on I and f_2 is ultimately strictly monotonically increasing with limit ∞. So the formula $\varphi(x, 0)$ describes an infinite semialgebraic subset of the real line with infinite complement, a contradiction, since every semialgebraic subset of the real line is finite or cofinite. In the case $a_2 < 0$ one argues similarly.

Case 3: $\omega_1, \omega_2 \neq 0$.

Subcase 3.1: ω_1, ω_2 are linearly dependent over $\mathbb{Q}$, and ($d_1 \neq 0$ or $d_2 \neq 0$, or $a_1 \neq 0$ or $a_2 \neq 0$.) Then by our hypothesis we may assume that f_1, f_2 form a $\cos - \sin$-pair with $\omega_1 = \omega_2, d_1 = d_2, a_1 = a_2$ Then we obtain that the function $f_1^2 + f_2^2 = t^{2d_1}e^{2a_1 t}$ is semialgebraic. Now we can argue as in case 2 that we can construct an infinite semialgebraic subset of the real line with infinite complement, a contradiction.

Subcase 3.2: ω_1, ω_2 are linearly independent over $\mathbb{Q}$. Then as in case 2, the formula $\varphi(x, 0)$ describes an infinite semialgebraic subset of the real line with infinite complement, a contradiction.

Remark: If $d_i = a_i = 0$ and $f_1 = \cos(\omega_1 t), f_2 = \sin(\omega_2 t)$, then the closure of the range of (f_1, f_2) is a simple semialgebraic set, namely the cube $-1 \leq x_1 \leq 1 \wedge -1 \leq x_2 \leq 1$. Hence for $\varphi(x_1.x_2) = (x_1 = 1 \wedge x_2 = 2)$ the unbounded reach set closure has a positive solution, namely this cube. $\square$

Example 4.2 *We review the examples in Examples 3.2 in view of the results of Theorem 3.1 and Theorem 4.1:*

1. *For this example, semialgebraic implicitization is not possible.*
2. *For this example, semialgebraic implicitization is possible for the case where $f = 0$ and g is a polynomial.*
3. *For this example, semialgebraic implicitization is possible by Theorem 3.1. In fact $g(t), f_i$ and h_i in Theorem 3.1 for this case are $g(t) = rt$, $f_1(t) = cos(t), f_2(t) = sin(t)$ and*

$$\begin{array}{rcl} h_1(x,y) & = & a \\ h_2(x,y) & = & b \\ h_3(x,y) & = & c \\ h_4(x,y) & = & rx + \frac{ab}{c}y \\ h_5(x,y) & = & -(c+\frac{a^2}{c})y \\ h_6(x,y) & = & by - \frac{ra}{c}x \\ h_7(x,y) & = & (c+\frac{b^2}{c})y \\ h_8(x,y) & = & rx - \frac{ab}{c}y \\ h_9(x,y) & = & -(ay + \frac{rb}{c}x). \end{array}$$

Here x, y correspond to $\cos(rt), \sin(rt)$, respectively.

4. *For this example, semialgebraic implicitization is not possible by Theorem 4.1.*

5. *For this example, by Theorem 3.1 and Theorem 4.1, semialgebraic implicitization is possible iff one of the followings holds*

$$(a)\ \frac{\beta}{\alpha} \in \mathbb{Q}, \quad (b)\ \alpha = 0, \beta \in \mathbb{Q}, \quad (c)\ \beta = 0, \alpha \in \mathbb{Q}.$$

There is no $g(t)$ in Theorem 3.1 for this case and h_i are as follows:

$$\begin{array}{rcl} h_1(x,y,u,v) & = & x \\ h_2(x,y,u,v) & = & y \\ h_3(x,y,u,v) & = & \gamma v \\ h_4(x,y,u,v) & = & -\gamma u \\ & \vdots & \\ h_{16}(x,y,u,v) & = & \gamma y. \end{array}$$

Here x, y, u, v correspond to $\cos(\alpha t), \sin(\alpha t), \cos(\beta t), \sin(\beta t)$, respectively.

This theorem clearly shows the limitations of the approach presented in [LPY99a, LPY99b]. In fact we have the following immediate corollary:

Corollary 4.3 *Let $\dot{y} = Ay$ with constant $n \times n$-matrix A be a homogeneous system of linear differential equations. Then exact semialgebraic implicitization is possible for a fundamental system of solutions of the system iff one of the following cases holds:*

1. *All eigenvalues of A are zero, i. e. A is a nilpotent matrix.*

2. *All eigenvalues* $\lambda_1, \ldots, \lambda_n$ *of* A *are non-zero, pairwise distinct reals, and* $dim_{\mathbb{Q}}(span(\lambda_1, \ldots, \lambda_n)) \leq 1$.

3. *All eigenvalues* $\lambda_1, \ldots, \lambda_n$ *of* A *are purely imaginary, say of the form* $\lambda_i = \mu_i\sqrt{-1}$ *with non-zero pairwise distinct reals* μ_i*, and* $dim_{\mathbb{Q}}(span(\mu_1, \ldots, \mu_n)) \leq 1$.

To illustrate this result we display two homogeneous linear differential systems with constant coefficients, where exact semialgebraic implicitization fails even for one specific solution. As a consequence the unbounded reach set is in general not semialgebraic for these systems.

Example 4.4 *1. Let* S *be the system* $\dot{y} = Ay$ *with*

$$A := \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

Since A *has* 1 *as only eigenvalue, the criteria of the corollary fail; in fact semialgebraic unbounded implicitization fails for the solution* $f(t) = (e^t, te^t)^T$.
2. Let S *be the system* $\dot{y} = Ay$ *with*

$$A := \begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \end{pmatrix}$$

Then A *has eigenvalues* $0, 0, i, -i$*, and so the criteria of the corollary fail; in fact semialgebraic unbounded implicitization fails for the solution* $f(t) = (\sin(t), \cos(t), t\sin(t), t\cos(t))^T$.

Nevertheless theorem 5.1 provides exact semialgebraic implicitization for a few new types of functions:

Example 4.5 *Unbounded exact semialgebraic implicitization is possible e.g. for the following pairs of functions:*

1. $(e^{\frac{2}{3}ct}, e^{\frac{7}{5}ct})$ *for an arbitrary non-zero real c.*

2. $(t^{10}e^{\frac{2}{3}ct}, e^{t^{21}\frac{7}{5}ct})$ *for an arbitrary non-zero real c.*

3. $(\cos(\frac{2}{3}ct), \sin(\frac{7}{5}ct))$ *for an arbitrary non-zero real c.*

5 Approximate solutions

In this section we study the cases, where an exact semialgebraic unbounded or bounded implicitization is definitely not possible. In these cases we want to find a semialgebraic superset of the true reach set and a semialgebraic subset of the true control parameter set, both if possible such that the set difference to the true reach set or control parameter set is in some sense "small enough."

For this purpose we revisit the reduction of these problems to the corresponding implicitization problem. Then an inspection of the reduction formulas shows that an overestimation of the implicitization problem leads to an overestimation of the forward reach set and an underestimation of the backward reach set and of the control parameter set.

Hence we are reduced to the problem of finding a superset of the true range of a transcendental vector valued function on a compact or upper semiinfinite closed interval.

One strategy to find overestimations of the range is *separation of variables:* It comes in two flavours: Separation of variables in different components, and separation of variables in products.

Separation of variables in different components : Let $f(t) = (f_1(t), \ldots, f_n(t))$ be defined on an interval I. Then separation of variables in different components yields the function $g(t) = (f_1(t_1), \ldots, f_n(t_n))$ defined on the cube I^n with $\text{range}(g) \supseteq \text{range}(f)$. The range of g is easily computed as a box $J_1 \times \cdots \times J_n$, where J_i is the range of f_i. Notice that this box is in fact the smallest box containing the range of f.

Example 5.1 *Let I be the upper semiinfinite interval* $[0, \infty)$.

1. *Let $f_1 := \cos(t)$, $f_2 := \sin(t)$. Then the true range of f is the unit circle. Separation of variables in different components yields as overestimation the closed unit square.*

2. *Let $f_1 := \cosh(t)$, $f_2 := \sinh(t)$. Then the true range of f is the hyperbola $\{(x, y) \mid x^2 - y^2 = 1\}$. Separation of variables in different components yields as overestimation the "quadrant" $\{(x, y) \mid x, y \geq 1\}$.*

3. *Let $f_1 := e^t \cos(t)$, $f_2 := e^t \sin(t)$. Then the true range of f is an expanding exponential spiral. Separation of variables in different components yields as overestimation the full plane $\mathbb{R}^2$.*

4. *Let $f_1 := e^{-t}\cos(t)$, $f_2 := e^{-t}\sin(t)$. Then the true range of f is a contracting exponential spiral. Separation of variables in different components yields as overestimation a closed box $\{(x,y) \mid -e^{\pi} \leq x \leq 1, -e^{3\pi/2} \leq y \leq e^{\pi/2}\}$.*

Separation of variables in products : Suppose the component functions of the given functions are products $f_i(t) := f_{i,1}(t)\cdots f_{i,m}(t)$, where each $f_{i,j}(t)$ is defined on the interval I. Put $g_j(t) := (f_{1,j},\ldots,f_{n,j})^T$. Then each g_j is also defined on the interval I. Let B_j be the range g_j, and put $C := B_1\cdots B_m$, where the multiplication is performed on the elements componentwise. Then C is obviously a superset of the range of f.

