Decidable and Semi-decidable Controller Synthesis for Classes of Discrete Time Hybrid Systems[1]

René Vidal Shawn Schaffert Omid Shakernia John Lygeros Shankar Sastry

Department of EECS, University of California, Berkeley CA 94720-1774, USA
{rvidal,sms,omids,lygeros,sastry}@eecs.berkeley.edu

Abstract

In this paper, we study classes of discrete time hybrid systems for which the classical algorithm for computing the maximal controlled invariant set and the least restrictive controller is computable and guaranteed to terminate in a finite number of iterations. We show how the algorithm can be encoded using quantifier elimination, which leads to a semi-decidability result for definable hybrid systems. For discrete time linear systems with linear constraints that are either controllable or nilpotent and have bounded disturbances, we show that the controlled invariance algorithm terminates in a number of iterations which is at most the dimension of the state space. Both in the hybrid and in the linear case, our results are much more general than the corresponding ones for continuous time systems. Finally we show that for linear systems with ellipsoidal constraints, an approximated solution can be obtained using robust convex programming. We provide an example showing that our algorithm gives better estimations than other ellipsoidal methods and is more efficient than the exact method for linear constraints.

1 Introduction

In this paper, we study classes of discrete time hybrid systems (DTHS) for which the the controlled invariance algorithm (CIA) is computable and guaranteed to terminate in a finite number of iterations.

The CIA for discrete time systems was proposed by Bertsekas and Rhodes [2]. They prove that, under certain conditions, the maximal controlled invariant set is a fixed point of a predecessor operator. However, the predecessor operator is not computable in general and the algorithm is not guaranteed to terminate.

Dórea and Hennet [7] consider a discrete time linear system with polyhedral state and disturbance constraints. They show that each iteration of the CIA is computable using linear programming and Fourier-Motzkin elimination. However, there is no analysis of the conditions under which the algorithm terminates. Shamma [22] proved that the CIA terminates in a *sufficiently large* number of iterations provided that a certain set is nonempty. However, the paper does not give any upper bound for the *sufficiently large* number of iterations. Also, given a system, there is no way to tell if the algorithm will terminate without fully computing each iteration of the CIA.

Blanchini [3] also studied the termination of the CIA for a modified version of the controlled invariance problem (CIP) that includes a "speed of convergence" parameter. When this parameter is less than one, the author gives conditions that guarantee termination in a *sufficiently large* but finite number of iterations. Unfortunately, this theorem is not valid for the case we consider here where the "speed of convergence" parameter is one.

The termination of the CIA has also been studied for continuous time hybrid systems. Many of the existing results have been motivated by the discovery of classes of linear systems for which the computation of reachable sets is decidable [11, 15]. For example, see [18] for decidable controller synthesis of continuous time linear systems and [19, 20] for semi-decidable controller synthesis for continuous time hybrid systems.

Even with guaranteed termination, the computational complexity of exactly solving the CIP through quantifier elimination may be doubly exponential [1]. Therefore, it is computationally attractive to consider ellipsoidal approximations to the problem [2, 9]. Indeed, ellipsoidal methods and convex programming [5, 8] have been successfully applied to many control problems, in particular to the computation of reachable sets [10].

Paper outline: Section 2 applies the concept of controlled invariance [2, 4, 21, 26] to a discrete time version of the hybrid automata in [13, 14]. Sections 3 and 4 study classes of DTHS for which the CIA is computable and guaranteed to terminate in a finite number of iterations. The proposed conditions for decidability are illustrated with some examples. Section 5 presents a polynomial time algorithm based on convex programming for solving the CIP. A numerical example compares our algorithm to the linear method in [7] and the ellipsoidal method in [2]. Section 6 concludes the paper.

[1] Research supported by grants DARPA F33615-98-C-3614, ARO MURI DAAH04-96-1-0341 and ONR N00014-97-1-0946. We thank the comments of L. El Ghaoui, G. Pappas and S. Simić.