Example 5.2 *Let I be the upper semiinfinite interval $[0,\infty)$.*

1. *Let $f_1 := e^t\cos(t)$, $f_2 := e^t\sin(t)$. Then the true range of f is an expanding exponential spiral. Separation of variables in products yields as overestimation the annulus $\{(x,y) \mid x^2+y^2 \geq 1\}$. This is a better approximation as by separation of variables in different components.*

2. *Let $f_1 := e^{-t}\cos(t)$, $f_2 := e^{-t}\sin(t)$. Then the true range of f is a contracting exponential spiral. Separation of variables in products yields as overestimation the closed disk $\{(x,y) \mid x^2+y^2 \leq 1\}$. This approximation is incomparable to that by separation of variables in different components. Consequently the intersection of both is also an overestimation of the true range of f better than both.*

Of course both approaches can always be combined by taking intersection of both overestimations, as shown in the last example. Whenever the results of both single methods are incomparable with respect to inclusion, then the intersection of both yields an improvement of both overestimations. (See Section 7 for the computational examples of approximate solutions.)

6 Complexity

In this section we briefly discuss the complexity of our approach. From the results on complexity of quantifier elimination in [BPR96] we can give upper bounds for the asymptotic complexity of our approach. Reviewing the formulas used in the problem reductions in Section 2.2 we find the following:

Forward reach set : Pointwise reach set and discrete point reach set problems are described by purely existential formula. Hence the complexity of quantifier elimination is simple exponential in the dimension of the differential system. For fixed dimension it runs in a polynomial time. The complexity of bounded and unbounded reach set problems for a fixed number k of points is the same as for the pointwise reach set and discrete reach set problems. The complexity of the unbounded closure reach set problem is of type $e^{n^{O(1)}}$ (generalized singly exponential).

Backward reach set : The backward pointwise reach set and discrete reach set problems can be solved in singly exponential time. The complexity of backward bounded and unbounded reach set is of type $e^{n^{O(1)}}$.

Control parameter set: The upper complexity bounds for the control parameter set problems are same as for the corresponding backward reach set problems.

7 Computational examples in REDLOG and QEPCAD

In this section we report on experimental results in reach set and control parameter set computation. We use the reduction described in Section 2.2 to implicitization problems for a finite system of basic solutions.

In most cases we consider parametric linear differential systems with constant coefficients and inhomogeneous part consisting of a linear combination of basic functions of the form $v^t x^m e^{ax}\sin(\omega x)$ and $v^t x^m e^{ax}\cos(\omega x)$ for a constant vector v, non-negative integers m and reals a, ω. In order to show the generality of the approach for parametric inhomogeneous systems, we also consider an example with non-constant coefficients and special solutions given by unevaluated integrals.

The input sets M and N in all examples will be semilinear; as a result the outputs for pointwise and discrete point problems consist also of semilinear sets. All computations are performed in the REDLOG package [DS96, DS97] of REDUCE 3.7 and the QEPCAD [CH91]. The main algorithm employed is the linear and quadratic quantifier elimination [Wei88, LW93, Wei97] of REDLOG and quantifier elimination based on cylindrical algebraic decomposition [Col98, CH91] of QEPCAD. The (approximate) evaluation of the special solution functions by vectors with rational components uses an additional procedure "`evalnum`" that yields rational approximations of reals to any specified accuracy. This procedure was kindly provided to us by T. Sturm *. All the computations are executed on a SUN SPARC station Ultra I (140MHz).

Example 7.1 (Pointwise & discrete point reach set computation) *For computational example in the* REDLOG *package we first specify templates for the input of pointwise reach set computation. Note that in the following* t_0 =`u` *and* t_1 *and further points correspond to* `v`. *For the system with dimension 2 the input template looks as follows:*

```
% n : dimension of the system
% basic functions for n=2:
f01t:=  ;     % special solution
f02t:=  ;     % special solution
f11t:=  ;     % fundamental solution
f12t:=  ;     % fundamental solution
f21t:=  ;     % fundamental solution
f22t:=  ;     % fundamental solution

u:=    ;  v:=    ;       % starting point u , end point v

load evalf;                              % to use evalnum

f01u:= evalnum( sub({t=u},f01t));        % basic functions
f02u:= evalnum( sub({t=u},f02t));        % at start point u
f11u:= evalnum( sub({t=u},f11t));
f12u:= evalnum( sub({t=u},f12t));
f21u:= evalnum( sub({t=u},f21t));
f22u:= evalnum( sub({t=u},f22t));

f01v:= evalnum( sub({t=v},f01t));        % basic functions
f02v:= evalnum( sub({t=v},f02t));        % at end point v
f11v:= evalnum( sub({t=v},f11t));
f12v:= evalnum( sub({t=v},f12t));
f21v:= evalnum( sub({t=v},f21t));
f22v:= evalnum( sub({t=v},f22t));

% Constraint formula in variables y1,y2 describing the set M
phi:=   ;

% Parametric constraint formula
```

*e-mail: `sturm@uni-passau.de` ; `http://www.fmi.uni-passau.de/~sturm/`

```
phit :=  sub({y1 = x0*f01t + x1*f11t + x2*f21t,
              y2 = x0*f02t + x1*f12t + x2*f22t},phi);

% Constraint formula at point u
phiu :=  sub({y1 = x0*f01u + x1*f11u + x2*f21u,
              y2 = x0*f02u + x1*f12u + x2*f22u},phi);

% Condition at point v
gammav:= y1 = x0*f01v + x1*f11v + x2*f21v and
         y2 = x0*f02v + x1*f12v + x2*f22v;

% Main formula : ``psi'' gives reach sets!
psi:= ex({x0,x1,x2}, phiu and gammav);

% Parametric condition at point t
gammat:= y1 = x0*f01t + x1*f11t + x2*f21t and
         y2 = x0*f02t + x1*f12t + x2*f22t;

% Parametric main formula
psit:= ex({x0,x1,x2}, phiu and gammat);
```

Now we show concrete examples below. The following specific example is the homogeneous system

$$\dot{y} = \begin{pmatrix} 0 & 1 \\ -1 & -2 \end{pmatrix} y$$

and uses the template with the specific values of `u,v`*.*

```
f01t:= 0 ;                     % special solution
f02t:= 0 ;                     % special solution
f11t:= e^(-t) ;                % fundamental solution
f12t:= -e^(-t);                % fundamental solution
f21t:= t*e^(-t);               % fundamental solution
f22t:= (1-t)*e^(-t);           % fundamental solution

load evalf;

% Constraint formula in variables y1,y2
phi:= 0 <= y1 <= 1 and 1 <= y2 <= 3;

% Constraint formula at starting point u
```

```
u:= 1;

% basic functions at start point u = 1
f01u:= evalnum( sub({t=u},f01t));
f02u:= evalnum( sub({t=u},f02t));
f11u:= evalnum( sub({t=u},f11t));
f12u:= evalnum( sub({t=u},f12t));
f21u:= evalnum( sub({t=u},f21t));
f22u:= evalnum( sub({t=u},f22t));

phiu := sub({y1=x0*f01u + x1*f11u + x2*f21u,
             y2=x0*f02u + x1*f12u + x2*f22u},phi);

phiu := ( - 103548857136061*x1 - 103548857136061*x2 <= 0
         and 103548857136061*x1 + 103548857136061*x2
             - 281474976710656 <= 0)
         and (103548857136061*x1 + 281474976710656 <= 0
         and -103548857136061*x1 - 844424930131968 <= 0)
```

Then we compute the reach set at end point `v = 2` *as follows;*

```
v:= 2;

% basic functions at end point v = 2
f01v:= evalnum( sub({t=v},f01t));
f02v:= evalnum( sub({t=v},f02t));
f11v:= evalnum( sub({t=v},f11t));
f12v:= evalnum( sub({t=v},f12t));
f21v:= evalnum( sub({t=v},f21t));
f22v:= evalnum( sub({t=v},f22t));

gammav := y1 = x0*f01v + x1*f11v + x2*f21v and
          y2 = x0*f02v + x1*f12v + x2*f22v;

gammav := - 76186991394311*x1 - 152373982788622*x2
        + 562949953421312*y1 = 0    and
          76186991394311*x1 + 76186991394311*x2
        + 562949953421312*y2 = 0

gammav1 := gammav;

% Main formula
```

```
psi:= rlqe(ex({x0,x1,x2}, phiu and gammav));

psi := 207097714272122*y1 + 414195428544244*y2
       - 76186991394311 >= 0
       and 207097714272122*y1 + 414195428544244*y2
       - 228560974182933 <= 0
       and 207097714272122*y2 + 76186991394311 >= 0
       and y2 <= 0
```

Time: 50 ms

Then we compute the reach set at end point `v = 3` *as follows;*

```
v:=3;

% basic functions at end point v = 3
f01v:= evalnum( sub({t=v},f01t));
f02v:= evalnum( sub({t=v},f02t));
f11v:= evalnum( sub({t=v},f11t));
f12v:= evalnum( sub({t=v},f12t));
f21v:= evalnum( sub({t=v},f21t));
f22v:= evalnum( sub({t=v},f22t));

gammav := y1 = x0*f01v + x1*f11v + x2*f21v and
          y2 = x0*f02v + x1*f12v + x2*f22v;

gammav :=  - 112110511274691*x1 - 336331533824072*x2
         + 2251799813685248*y1 = 0  and
           112110511274691*x1 + 224221022549382*x2
         + 2251799813685248*y2 = 0

gammav2 := gammav;

% Main formula
psi:= rlqe(ex({x0,x1,x2}, phiu and gammav));

psi := 92871322523469874821861857208*y1
        + 139306983785204398037364241568*y2
        - 62843833691362528595729 35395 >= 0 and
       92871322523469874821861857208*y1
        + 139306983785204398037364241568*y2
        - 188531501074087585787188061 85 <= 0 and
```

```
        46435661261734937410930928604*y1
         + 92871322523469460626433312964*y2
         + 6284383369136252859572935395 >= 0 and
        112110511274691*y1 + 224221022549381*y2 <= 0
```

Time: 60 ms

Next we compute the discrete reach set at point `v = 2, 3` *with same starting point* `u = 1` *as follows:*

```
psi:= rlqe(ex({x0,x1,x2}, phiu and (gammav1 or gammav2)));
```

```
psi := (207097714272122*y1 + 414195428544244*y2 - 76186991394311 >= 0
   and 207097714272122*y1 + 414195428544244*y2 - 228560974182933 <= 0
   and 207097714272122*y2 + 76186991394311 >= 0 and y2 <= 0) or (
   92871322523469874821861857208*y1 + 139306983785204398037364241568*y2
   - 6284383369136252859572935395 >= 0 and 92871322523469874821861857208*y1
   + 139306983785204398037364241568*y2 - 18853150107408758578718806185 <= 0
   and 46435661261734937410930928604*y1 + 92871322523469460626433312964*y2
   + 6284383369136252859572935395 >= 0 and
   112110511274691*y1 + 224221022549381*y2 <= 0) or (
   63112607100126548709026791768*y1 + 69653491892602199018682120784*y2
   - 17082725115393302925039971491 >= 0 and 63112607100126548709026791768*y1
   + 69653491892602199018682120784*y2 - 51248175346179908775119914473 <= 0
   and 207097714272122*y2 + 76186991394311 >= 0 and y2 <= 0 and (
   17082725115393379112031365802*y1 + 27881066861650352990507007587*y2 = 0
   or 80526943027863*y1 - 112110511274690*y2 = 0))
```

Time: 180 ms

Example 7.2 (Inhomogeneous systems with polynomial coefficients) *Consider the inhomogeneous system $\dot{y} = Ay + b$ with*

$$A := \begin{pmatrix} 0 & 2t \\ -2t & 0 \end{pmatrix}, \quad b := r_1 \begin{pmatrix} 2t\cos(t^2) \\ 2t\sin(t^2) \end{pmatrix}$$

This system is an inhomogeneous system in Example 3.3 with $p(t) = t^2$ and $r_2 = 0$. Then all solutions are of the form

$$\begin{aligned} y_1 &= C_1\sin(t^2) + C_2\cos(t^2) + r_1\sin(t^2), \\ y_2 &= C_1\cos(t^2) - C_2\sin(t^2) \end{aligned}$$

for some real numbers C_1, C_2, r_1. So

$$\begin{pmatrix} \sin(t^2) \\ \cos(t^2) \end{pmatrix}, \begin{pmatrix} \cos(t^2) \\ -\sin(t^2) \end{pmatrix}$$

are fundamental solutions of the homogeneous systems $\dot{y} = Ay$ and

$$\begin{pmatrix} \sin(t^2) \\ 0 \end{pmatrix}$$

is a special solution for $r_1 = 1$. For this system we illustrate the computations in forward/backward unbounded reach set problems and the control parameter set problems below (Note that we set $t_0 = 0$):

- **Forward unbounded reach set:** *A quantifier-free formula $\mu(y_{11}, y_{12}, y_{21}, y_{22}, z_{11}, z_{12})$ ($\rightarrow$ `mu`) is obtained from the following first-order formula* `mufof` *by using quantifier elimination.*

```
mufof:= ex({u,v},(u^2+v^2=1 and y11=v and y12=u and y21=u
                  and y22=-v and z11=v and z12=0));

mu:=rlqe(mufof);

        2       2
mu := y11  + y12  - 1 = 0 and y11 + y22 = 0 and y11 - z11 = 0
      and y12 - y21 = 0  and z12 = 0
```

Time: 10 ms

Then we set $r_1 = 1$ and moreover $\varphi = 0 \leq x_1 \leq 1$ and $x_2 = 0$.

```
r1:=1;

freachaux:=ex({y11,y12,y21,y22,z11,z12},(mu and
               y1= x1*y11+x2*y21+r1*z11 and
              y2= x1*y12+x2*y22+r1*z12));

freachaux1:= rlqe(freachaux,{0<=x1,x1<=1,0<=x2,x2<=1});
```

Time: 470 ms plus GC time: 50 ms †

```
phi:= 0 <= x1 <= 1;  x2:=0;     % phi describing the set M
```

Then forward unbounded reach set problem is solved by using real quantifier elimination as follow:

†See Appendix for `freachaux1`.

```
freach:=   ex(x1,(phi and freachaux1));

ans := rlqepcad(freach);              %  call QEPCAD from REDLOG
               2        2
ans :=      y1  + 4*y2  - 4 <= 0
```

Time: 10 ms

• **Backward unbounded reach set:** `mu` *is the same formula as in forward unbounded reach set. We also set* $r_1 = 1$ *and* $\psi(x_1, x_2) = -\frac{1}{2} \leq x_1 \leq \frac{1}{2} \wedge -\frac{1}{2} \leq x_1 \leq \frac{1}{2}$.

```
r1:=1;

breachaux:=all({y11,y12,y21,y22,z11,z12},(mu impl
                ( -1/2 < x1*y11+x2*y21+r1*z11 < 1/2 and
                  -1/2 < x1*y12+x2*y22+r1*z12 < 1/2 )));

breachaux1:= rlsimpl(rlqe(breachaux));
```

Time: 19580 ms plus GC time: 1420 ms [‡].

Then the backward unbounded reach set problem is solved by using real quantifier elimination as follow:

```
breach:=  ex({x1,x2},(y1=x2 and y2=x1 and breachaux1));

ans := rlqe(breach);

ans := 2*y1 + 1 > 0 and 2*y1 - 1 < 0 and 2*y2 + 3 > 0 and 2*y2 + 1 < 0

          2                              4       2        2
 and (4*y1  - 1 < 0 or y1 = 0) and (16*y1  - 8*y1  - 16*y2  - 32*y2 - 15 > 0

      2     2                                     6        4   2        4
 or y1  + y2  + 2*y2 + 1 <> 0 or y2 + 1 = 0 or (16*y1  + 16*y1 *y2  + 32*y1 *y2

       4        2   2        2           2       2                        6
 + 8*y1  - 40*y1 *y2  - 80*y1 *y2 - 39*y1  + 9*y2  + 18*y2 + 9 > 0 and 16*y1

        4   2       4         3           3       2   2     2
 + 16*y1 *y2  - 8*y1  + 16*y1 *y2 + 16*y1  - 8*y1 *y2  + y1  - 4*y1*y2 - 4*y1
```