2 Controlled Invariance of DTHS

2.1 Discrete Time Hybrid Systems

A **discrete time hybrid system** (DTHS) is a collection $H = (S, V, \text{Init}, \text{Inv}, r, R)$ consisting of:

- A finite collection of state variables, S, partitioned as $S = Q \cup X$, with $\mathbf{Q}$ being discrete and $\mathbf{X}$ being continuous. We use $\mathbf{Y}$ to denote the set of valuations of Y. We use $s = (q, x) \in \mathbf{S}$ to denote the state of the system.
- A finite collection of input variables, V, partitioned as $V = \Sigma \cup U \cup \Delta \cup D$. We use (σ, u) to denote the control inputs, with $\sigma \in \mathbf{\Sigma}$ and $u \in \mathbf{U}$ denoting the discrete and continuous control inputs, respectively. Similarly, we use (δ, d) to denote the disturbance inputs of the system, with $\delta \in \mathbf{\Delta}$ and $d \in \mathbf{D}$ denoting the discrete and continuous disturbance inputs, respectively.
- A set of initial states, $\text{Init} \subseteq \mathbf{S}$,
- An invariant set $\text{Inv} \subseteq \mathbf{S} \times \mathbf{V}$,
- A continuous reset relation, $r : \mathbf{S} \times \mathbf{V} \to 2^{\mathbf{X}}$ and
- A discrete reset relation $R : \mathbf{S} \times \mathbf{V} \to 2^{\mathbf{S}}$.

A sequence $\chi = (s, v) \in (\mathbf{S} \times \mathbf{V})^*$ is called an **execution** of the DTHS H if $s[0] \in \text{Init}$, and for all $k \geq 0$,

- $s[k+1] \in R(s[k], v[k])$, or
- $(s[k], v[k]) \in \text{Inv}$, $q[k+1] = q[k]$ and $x[k+1] \in r(s[k], v[k])$.

Like their continuous time counterparts, DTHS can be thought of as directed graphs, with nodes $q \in \mathbf{Q}$ and edges (q, q') for all $q, q' \in \mathbf{Q}$ such that

$$\exists x, x' \in \mathbf{X}, v \in \mathbf{V} \text{ with } (q', x') \in R(q, x, v).$$

With each node of the graph, $q \in \mathbf{Q}$, we associate a set of initial conditions, an invariance relation and a transition relation given by

$$\begin{aligned} \text{Init}_q &= \{x \in \mathbf{X} \mid (q, x) \in \text{Init}\}, \\ \text{Inv}_q(v) &= \{x \in \mathbf{X} \mid (q, x, v) \in \text{Inv}\}, \\ r_q(x, v) &= \{x' \in \mathbf{X} \mid x' \in r(q, x, v)\}. \end{aligned}$$

With each edge, (q, q') of the graph we associate a guard relation and a reset relation given by:

$$\begin{aligned} G_{qq'}(v) &= \{x \in \mathbf{X} \mid \exists x' \in \mathbf{X}, (q', x') \in R(q, x, v)\}, \\ R_{qq'}(x, v) &= \{x' \in \mathbf{X} \mid (q', x') \in R(q, x, v)\}. \end{aligned}$$

For pairs (q, q') which are not edges, we can set $G_{qq'}(v) = R_{qq'}(x, v) = \emptyset$ for all $x \in \mathbf{X}$ and $v \in \mathbf{V}$.

2.2 Controlled Invariance of DTHS

A (memoryless) *controller*, g, is a map $g : \mathbf{S} \to 2^{\mathbf{\Sigma} \times \mathbf{U}}$. A controller is called *non-blocking* if $g(s) \neq \emptyset$ for all $s \in \mathbf{S}$. We say that a controller g *solves the problem* $(H, \Box F)$, if and only if, g is non-blocking and the closed loop causal executions stay in F forever. If such a controller exists we say that $(H, \Box F)$ *can be solved.*

A set $W \subseteq \mathbf{S}$ is called a **controlled invariant set** of H if $(H', \Box W)$ can be solved, where H' is the same as H, but with $\text{Init}' = W$. We say that the controller that solves $(H', \Box W)$ *renders the set W invariant.* Also, given a set $F \subseteq \mathbf{S}$, a set $W \subseteq F$ is called a *maximal controlled invariant subset of F*, if it is controlled invariant and it is not a proper subset of any other controlled invariant subset of F.

Many controllers may be able to solve a particular problem. We would like to find a controller that imposes less restrictions on the inputs it allows. A controller g that solves $(H, \Box F)$ is called a **least restrictive controller** if it is maximal among the controllers that solve $(H, \Box F)$ in the partial order defined by $\preceq$, where $g_1 \preceq g_2$ if for all $s \in \mathbf{S}$, $g_1(s) \subseteq g_2(s)$.

Notice that, the problem $(H, \Box F)$ can be solved if and only if there exists a unique maximal controlled invariant set $\hat{W}$ with $\text{Init} \subseteq \hat{W} \subseteq F$, and a unique least restrictive controller, $\hat{g}$, that renders $\hat{W}$ invariant [25].

Controlled Invariance Problem (CIP) *Given a DTHS and a set $F \subseteq \mathbf{S}$ compute the maximal controlled invariant subset of F, $\hat{W}$, and the least restrictive controller, $\hat{g}$, that renders $\hat{W}$ invariant.*

2.3 The Controlled Invariance Algorithm (CIA)

We first present a conceptual algorithm for solving the CIP for general DTHS. The algorithm is an extension of the algorithm proposed by Bertsekas [2]. It is based on the the computation of the operator $\text{Pre} : 2^{\mathbf{S}} \to 2^{\mathbf{S}}$, $W \mapsto \{s \in W \mid \psi(s)\}$, where

$$\begin{aligned} \psi(s) = \exists \sigma \in \mathbf{\Sigma}\, \exists u \in \mathbf{U}\, \forall \delta \in \mathbf{\Delta}\, \forall d \in \mathbf{D}\, \forall q' \in \mathbf{Q}, \\ [x \in \text{Inv}_q(v) \Rightarrow r_q(x, \sigma, u, \delta, d) \subseteq W_q] \wedge \\ [x \in G_{qq'}(v) \Rightarrow R_{qq'}(x, \sigma, u, \delta, d) \subseteq W_{q'}] \end{aligned} \tag{1}$$

with $W_q = \{x \in \mathbf{X} \mid (q, x) \in W\}$.

Algorithm 1 (Controlled Invariance Algorithm)

```
initialization: $W^0 = F$, $W^{-1} = \mathbf{S}$, $l = 0$
while $W^l \neq W^{l-1}$ do
    $W^{l+1} = \text{Pre}(W^l)$
    $l = l + 1$
end while
set $\hat{W} = \bigcap_{l \geq 0} W^l$
set $\hat{g}(s) = \begin{cases} \{(\sigma, u) \in \mathbf{\Sigma} \times \mathbf{U} \mid \phi(s, \sigma, u)\} & s \in \hat{W} \\ \mathbf{\Sigma} \times \mathbf{U} & s \notin \hat{W} \end{cases}$
```

where ϕ is the same as ψ in (1), but without the quantifiers for σ and u. To implement the controlled invariance algorithm one needs to be able to (a) encode sets of states, perform intersection and complementation, and test for emptiness, (b) compute the Pre of a set, and (c) guarantee that a fixed point is reached after a finite number of iterations. For classes of DTHS for which (a) and (b) are satisfied we say that the CIP is *semi-decidable*; if all three conditions are satisfied we say that the CIP is *decidable*.

3 Semi-decidable Controller Synthesis

In this section, we show how the CIA (which in general is not computable) can be implemented for a special class of DTHS. We say that $(H, \Box F)$, is **definable** in a theory $\mathcal{T}$ if $\mathbf{Q}$, $\mathbf{\Sigma}$, and $\mathbf{\Delta}$ are finite sets; $\mathbf{X} = \mathbb{R}^n$, $\mathbf{U} \subseteq \mathbb{R}^{n_u}$ and $\mathbf{D} \subseteq \mathbb{R}^{n_d}$; and for all $q, q' \in \mathbf{Q}$, $x \in \mathbf{X}$ and $v \in \mathbf{V}$ the sets $\mathbf{U}$, $\mathbf{D}$, Init_q, $\mathrm{Inv}_q(v)$, $r_q(x, v)$, $G_{qq'}(v)$, $R_{qq'}(x, v)$, and $F_q = \{x \in \mathbf{X} | (q, x) \in F\}$ are definable in the same theory. For example, we denote by $\mathrm{Lin}(\mathbb{R})$ the theory of linear constraints and by $\mathrm{OF}(\mathbb{R})$ the theory of polynomial constraints.