‡See Appendix for `breachaux1`

```
        4        3        2                         6        4   2       4
 - 16*y2  - 32*y2  - 11*y2  + 8*y2 + 4 > 0 and 16*y1  + 16*y1 *y2  - 8*y1

        3           3       2   2     2                         4        3
 - 16*y1 *y2 - 16*y1  - 8*y1 *y2  + y1  + 4*y1*y2 + 4*y1 - 16*y2  - 32*y2

        2
 - 11*y2  + 8*y2 + 4 > 0

         3          3                    2
 and 4*y1 *y2 + 4*y1  - y1*y2 - y1 + 2*y2  + 4*y2 + 2 > 0

         3          3                    2                        2
 and 4*y1 *y2 + 4*y1  - y1*y2 - y1 - 2*y2  - 4*y2 - 2 < 0 and 4*y1  - 3 < 0

                       2
 and y1 = 0)) and (4*y2  + 8*y2 + 3 < 0 or y2 + 1 = 0) and (

     4       2        2                       2     2
16*y1  - 8*y1  - 16*y2  - 32*y2 - 15 > 0 or y1  + y2  + 2*y2 + 1 <> 0

 or y2 + 1 = 0) and (

    2   2       2          2       4        3        2
4*y1 *y2  + 8*y1 *y2 + 5*y1  + 4*y2  + 16*y2  + 23*y2  + 14*y2 + 3 < 0

      2     2                        2       2
 or y1  + y2  + 2*y2 + 1 = 0 or (4*y1  + 4*y2  + 8*y2 + 3 < 0 and y1 <> 0))

          4       2   2     2     2            2       2
 and (4*y1  + 4*y1 *y2  - y1  + y2  < 0 or 4*y1  + 4*y2  - 1 < 0)
```

Time: 420 ms

- **Control parameter set:** *The formula* `mu` *is the same as in the reach set cases. We also set* $\varphi = 0 \leq x_1 \leq 1, x_2 = 0$ *and* $\psi(x_1, x_2) = -\frac{1}{2} < x_1 < \frac{1}{2} \wedge -\frac{1}{2} < x_1 < \frac{1}{2}$.

```
controlaux:=all({y11,y12,y21,y22,z11,z12},(mu impl
                  (-1/2 < x1*y11+x2*y21+r1*z11 < 1/2 and
                   -1/2 < x1*y12+x2*y22+r1*z12 < 1/2)));

controlaux1:= rlqe(controlaux);
```

Time: 42820 ms plus GC time: 5680 ms [§]

[§] See Appendix for `controlaux1`.

```
phi:= 0 <= x1 <= 1;  x2:=0;
```

Then control parameter set problem is solved by using real quantifier elimination as follow:

```
control:=ex(x1,(phi and controlaux1));

ans:=rlqe(control);
                                                         2
ans := (2*r1 + 1 > 0 and 2*r1 - 1 < 0 and (4*r1  - 1 < 0 or r1 = 0))


                                                2
 or (r1 + 1 >= 0 and r1 <= 0 and (4*r1  - 1 < 0 or r1 = 0))


                                             2
 or (r1 + 1 > 0 and r1 <= 0 and (r1  + r1 < 0 or r1 = 0))


                                                     2
 or (2*r1 + 1 >= 0 and 2*r1 - 1 < 0 and (r1  - r1 < 0 or r1 = 0))
```

Time: 70 ms

We can immediately reach the simplified result $-1 \leq r_1 < \frac{1}{2}$ *from the formula* `ans` *by hand calculation.*

Example 7.3 (Approximate solutions) *Consider the inhomogeneous system* $\dot{y} = Ay + b$ *with*

$$A := \begin{pmatrix} -1 & -1 \\ 1 & -1 \end{pmatrix}, \quad b := r_1 \begin{pmatrix} e^{-t}\cos(t) \\ -e^{-t}\sin(t) \end{pmatrix}.$$

Then all solutions are of the form

$$\begin{aligned} y_1 &= C_1 e^{-t}\cos(t) - C_2 e^{-t}\sin(t) + r_1 e^{-t}\sin(t), \\ y_2 &= C_1 e^{-t}\sin(t) + C_2 e^{-t}\cos(t) \end{aligned}$$

for some real numbers C_1, C_2, r_1*. So*

$$\begin{pmatrix} e^{-t}\cos(t) \\ e^{-t}\sin(t) \end{pmatrix}, \begin{pmatrix} -e^{-t}\sin(t) \\ e^{-t}\cos(t) \end{pmatrix}$$

are fundamental solutions of the homogeneous system $\dot{y} = Ay$ *and*

$$\begin{pmatrix} e^{-t}\sin(t) \\ 0 \end{pmatrix}$$

is a special solution of the inhomogeneous system with $r_1 = 1$*. For this system exact semialgebraic unbounded implicitization is not possible. Hence we solve the forward/backward unbounded reach set problems and control parameter set problem by approximation techniques shown in Section 5 (Note that we set* $t_0 = 0$*):*

□ **Separation of variables in different components** *First we use the box mentioned in Example 5.1 -4. for overestimation of the true range.*

• **Forward unbounded reach set :**

```
% for setting of the box
b1:= evalnum(-e^(pi));

          - 101773842498125
   b1 := --------------------
            4398046511104

b2:=evalnum(-e^((3*pi)/2));

          - 61197595913895
   b2 := -------------------
            549755813888

b3:=evalnum(e^(pi/2));

         14558270
   b3 := ----------
         3026367
```

A quantifier-free formula $\mu(y_{11}, y_{12}, y_{21}, y_{22}, z_{11}, z_{12})$ *(* $\rightarrow$ `mu` *) is obtained from the following first-order formula* `mufof` *by using quantifier elimination.*

```
mufof := ex({u,v},( b1 <= u <= 1 and b2 <= v <= b3 and
                    y11 = u and y12 = v and y21 = -v and y22 = u and
                    z11 = v and z12 = 0 ));

mu := rlqe(mufof);

  mu := 4398046511104*y11 + 101773842498125 >= 0 and y11 - y22 = 0
        and y11 - 1 <= 0 and 549755813888*y12 + 61197595913895 >= 0
        and 3026367*y12 - 14558270 <= 0 and y12 + y21 = 0
        and y12 - z11 = 0 and z12 = 0
```

Time: 30 ms

Then we set $r_1 = 1$ and moreover $\varphi = 0 \le x_1 \le 1$ and $x_2 = 0$.

```
r1:=1;
```

```
freachaux:=ex({y11,y12,y21,y22,z11,z12},(mu and
               y1= x1*y11+x2*y21+r1*z11 and
               y2= x1*y12+x2*y22+r1*z12));
```

```
freachaux1:= rlqe(freachaux,{0<=x1,x1<=1,0<=x2,x2<=1});
```

Time: 4450 ms plus GC time: 260 ms

```
phi:= 0 <= x1 <= 1; x2:=0;
```

Then forward unbounded reach set problem is overestimated by using real quantifier elimination as follow:

```
freach:=   ex(x1,(phi and freachaux1));
```

```
ans := rlsimpl(rlqe(freach));
```

Time: 4450 ms plus GC time: 260 ms ¶

• **Backward unbounded reach set :** `mu` *is same as forward case. Then we also set $r_1 = 1$ and moreover $\psi(x_1, x_2) = -\frac{1}{2} < x_1 < \frac{1}{2} \wedge -\frac{1}{2} < x_1 < \frac{1}{2}$.*

```
r1:=1;
```

```
breachaux:=all({y11,y12,y21,y22,z11,z12},(mu impl
               ( -1/2 < x1*y11+x2*y21+r1*z11 < 1/2 and
                 -1/2 < x1*y12+x2*y22+r1*z12 < 1/2 )));
```

```
breachaux1:= rlsimpl(rlqe(breachaux));
```

Time: 390 ms plus GC time: 60 ms

Then backward unbounded reach set problem is underestimated by using real quantifier elimination as follow:

¶The result **ans** has 309 atomic formulas in y_1, y_2. It is too large to show in this paper.

```
breach:=  ex({x1,x2},(y1=x2 and y2=x1 and breachaux1));

ans := rlqe(breach);
```

Time: 380 ms ‖.