In order to determine whether each iteration of the CIA is computable one needs to be able to decide whether $W^{l+1} = \{s : \psi^{l+1}(s)\} \subset W^l = \{s : \psi^l(s)\}$ or not. Since $\mathrm{OF}(\mathbb{R})$ is decidable [17, 23], the question can be solved if the formulas ψ^l and ψ^{l+1} belong to $\mathrm{OF}(\mathbb{R})$. We show this by induction when $(H, \Box F)$ is definable in $\mathrm{OF}(\mathbb{R})$. First, it is clear that ψ^0 is definable in $\mathrm{OF}(\mathbb{R})$. Now assume that ψ^l is definable. Then ψ^{l+1} (See (1)) is not directly in $\mathrm{OF}(\mathbb{R})$ since it contains some quantifiers on discrete variables. Nevertheless, it is straightforward to see that $\psi^{l+1}(s)$ is equivalent to a first order formula in the corresponding language, since the existential quantifier over σ is equivalent to a disjunction and the universal quantifier over δ is equivalent to a conjunction. We conclude that for definable systems the question $W^{l+1} \subset W^l$ can be decided, hence each iteration of the algorithm is computable. However, there is no guarantee that the algorithm will terminate in a finite number of iterations. We have just proven that:

Theorem 1 *If $(H, \Box F)$ is definable in $\mathrm{OF}(\mathbb{R})$, then the CIP is semi-decidable.*

Remark 1 *Notice that the class of systems for which the CIP is semi-decidable is more general than its continuous time counterpart: While for DTHS the problem is semi-decidable when $(H, \Box F)$ is definable in $\mathrm{OF}(\mathbb{R})$, in the continuous case the system is required to be triangular and $\mathbf{U}$ and $\mathbf{D}$ must be rectangles [20].*

Remark 2 *Different methods have been proposed for performing quantifier elimination in $\mathrm{OF}(\mathbb{R})$ [1, 17, 23], and the process can be automated using symbolic tools [6]. However, the quantifier elimination procedure is, in general, hard, both in theory and in practice, since the solvability may be doubly exponential [1].*

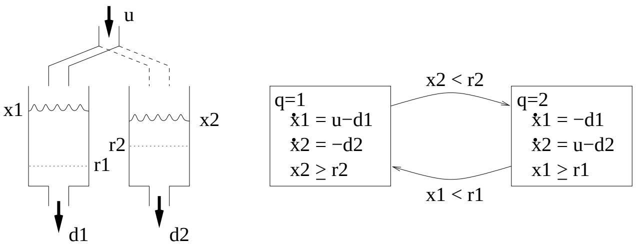

Figure 1: The water tank system

Example 1 (Semi-decidable controller synthesis) *Consider the water tank system shown in Figure 1. For $i = 1, 2$, let x_i denote the volume of water in Tank i, and d_i denote the flow of water out of Tank i. Let u denote the flow of water into the system, dedicated exclusively to either Tank 1 or Tank 2 at each time instant. The control task is to keep the water volumes above levels l_1 and l_2, respectively. This is to be achieved by a switched control strategy that switches the inflow to Tank 1 whenever $x_1 < r_1$ and to Tank 2 whenever $x_2 < r_2$. We assume that $x_1[0] \geq r_1 > l_1 > 0$ and $x_2[0] \geq r_2 > l_2 > 0$. The continuous dynamics of the water tank are discretized with period τ, so that it can be modeled as the following DTHS:*

- $\mathbf{Q}$=$\{1, 2\}$, $\mathbf{\Sigma}$= $\mathbf{\Delta}$ =$\{1\}$;
- $\mathbf{X}$=$\mathbb{R}^2$, $\mathbf{U}$=$[u_m, u_M]$, $\mathbf{D}$=$[d_m, d_M]$×$[d_m, d_M]$;
- $\mathrm{Init} = \mathbf{Q} \times \{x \in \mathbf{X} :\ (x_1 \geq r_1) \wedge (x_2 \geq r_2)\}$;
- Inv_1=$\{x \in \mathbf{X} : x_2 \geq r_2\}$, Inv_2=$\{x \in \mathbf{X} : x_1 \geq r_1\}$;
- G_{12}=$\{x \in \mathbf{X} :\ x_2 < r_2\}$, G_{21}=$\{x \in \mathbf{X} :\ x_1 < r_1\}$;
- $r_q(x, u, d) = x + \tau(b_q^T u - d)$, $b_1 = (1, 0)$, $b_2 = (0, 1)$;
- $R_{12}(x, u, d) = R_{21}(x, u, d) = x - \tau d$.

We implemented the CIA in Mathematica *for a water tank system with the following parameters: $u_m = 0$, $u_M = 12$, $d_m = 0$, $d_M = 1$, $\tau = 1$, $r_1 = r_2 = 20$ and $l_1 = l_2 = 10$. The algorithm converges after 11 iterations to the following solution:*

$$\hat{W}_1 = \{x \in \mathbf{X} \mid x \geq (10, 20) \vee x \geq (21, 11)\}$$
$$\hat{W}_2 = \{x \in \mathbf{X} \mid x \geq (20, 10) \vee x \geq (11, 21)\}.$$

4 Decidable Controller Synthesis

Even though for a definable CIP each iteration of the CIA is computable, termination in a finite number of iterations is not guaranteed. To the best of our knowledge, the only existing results on the decidability of the CIP are for linear systems with polyhedral constraints, so we restrict our attention to this class of systems.

More formally, a *linear CIP* (LCIP) consists of [25]:

- a Linear DTS (LDTS), i.e. a DTS with $\mathbf{X} = \mathbb{R}^n$, $\mathbf{U} = \{u \in \mathbb{R}^{n_u} \mid Eu \leq \eta\} \subseteq \mathbb{R}^{n_u}$, $\mathbf{D} = \{d \in \mathbb{R}^{n_d} \mid Gd \leq \gamma\} \subseteq \mathbb{R}^{n_d}$, $\mathrm{Init} = \{x \in \mathbf{X} \mid Jx \leq \theta\}$ and a reset relation given by $f(x, u, d) = \{Ax + Bu + Cd\}$, where $A \in \mathbb{Q}^{n \times n}$, $B \in \mathbb{Q}^{n \times n_u}$, $C \in \mathbb{Q}^{n \times n_d}$, $E \in \mathbb{Q}^{m_u \times n_u}$, $G \in \mathbb{Q}^{m_d \times n_d}$, $\eta \in \mathbb{Q}^{m_u}$, $\gamma \in \mathbb{Q}^{m_d}$, $J \in \mathbb{Q}^{n \times m_i}$ and $\theta \in \mathbb{Q}^{m_i}$ with m_u, m_d and m_i being the number of constraints on the control, disturbance and initial conditions, respectively; and,
- a set $F = \{x \in \mathbb{R}^n \mid Mx \leq \beta\}$ where $M \in \mathbb{Q}^{m \times n}$, $\beta \in \mathbb{Q}^m$ and m is the number of state constraints.