• **Control parameter set :** `mu` *is same as reach set cases. Then we also set* $\varphi = 0 \leq x_1 \leq 1, x_2 = 0$ *and* $\psi(x_1, x_2) = -\frac{1}{2} < x_1 < \frac{1}{2} \wedge -\frac{1}{2} < x_1 < \frac{1}{2}$.

```
controlaux:=all({y11,y12,y21,y22,z11,z12},(mu impl
               (-1/2 < x1*y11+x2*y21+r1*z11 < 1/2 and
                -1/2 < x1*y12+x2*y22+r1*z12 < 1/2)));

controlaux1:= rlqe(controlaux);
```

Time: 380 ms plus GC time: 70 ms

```
phi:= 0 <= x1 <= 1 ; x2:=0;
```

Then control parameter set problem is underestimated by using real quantifier elimination as follow:

```
control:=ex(x1,(phi and controlaux1));

ans := rlsimpl(rlqepcad(control));

ans := (64974643610017061103443369555*r1 - 267664549879182870697738824 <= 0

        and 61197595913895*r1 + 274877906944 > 0) or (

        64974643610017061103443369555*r1 - 267664549879182870697738824 >= 0

        and 61197595913895*r1 - 274877906944 < 0)
```

Time: 40 ms

The last elimination step can be performed in REDLOG *(Time: 5560 ms plus GC time: 220 ms), but with a much longer output (125 atomic formulas).*

‖The result `ans` has 60 atomic formulas in y_1, y_2. See Appendix for `ans`

□ **Separation of variables in products** *Next we use alternatively the closed disc described in Example 5.2 -2. for overestimation of the true range.*

• **Forward unbounded reach set :** *A quantifier-free formula* $\mu(y_{11}, y_{12}, y_{21}, y_{22}, z_{11}, z_{12})$ *(* $\to$ `mu` *) is obtained from the following first-order formula* `mufof` *by using quantifier elimination.*

```
mufof := ex({u,v},(u^2 + v^2 <= 1 and
                   y11 = u and y12 = v and y21 = -v and y22 = u and
                   z11 = v and z12 = 0 ));


mu := rlqe(mufof);


          2      2
  mu := y11  + y12  - 1 <= 0 and y11 - y22 = 0 and y12 + y21 = 0 and
        y12 - z11 = 0 and z12 = 0
```

Time: 10 ms

Then we set $r_1 = 1$ *and moreover* $x_1 = 1, x_2 = 1$.

```
r1:=1;

x1=1; x2:=1;

freachaux:=ex({y11,y12,y21,y22,z11,z12},(mu and
               y1= x1*y11+x2*y21+r1*z11 and
               y2= x1*y12+x2*y22+r1*z12));
```

Time: 8590 ms plus GC time: 770 ms

Then forward unbounded reach set problem is overestimated by using real quantifier elimination as follow:

```
freachaux1:= rlqe(freachaux,{0<=x1,x1<=1,0<=x2,x2<=1});


                  2               2
freachaux1 := 2*y1  - 2*y1*y2 + y2  - 1 <= 0
```

Time: 20 ms

• **Backward unbounded reach set :** *The formula* μ *is the same as in the forward case. We also set* $r_1 = 1$ *and moreover* $\psi(x_1, x_2) = -\frac{1}{2} < x_1 < \frac{1}{2} \wedge -\frac{1}{2} < x_1 < \frac{1}{2}$.

```
r1:=1;

breachaux:=all({y11,y12,y21,y22,z11,z12},(mu impl
                ( -1/2 < x1*y11+x2*y21+r1*z11 < 1/2 and
                  -1/2 < x1*y12+x2*y22+r1*z12 < 1/2 )));

breachaux1:= rlsimpl(rlqe(breachaux));


                     2        2                         2     2
breachaux1 := 4*x1  + 4*x2  - 8*x2 + 3 < 0 and x1  + x2  - 2*x2 + 1 >= 0

and 2*x1 + 1 > 0 and 2*x1 - 1 < 0 and 2*x2 + 1 > 0 and 2*x2 - 1 < 0 and (

   2     2                                               2
x1  + x2  - 2*x2 + 1 > 0 or (x1 <= 0 and (x1 < 0 or 4*x2  - 8*x2 + 3 < 0)))

          2       2                           2     2
and (4*x1  + 4*x2  - 1 < 0 or x1 = 0) and (x1  + x2  - 2*x2 + 1 > 0 or x1 = 0)

       2     2                                                   2
and (x1  + x2  - 2*x2 + 1 > 0 or (x1 >= 0 and (x1 > 0 or 4*x2  - 8*x2 + 3 < 0)))

          2                              2
and (4*x2  - 1 < 0 or x2 = 0) and (4*x1  - 1 < 0 or x1 = 0) and (

     2         4         3         2                       2     2
16*x1  - 16*x2  + 64*x2  - 88*x2  + 48*x2 - 9 < 0 or x1  + x2  - 2*x2 + 1 > 0

                                                  2       2
or x1 = 0) and ((x1 = 0 and x2 = 0) or (4*x1  + 4*x2  - 1 < 0 and x1 <> 0)

         2   2      2       4     2
or (4*x1 *x2  + x1  + 4*x2  - x2  < 0 and (x1 <> 0 or x2 <> 0)))
```

Time: 14170 ms plus GC time: 1990 ms

Then backward unbounded reach set problem is underestimated by using real quantifier elimination as follow:

```
breach:=  ex({x1,x2},(y1=x2 and y2=x1 and breachaux1));

ans := rlsimpl(rlqe(breach));
```

```
           2              2               2            2
ans := 4*y1  - 8*y1 + 4*y2  + 3 < 0 and y1  - 2*y1 + y2  + 1 >= 0

and 2*y1 + 1 > 0 and 2*y1 - 1 < 0 and 2*y2 + 1 > 0 and 2*y2 - 1 < 0 and (

  2            2                              2
y1  - 2*y1 + y2  + 1 > 0 or (y2 <= 0 and (4*y1  - 8*y1 + 3 < 0 or y2 < 0)))

         2      2                          2           2
and (4*y1 + 4*y2 - 1 < 0 or y2 = 0) and (y1 - 2*y1 + y2 + 1 > 0 or y2 = 0)

       2           2                             2
and (y1 - 2*y1 + y2 + 1 > 0 or (y2 >= 0 and (4*y1 - 8*y1 + 3 < 0 or y2 > 0)))

         2                             2
and (4*y1  - 1 < 0 or y1 = 0) and (4*y2  - 1 < 0 or y2 = 0) and (

     4        3        2                 2              2            2
16*y1  - 64*y1  + 88*y1  - 48*y1 - 16*y2  + 9 > 0 or y1  - 2*y1 + y2  + 1 > 0

                                            2       2
or y2 = 0) and ((y1 = 0 and y2 = 0) or (4*y1  + 4*y2  - 1 < 0 and y2 <> 0)

        4       2   2     2     2
or (4*y1  + 4*y1 *y2  - y1  + y2  < 0 and (y1 <> 0 or y2 <> 0)))
```

Time: 110 ms

• **Control parameter set :** *The formula* mu *is the same as in the reach set cases. We also set* $\varphi = 0 \leq x_1 \leq 1$,$x_2 = 0$ *and* $\psi(x_1, x_2) = -\frac{1}{2} < x_1 < \frac{1}{2} \wedge -\frac{1}{2} < x_1 < \frac{1}{2}$.

```
controlaux:=all({y11,y12,y21,y22,z11,z12},(mu impl
                  (-1/2 < x1*y11+x2*y21+r1*z11 < 1/2 and
                   -1/2 < x1*y12+x2*y22+r1*z12 < 1/2)));

controlaux1:= rlqe(controlaux);
```

Time: 27120 ms plus GC time: 4050 ms

```
phi:= 0 <= x1 <= 1; x2:=0;
```

Then control parameter set problem is underestimated by using real quantifier elimination as follow:

```
control:=ex(x1,(phi and controlaux1));

ans:=rlqepcad(control);
                                 2
ans := (2*r1 + 1 > 0 and 4*r1  + 4*r1 - 1 <= 0)
                                 2
    or (2*r1 - 1 < 0 and 4*r1  - 4*r1 - 1 <= 0)
```

Time: 20 ms

8 Conclusions

In this paper we have studied forward and backward reach set and control parameter set problems for continuous parametric open-loop systems described by a system of parametric linear differential equations with arbitrary coefficients.

The approach using quantifier elimination was introduced into reach set computations in [PY98]. In contrast to their work we extend their ad hoc approach for special types of differential systems to a systematic study of the type of results obtainable by an approach via real quantifier elimination. Thus we obtain a much wider systematic framework applicable to a considerably larger class of systems. The main observation is that all the problems can be reduced by exact symbolic algorithms to an implicitization problem for certain basic transcendental functions associated with the given system.

We have proved a theorem that determines the exact classes of vector-valued functions of the kind arising in linear differential systems with constant coefficients, where exact semialgebraic implicitization is possible. As a corollary we have obtained the exact limitations of the approach of [LPY99a, LPY99b] for linear differential systems with constant coefficients. We have also proposed several ways to overcome these limitations by approximate computations. All the different problems and solution methods have been illustrated by examples computed in the REDLOG-package of REDUCE and QEPCAD.

Further research will be concerned with an extension of these results to hybrid systems.

References

[ABJ75] B. Anderson, N. Bose, and E. Jury. Output feedback stabilization and related problems — solution via decision methods. *IEEE Trans. Auto. Control*, 20(1):53–65, 1975.

[ACH+95] R. Alur, C. Coucoubetis, N. Halbwachs, T. Henzinger, P. Ho, X. Nicolin, A. Olivere, J. Safakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical Computer Science*, 138:3–34, 1995.

[AD94] R. Alur and D.L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.

[AH00] H. Anai and S. Hara. Fixed-structure robust controller synthesis based on sign definite condition by a special quantifier elimination. In *Proceedings of American Control Conference 2000*, 2000.