In this section, we extend our results in [25] to a much larger class of linear systems. Our new results are based on the following theorem (See [24] for the proof of the theorem and propositions):

Theorem 2 *If $\exists\, l \geq 1$ such that $\forall\, x_0 \in F$ there is a sequence $\{u[k]\}_{k=0}^{l-1}$ in $\mathbf{U}$ such that for all sequences $\{d[k]\}_{k=0}^{l-1}$ in $\mathbf{D}$ we have $x[l] \in F$, then $W^{l+1} = W^l$.*

For LDTS, the theorem reduces to the following:

Proposition 1 *Let $\mathcal{C}^l(A,X) = [A^{l-1}X \cdots AX \quad X]$, $u^l = [u_0^T \cdots u_{l-1}^T]^T \in \mathbf{U}^l = \mathbf{U} \times \cdots \times \mathbf{U}$ and $d^l = [d_0^T \cdots d_{l-1}^T]^T \in \mathbf{D}^l = \mathbf{D} \times \cdots \times \mathbf{D}$. Let $\Lambda = [\Lambda_1\, \Lambda_2]$ be the matrix whose rows are the generators of the non-negative left kernel of $\begin{bmatrix} M\mathcal{C}^l(A,B) \\ E \end{bmatrix}$. If $\exists\, l \geq 1$ such that*

$$\max_{x \in F} \Lambda_1 M A^l x \leq \Lambda_1(\beta - \max_{d^l \in \mathbf{D}^l} M\mathcal{C}^l(A,C)d^l) + \Lambda_2 \eta$$

with the maximum taken componentwise, then the LCIP is decidable and the CIA terminates in at most $l+1$ steps.

Proposition 1 states that in order to check for the decidability of the LCIP problem it is sufficient to solve a finite number of linear programs (polynomial time) plus one Fourier-Motzkin elimination problem (worst case exponential [12]). However, there is no way of detecting if the algorithm terminates in an infinite number of iterations, which happens for $l = \infty$. The following propositions are special cases under which l is guaranteed to be finite.

Proposition 2 *If A is nilpotent of index l, $\Lambda_1(\beta - \max_{d^l \in \mathbf{D}^l} M\mathcal{C}^l(A,C)d^l) + \Lambda_2 \eta \geq 0$, then the LCIP is decidable and the CIA terminates in at most $l+1$ steps.*

Proposition 3 *If (A,B) is controllable, $\mathbf{U} = \mathbb{R}^{n_u}$ and*

$$\max_{d^n \in \mathbf{D}^n} M\mathcal{C}^n(A,C)d^n \leq \beta \tag{2}$$

then the CIP problem is decidable and the CIA terminates in at most $n+1$ steps.

Remark 3 *Notice that the class of linear systems for which the controlled invariance problem is decidable is also more general than its continuous counterpart: While in the discrete time case, the problem is decidable if (A,B) is controllable or A is nilpotent, in the continuous time case, the problem is decidable if (A,B) and (A,C) are normal, $\mathbf{U}$ and $\mathbf{D}$ are rectangles and A is either nilpotent or diagonalizable with real rational eigenvalues [18].*

Example 2 (Decidable controller synthesis) *We consider Example 2 in [25], which requires an infinite number of iterations to converge. The LDTS is defined by $\mathbf{U} = \mathbb{R}$, $\mathbf{D} = [-1, 1]$,*

$$A = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$
$$M = \begin{bmatrix} 1 & 1 \\ -1 & -3 \\ 1 & -1 \\ -3 & 1 \end{bmatrix} \text{ and } \beta = \begin{bmatrix} 100 \\ -50 \\ 100 \\ -50 \end{bmatrix}.$$

Even though the pair (A,B) is controllable, we have $\delta = \max_{d^2 \in \mathbf{D}^2} M\mathcal{C}^2(A,C)d^2 = [5\ 11\ 1\ 3]^T \not\leq \beta$. Therefore, this example does not satisfy all the conditions of Proposition 3.

If $\mathbf{D} = [17, 18]$, then $\delta = [90\ {-187}\ {-17}\ {-51}]^T \leq \beta$. Indeed, using our Controlled Invariance Toolbox *[16] for MATLAB, the algorithm terminates in 2 iterations.*

If $\mathbf{D} = [15, 20]$, then $\delta = [100\ {-165}\ {-15}\ {-45}]^T \not\leq \beta$. However, the algorithm still terminates in 2 iterations, showing that the conditions in Proposition 3 are not necessary. In this case, the conditions of Proposition 1 are met for $l = 2$.