[AHS96] R. Alur, T.A. Henzinger, and E.D. Sontag, editors. *Hybrid Systems III*, volume 1066 of *LNCS*. Springer-Verlag, 1996.

[AKNS95] P. Antsaklis, W. Kohn, A. Nerode, and A. Sastry, editors. *Hybrid Systems II*, volume 999 of *LNCS*. Springer-Verlag, 1995.

[AKNS97] P. Antsaklis, W. Kohn, A. Nerode, and A. Sastry, editors. *Hybrid Systems IV*, volume 1273 of *LNCS*. Springer-Verlag, 1997.

[All81] Norman L. Alling. *Real elliptic curves.* North-Holland, 1981.

[Ana99] H. Anai. Algebraic approach to analysis of discrete-time polynomial systems. In *Proc. of European Control Conference 1999*, 1999.

[BCR87] J. Bochnak, M. Coste, and M.-F. Roy. *Géometrie algébrique réelle.* Springer, Berlin, Heidelberg, New York, 1987.

[BPR96] Saugata Basu, Richard Pollack, and Marie-Francoise Roy. On the combinatorial and algebraic complexity of quantifier elimination. *J. ACM*, 43(6):1002–1045, 1996.

[Bra83] Martin Braun. *Differential Equations and Their Applications*, volume 15 of *Applied Mathematical Science.* Springer, 3rd edition, 1983.

[BWK98] T. Becker, V. Weispfenning, and H. Kredel. *Gröbner Bases, a Computational Approach to Commutative Algebra*, volume 141 of *Graduate Texts in Mathematics*. Springer, New York, corrected second printing edition, 1998.

[CH91] George E. Collins and Hoon Hong. Partial cylindrical algebraic decomposition for quantifier elimination. *Journal of Symbolic Computation*, 12(3):299–328, September 1991.

[CJ98] B. Caviness and J. Johnson, editors. *Quantifier Elimination and Cylindrical Algebraic Decomposition*. Texts and Monographs in Symbolic Computation. Springer, Berlin, Heidelberg, New York, 1998.

[Col98] Georg E. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In B.F. Caviness and J.R. Johnson, editors, *Quantifier Elimination and Cylindrical Algebraic Decomposition*, Texts and Monographs in Symbolic Computation, pages 85–121. Springer, Wien, New York, 1998.

[Cra86] J.J Craig. *Introduction to Robotics*. Addison-Wesley, 1986.

[Cro94] Jane Cronin. *Differential Equations*. Marcel Dekker, Inc, 1994.

[DOTY96] C. Daws, A. Olivere, S. Tripakis, and S. Yovine. The tool KRONOS. In *Hybrid Systems III*, volume 1066 of *LNCS*, pages 208–219. Springer, 1996.

[Dri98] Lou van den Dries. *Tame Topology and o-minimal structures*. Cambridge University Press, 1998.

[DS96] Andreas Dolzmann and Thomas Sturm. Redlog user manual. Technical Report MIP-9616, FMI, Universität Passau, D-94030 Passau, Germany, October 1996. Edition 1.0 for Version 1.0.

[DS97] Andreas Dolzmann and Thomas Sturm. Redlog: Computer algebra meets computer logic. *ACM SIGSAM Bulletin*, 31(2):2–9, June 1997.

[DWC95] P. Dorato, W.Yang, and C.Abdallah. Application of quantifier elimination theory to robust multi-object feedback design. *J. Symb. Comp.* **11**, pages 1–6, 1995.

[DYA96] P. Dorato, Wei Yang, and C. Abdallah. Application of quantifier elimination to robust multi-objective feedback design. *Journal of Symbolic Computation*, 1996. To appear.

[GNRR93] R.L. Grossman, A. Nerode, A.P. Ravn, and H Rischel, editors. *Hybrid Systems*, volume 736 of *LNCS*. Springer-Verlag, 1993.

[HHWT95] T. Henzinger, P.H. Ho, and H. Wong-Toi. A user guide to HYTECH. In *Proceedings of TACAS 95: Tools and Algorithms for Construction and Analysis of Systems*, volume 1019 of *LNCS*, pages 41–71. Springer, 1995.

[HKPV95] T.A. Henzinger, P.W. Kopke, A. Puri, and P. Varaiya. What's decidable about hybrid automata? In *Proceedings of the 27th Annual Symposium on Theory of Computing, STOC'95*, pages 373–382. ACM Press, 1995.

[HLS97] Hoon Hong, Richard Liska, and Stanly Steinberg. Testing stability by quantifier elimination. *Journal of Symbolic Computation*, 24(2):161–187, August 1997. Special issue on applications of quantifier elimination.

[HNSY94] T. Henzinger, X. Nicillin, J. Sifakis, and S. Yovine. Symbolic model checking for real-time systems. *Information and Computation*, 111(2):193–244, 1994.

[HS98] T. Henzinger and S. Sastry, editors. *Hybrid Systems : Computation and Control*, volume 1386 of *LNCS*. Springer-Verlag, 1998.

[Jir97] Mats Jirstrand. Nonlinear control system design by quantifier elimination. *Journal of Symbolic Computation*, 24(2):137–152, August 1997. Special issue on applications of quantifier elimination.

[Kam77] Erich Kamke. *Differentialgleichungen - Lösungsmethiden und Lösungen*. B.G.Teubner, Stuttgart, 1977.

[Kob84] Neal Koblitz. *Introduction to elliptic curves and modular forms*. Springer, 1984.

[Kov94] Peter Kovács. Computer algebra in robot-kinematics. In *Proceeding of the workshop: "Computer Algebra in Science and Engineering", Bielefeld, Aug'94*, pages 303–316, Singapore, 1994. World Scientific.

[KPSY93] Y. Kesten, A. Pnueli, J. Sifakis, and S. Yovine. Integration graphs: a class of decidable hybrid systems. In *Proceedings of the 1st Workshop on Theory of Hybrid Systems*, volume 736 of *LNCS*, pages 179–208. Springer-Verlag, 1993.

[KV97] A. Kurzhanski and I. Vĺyi. *Ellipsoidal Calculus for Estimation and Control.* Birkhäuser, 1997.

[LPS00] G. Lafferriere, G.S. Pappas, and S. Sastry. O-minimal hybrid systems. *Mathematics of control, Signals and Systems*, 13(1):1–21, 2000.

[LPY99a] G. Lafferriere, G.S. Pappas, and S. Yovine. A new class of decidable hybrid systems. In *Hybrid Systems: Computation and Control*, volume 1569 of *LNCS*, pages 137–151. Springer, 1999.

[LPY99b] G. Lafferriere, G.S. Pappas, and S. Yovine. Reachability computation for linear hybrid systems. In *Proceedings of the 14th IFAC World Congress*, volume E, pages 7–12, Beijin, P.R.China, 1999.

[LW93] Rüdiger Loos and Volker Weispfenning. Applying linear quantifier elimination. *The Computer Journal*, 36(5):450–462, 1993. Special issue on computational quantifier elimination.

[Mal97] O. Maler, editor. *Hybrid and Real-Time Systems*, volume 1201 of *LNCS*. Springer-Verlag, 1997.

[MC92] D. Manocha and J.F. Canny. Algorithm for implicitizing rational parametric surfaces. In R.E. Barnhill and W. Boehm, editors, *Computer Aided Geometric Design*, volume 9. North-Holland, 1992.

[Neš98] D. Nešić. Two algorithms arising in analysis of polynomial models. In *Proceedings of 1998 American Control Conference*, pages 1889–1893, 1998.

[PV94] A. Puri and P. Varaiya. Decidability of hybrid systems with rectangular differential inclusions. In *Computer Aided Verification*, pages 95–104, 1994.

[PY98] G.J. Pappas and S. Yovine. Decidable hybrid systems. Technical report, University of California at Berkeley, 1998.

[RR96] O. Roux and V. Rusu. Uniformity for the decidability of hybrid automata. In *Proceedings of the 8th Conference on COmputer-Aided Verification, CAV'96*, volume 1145 of *LNCS*, pages 411–414. Springer-Verlag, 1996.

[Tar48] Alfred Tarski. A decision method for elementary algebra and geometry. Technical report, University of California, 1948. Second edn., rev. 1951.

[TPS98] G. Tomlin, G.J. Pappas, and S. Sastry. Conflict resolution for air traffic management: A study in mul ti-agent hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):509–521, 1998.

[Var97] P. Varaiya. Reach set computation using optimal control. *preprint*, 1997.

[VvS99] F. Vaandrager and J. von Schuppen, editors. *Hybrid Systems : Computation and Control*, volume 1569 of *LNCS*. Springer-Verlag, 1999.

[Wei88] Volker Weispfenning. The complexity of linear problems in fields. *Journal of Symbolic Computation*, 5(1–2):3–27, February–April 1988.

[Wei97] Volker Weispfenning. Quantifier elimination for real algebra—the quadratic case and beyond. *Applicable Algebra in Engineering Communication and Computing*, 8(2):85–101, February 1997.

[Wei99] Volker Weispfenning. Semilinear motion planning in REDLOG. Technical report, Universität Passau, D-94030 Passau, May 1999. available at the electronic proceedings of IMACS-ACA'99, http://math.unm.edu/aca.html.

9 Appendix

Some results for the computational example in Section 7.

Example 7.2

• Forward reach set problem :

```
freachaux1:= rlqe(freachaux,{0<=x1,x1<=1,0<=x2,x2<=1});
```

```
                  2            2               2          2
freachaux1 := (x1  + 2*x1 + x2  + 1 = 0 and x1  + x1 + x2  = 0

 and x1*y2 + x2*y1 + y2 = 0 and (x1 + y1 + 1 = 0 or x1 - y1 + 1 = 0)

                               4       3       2   2     2   2     2   2     2
 and (x2 = 0 or y1 = 0)) or (x1  + 2*x1  + 2*x1 *x2  - x1 *y1  - x1 *y2  + x1

          2          2     4     2   2     2   2                  2
 + 2*x1*x2  - 2*x1*y2  + x2  - x2 *y1  - x2 *y2  - 2*x2*y1*y2 - y2  = 0

       2          2             4       3     2   2     2   2       2          2
 and x1  + x1 + x2  <> 0) or (x1  + 3*x1  + x1 *x2  - x1 *y1  + 3*x1  + 2*x1*x2

                        2          2   2     2
 + 2*x1*x2*y1*y2 - x1*y1  + x1 + x2 *y1  + x2  + 2*x2*y1*y2 = 0

       2            2
 and x1  + 2*x1 + x2  + 1 = 0 and x2 > 0 and y1 <> 0) or (

  2            2     2                2            2                  5
x1  + 2*x1 + x2  - y1  + 1 >= 0 and x1  + 2*x1 + x2  + 1 <> 0 and ((x1 *y2

       4          3   2        3             3          2   2          2
 + 4*x1 *y2 + 2*x1 *x2 *y2 + x1 *x2*y1 + 6*x1 *y2 + 5*x1 *x2 *y2 + 2*x1 *x2*y1

       2           4           3             2                           4
 + 4*x1 *y2 + x1*x2 *y2 + x1*x2 *y1 + 4*x1*x2 *y2 + x1*x2*y1 + x1*y2 + x2 *y2

     3        2               4       3       2   2     2   2     2   2     2
 + x2 *y1 + x2 *y2 <= 0 and x1  + 2*x1  + 2*x1 *x2  - x1 *y1  - x1 *y2  + x1

          2          2     4     2   2     2   2                  2
 + 2*x1*x2  - 2*x1*y2  + x2  - x2 *y1  - x2 *y2  - 2*x2*y1*y2 - y2  = 0) or (

  5          4          3   2        3             3          2   2
x1 *y2 + 4*x1 *y2 + 2*x1 *x2 *y2 + x1 *x2*y1 + 6*x1 *y2 + 5*x1 *x2 *y2

       2             2           4           3             2
 + 2*x1 *x2*y1 + 4*x1 *y2 + x1*x2 *y2 + x1*x2 *y1 + 4*x1*x2 *y2 + x1*x2*y1

             4        3        2               4       3       2   2     2   2
 + x1*y2 + x2 *y2 + x2 *y1 + x2 *y2 >= 0 and x1  + 2*x1  + 2*x1 *x2  - x1 *y1

     2   2     2          2          2     4     2   2     2   2
 - x1 *y2  + x1  + 2*x1*x2  - 2*x1*y2  + x2  - x2 *y1  - x2 *y2  - 2*x2*y1*y2

     2
 - y2  = 0)))
```

• Backward reach set problem :

```
breachaux1:= rlqe(breachaux);
```

```
breachaux1 := 2*x1 + 3 > 0 and 2*x1 + 1 < 0 and 2*x2 + 1 > 0 and 2*x2 - 1 < 0

          2                              2                4       2
 and (4*x2  - 1 < 0 or x2 = 0) and (16*x1  + 32*x1 - 16*x2  + 8*x2  + 15 < 0

      2            2                                 4        3        2   4
 or x1  + 2*x1 + x2  + 1 <> 0 or x1 + 1 = 0 or (16*x1  + 32*x1  - 16*x1 *x2

       2   2        2           3                         6       4        3
 + 8*x1 *x2  + 11*x1  + 16*x1*x2  - 4*x1*x2 - 8*x1 - 16*x2  + 8*x2  + 16*x2

     2                         4        3        2   4       2   2        2
 - x2  - 4*x2 - 4 < 0 and 16*x1  + 32*x1  - 16*x1 *x2  + 8*x1 *x2  + 11*x1

           3                         6       4        3     2
 - 16*x1*x2  + 4*x1*x2 - 8*x1 - 16*x2  + 8*x2  - 16*x2  - x2  + 4*x2 - 4 < 0 and

      2   4        2   2       2           4           2                6
 16*x1 *x2  - 40*x1 *x2  + 9*x1  + 32*x1*x2  - 80*x1*x2  + 18*x1 + 16*x2

       4        2
 + 8*x2  - 39*x2  + 9 > 0

         2          3                      3
 and 2*x1  + 4*x1*x2  - x1*x2 + 4*x1 + 4*x2  - x2 + 2 > 0

         2          3                      3                      2
 and 2*x1  - 4*x1*x2  + x1*x2 + 4*x1 - 4*x2  + x2 + 2 > 0 and 4*x2  - 3 < 0

                       2
 and x2 = 0)) and (4*x1  + 8*x1 + 3 < 0 or x1 + 1 = 0) and (

     2                4       2               2            2
16*x1  + 32*x1 - 16*x2  + 8*x2  + 15 < 0 or x1  + 2*x1 + x2  + 1 <> 0

 or x1 + 1 = 0) and (

    4        3       2   2        2          2               2
4*x1  + 16*x1  + 4*x1 *x2  + 23*x1  + 8*x1*x2  + 14*x1 + 5*x2  + 3 < 0

      2            2                 2              2
 or x1  + 2*x1 + x2  + 1 = 0 or (4*x1  + 8*x1 + 4*x2  + 3 < 0 and x2 <> 0))

          2   2     2       4     2            2       2
 and (4*x1 *x2  + x1  + 4*x2  - x2  < 0 or 4*x1  + 4*x2  - 1 < 0)
```

• Control parameter set problem :

```
controlaux1:= rlqe(controlaux);

controlaux1 := 2*r1 + 2*x1 + 1 > 0 and 2*r1 + 2*x1 - 1 < 0 and 2*x2 + 1 > 0

                           2
 and 2*x2 - 1 < 0 and (4*x2  - 1 < 0 or x2 = 0) and (

     2                   2        4       2
16*r1  + 32*r1*x1 + 16*x1  - 16*x2  + 8*x2  - 1 < 0

      2               2     2                              2   4        2   2
 or r1  + 2*r1*x1 + x1  + x2  <> 0 or r1 + x1 = 0 or (16*r1 *x2  - 40*r1 *x2

       2              4              2                   2   4        2   2
 + 9*r1  + 32*r1*x1*x2  - 80*r1*x1*x2  + 18*r1*x1 + 16*x1 *x2  - 40*x1 *x2

       2        6       4     2             2                              2   2
 + 9*x1  + 16*x2  - 8*x2  + x2  > 0 and 4*x2  - 3 < 0 and x2 = 0 and (16*r1 *x1

       2           3                     3                  4        2   4
 - 4*r1  + 32*r1*x1  - 8*r1*x1 - 16*r1*x2  + 4*r1*x2 + 16*x1  - 16*x1 *x2

       2   2       2           3                  6       4     2
 + 8*x1 *x2  - 5*x1  - 16*x1*x2  + 4*x1*x2 - 16*x2  + 8*x2  - x2  < 0 or x1 > 0)

            2   2       2           3                     3                  4
 and ((16*r1 *x1  - 4*r1  + 32*r1*x1  - 8*r1*x1 + 16*r1*x2  - 4*r1*x2 + 16*x1

        2   4       2   2       2           3                  6       4     2
 - 16*x1 *x2  + 8*x1 *x2  - 5*x1  + 16*x1*x2  - 4*x1*x2 - 16*x2  + 8*x2  - x2

             2                    3               2          3
 < 0 and 2*r1  + 4*r1*x1 - 4*r1*x2  + r1*x2 + 2*x1  - 4*x1*x2  + x1*x2 > 0) or (

                  2   2       2           3                     3
x1 <= 0 and (16*r1 *x1  - 4*r1  + 32*r1*x1  - 8*r1*x1 + 16*r1*x2  - 4*r1*x2

        4        2   4       2   2       2           3                  6
 + 16*x1  - 16*x1 *x2  + 8*x1 *x2  - 5*x1  + 16*x1*x2  - 4*x1*x2 - 16*x2

       4     2
 + 8*x2  - x2  > 0

        2                    3               2          3
 or 2*r1  + 4*r1*x1 - 4*r1*x2  + r1*x2 + 2*x1  - 4*x1*x2  + x1*x2 > 0))) and (
```

```
     2   2       2           3                     3                  4
16*r1 *x1  - 4*r1  + 32*r1*x1 - 8*r1*x1 - 16*r1*x2  + 4*r1*x2 + 16*x1

        2   4       2   2       2           3                  6       4     2
 - 16*x1 *x2  + 8*x1 *x2  - 5*x1  - 16*x1*x2  + 4*x1*x2 - 16*x2  + 8*x2  - x2

            2                    3               2          3
 > 0 or 2*r1  + 4*r1*x1 + 4*r1*x2  - r1*x2 + 2*x1  + 4*x1*x2  - x1*x2 > 0) and (

     2   2       2           3                     3                  4
16*r1 *x1  - 4*r1  + 32*r1*x1 - 8*r1*x1 - 16*r1*x2  + 4*r1*x2 + 16*x1

        2   4       2   2       2           3                  6       4     2
 - 16*x1 *x2  + 8*x1 *x2  - 5*x1  - 16*x1*x2  + 4*x1*x2 - 16*x2  + 8*x2  - x2

                           2   2       2           3                     3
 < 0 or x1 < 0) and ((16*r1 *x1  - 4*r1  + 32*r1*x1 - 8*r1*x1 + 16*r1*x2

                  4        2   4       2   2       2           3
 - 4*r1*x2 + 16*x1  - 16*x1 *x2  + 8*x1 *x2  - 5*x1  + 16*x1*x2  - 4*x1*x2

        6       4     2
 - 16*x2  + 8*x2  - x2  < 0

         2                    3               2          3
 and 2*r1  + 4*r1*x1 - 4*r1*x2  + r1*x2 + 2*x1  - 4*x1*x2  + x1*x2 > 0) or (

                  2   2       2           3                     3
x1 >= 0 and (16*r1 *x1  - 4*r1  + 32*r1*x1 - 8*r1*x1 + 16*r1*x2  - 4*r1*x2

        4        2   4       2   2       2           3                  6
 + 16*x1  - 16*x1 *x2  + 8*x1 *x2  - 5*x1  + 16*x1*x2  - 4*x1*x2 - 16*x2

       4     2
 + 8*x2  - x2  > 0

        2                    3               2          3
 or 2*r1  + 4*r1*x1 - 4*r1*x2  + r1*x2 + 2*x1  - 4*x1*x2  + x1*x2 > 0)))))

          2                 2
 and (4*r1  + 8*r1*x1 + 4*x1  - 1 < 0 or r1 + x1 = 0) and (

     2                   2        4       2
16*r1  + 32*r1*x1 + 16*x1  - 16*x2  + 8*x2  - 1 < 0

      2               2     2                               4        3
 or r1  + 2*r1*x1 + x1  + x2  <> 0 or r1 + x1 = 0) and (4*r1  + 16*r1 *x1

        2   2       2   2     2           3             2                 4
```

```
 + 24*r1 *x1  + 4*r1 *x2  - r1  + 16*r1*x1  + 8*r1*x1*x2  - 2*r1*x1 + 4*x1

       2   2     2     2          2               2     2
 + 4*x1 *x2  - x1  + x2  < 0 or r1  + 2*r1*x1 + x1  + x2  = 0

         2                 2       2
 or (4*r1  + 8*r1*x1 + 4*x1  + 4*x2  - 1 < 0 and x2 <> 0)) and (

                            2       2
(x1 = 0 and x2 = 0) or (4*x1  + 4*x2  - 1 < 0 and x1 <> 0)

         2   2     2       4     2
 or (4*x1 *x2  + x1  + 4*x2  - x2  < 0 and (x1 <> 0 or x2 <> 0)))

Time: 42820 ms  plus GC time: 5680 ms

rlatnum(controlaux1);

44

rlvarl(controlaux1);

{{r1,x1,x2},{}}
```

Example 7.3

```
ans := 29116540*y1 - 6052734*y2 - 26090173 > 0

 and 29116540*y1 - 6052734*y2 - 32142907 < 0

 and 6052734*y1 + 29116540*y2 + 3026367 > 0

 and 6052734*y1 + 29116540*y2 - 3026367 < 0 and (

                        2
308004998399523061875*y1  - 64027948581210030080*y1*y2 + 6655051412835139584*y1

                  2
 < 0 or 6052734*y1  + 29116540*y1*y2 - 3026367*y1 < 0 or y1 = 0) and (

                        2
308004998399523061875*y1  - 64027948581210030080*y1*y2 - 6655051412835139584*y1

                  2
 < 0 or 6052734*y1  + 29116540*y1*y2 + 3026367*y1 < 0 or y1 = 0) and (
```

```
                                                     2
64027948581210030080*y1*y2 + 308004998399523061875*y2  - 57372897168374890496*y2

                                   2
 < 0 or 29116540*y1*y2 - 6052734*y2  - 26090173*y2 > 0 or y2 = 0) and (

                                                     2
64027948581210030080*y1*y2 + 308004998399523061875*y2  - 70682999994045169664*y2

                                   2
 < 0 or 29116540*y1*y2 - 6052734*y2  - 32142907*y2 > 0 or y2 = 0) and (

                                      2
549755813888*y1*y2 - 61197595913895*y2  - 274877906944*y2 > 0

                               2
 or 6052734*y1*y2 + 29116540*y2  - 3026367*y2 < 0 or y2 = 0) and (

                        2
308004998399523061875*y1  - 64027948581210030080*y1*y2 + 66550514128351395584*y1


                  2
 < 0 or 6052734*y1  + 29116540*y1*y2 - 3026367*y1 < 0 or y1 = 0 or y2 = 0) and (

                  2
101773842498125*y1  + 489580767311160*y1*y2 + 2199023255552*y1 < 0

                   2
 or 549755813888*y1  - 61197595913895*y1*y2 - 274877906944*y1 < 0 or y1 = 0

                                                       2
 or y2 = 0) and (549755813888*y1*y2 - 61197595913895*y2  + 274877906944*y2 > 0

                               2
 or 6052734*y1*y2 + 29116540*y2  + 3026367*y2 < 0 or y2 = 0) and (

                        2
308004998399523061875*y1  - 64027948581210030080*y1*y2 - 66550514128351395584*y1

                  2
 < 0 or 6052734*y1  + 29116540*y1*y2 + 3026367*y1 < 0 or y1 = 0 or y2 = 0) and (

                  2
101773842498125*y1  + 489580767311160*y1*y2 - 2199023255552*y1 < 0

                   2
 or 549755813888*y1  - 61197595913895*y1*y2 + 274877906944*y1 < 0 or y1 = 0
```

```
                                  2
 or y2 = 0) and (61197595913895*y1  + 549755813888*y1*y2 - 122670069734734*y1

 - 549755813888*y2 + 61472473820839 < 0

               2
 or 29116540*y1  - 6052734*y1*y2 - 55206713*y1 + 6052734*y2 + 26090173 < 0

 or y1 - 1 = 0) and (
                                                      2
64027948581210030080*y1*y2 + 308004998399523061875*y2  - 57372897168374890496*y2

                                   2
 < 0 or 29116540*y1*y2 - 6052734*y2  - 26090173*y2 > 0 or y1 - 1 = 0 or y2 = 0)

                                                2
and (489580767311160*y1*y2 - 101773842498125*y2  - 491779790566712*y2 > 0

                                           2
 or 61197595913895*y1*y2 + 549755813888*y2  - 61472473820839*y2 < 0

                                                   2
 or y1 - 1 = 0 or y2 = 0) and (61197595913895*y1  + 549755813888*y1*y2

 - 122120313920846*y1 - 549755813888*y2 + 60922718006951 < 0

               2
 or 29116540*y1  - 6052734*y1*y2 - 61259447*y1 + 6052734*y2 + 32142907 < 0

 or y1 - 1 = 0) and (
                                                      2
64027948581210030080*y1*y2 + 308004998399523061875*y2  - 70682999994045169664*y2

                                   2
 < 0 or 29116540*y1*y2 - 6052734*y2  - 32142907*y2 > 0 or y1 - 1 = 0 or y2 = 0)

                                                2
and (489580767311160*y1*y2 - 101773842498125*y2  - 487381744055608*y2 > 0

                                           2
 or 61197595913895*y1*y2 + 549755813888*y2  - 60922718006951*y2 < 0

 or y1 - 1 = 0 or y2 = 0)
```