5 CIP for LDTS with Ellipsoidal Constraints

Although for the classes of linear systems in Propositions 2 and 3 the LCIP is decidable, the computational complexity of exactly solving the problem is worst case exponential [12]. Because of this, in this section we specialize the implementation of the CIA to the case where the sets F, $\mathbf{U}$ and $\mathbf{D}$ are ellipsoids. Notice that given a LDTS, if F and $\mathbf{U}$ are convex so are $\hat{W}$ and $\hat{g}(x)$. Hence the CIP is suitable for convex optimization algorithms, such as convex programming (CP).

Let $\mathcal{E}_1(P, \hat{x}) = \{x \mid (x - \hat{x})^T P^{-1}(x - \hat{x}) \leq 1\}$ be an ellipsoid with positive definite *shape matrix* $P \succ 0$ and *center* $\hat{x}$. A second representation is $\mathcal{E}_2(E, \hat{x}) = \{x \mid x = \hat{x} + Ez, ||z|| \leq 1\}$ where $E = P^{1/2}$. A third representation is given by the linear matrix inequality (LMI):

$$\begin{bmatrix} P & x - \hat{x} \\ (x - \hat{x})^T & 1 \end{bmatrix} \succeq 0.$$

To make the subsequent LMIs more readable, we use a $*$ to denote elements in the lower triangular part of a symmetric matrix. Further, we use $\log\det(E)$ as a measure of the volume of the ellipsoid $\mathcal{E}_2(E, \hat{x})$.

5.1 CP inner approximations of $\mathrm{Pre}^m(W^0)$

We assume that $F = W^0 = \mathcal{E}_1(\Omega_0, \hat{x}_0)$, $\mathbf{U} = \mathcal{E}_1(\Gamma, \hat{u})$ and $\mathbf{D} = \mathcal{E}_1(\Delta, \hat{d})$, where $\Omega_0 \succ 0 \in \mathbb{R}^{n \times n}$, $\Gamma \succ 0 \in \mathbb{R}^{n_u \times n_u}$, $\Delta \succ 0 \in \mathbb{R}^{n_d \times n_d}$, $\hat{x}_0 \in \mathbb{R}^n$, $\hat{u} \in \mathbb{R}^{n_u}$ and $\hat{d} \in \mathbb{R}^{n_d}$.

At each iteration of the controlled invariance algorithm we need to compute $W^{l+1} = \text{Pre}(W^l)$ defined by

$$\left\{x \in W^l \mid \exists u_l \in \mathbf{U}\ \forall d_l \in \mathbf{D},\ Ax + Bu_l + Cd_l \in W^l\right\}.$$

For $l = 0$, the inner formula is equivalent to

$$\forall d_0\ ,\ ||\Delta^{-1/2}(d_0 - \hat{d})|| \leq 1 \Rightarrow \begin{bmatrix} \Omega_0 & v_0 \\ v_0^T & 1 \end{bmatrix} \succeq 0, \qquad (3)$$

with $v_0 = Ax + Bu_0 + Cd_0 - \hat{x}_0$. In order to replace the universal quantifier $\forall$ in (3) by an existential quantifier $\exists$, we use a lemma from [8]:

Lemma 1 *Let $\mathcal{F} = \mathcal{F}^T$, $\mathcal{L}$ and $\mathcal{R}$ be given matrices of appropriate size. We have*

$$\left(\forall Z, ||Z|| \leq \rho\ \Rightarrow\ \mathcal{F} + \mathcal{L}Z\mathcal{R} + (\mathcal{L}Z\mathcal{R})^T \succeq 0\right) \iff \left(\exists \tau \geq 0 \mid \begin{bmatrix} \mathcal{F} - \tau\mathcal{R}^T\mathcal{R} & \rho\mathcal{L} \\ \rho\mathcal{L}^T & \tau I \end{bmatrix} \succeq 0\right).$$

We apply Lemma 1 to $Z = \Delta^{-1/2}(d_0 - \hat{d})$, $\rho = 1$,

$$\mathcal{F} = \begin{bmatrix} \Omega_0 & v_{01} \\ v_{01}^T & 1 \end{bmatrix},\ \mathcal{L} = \begin{bmatrix} C\Delta^{1/2} \\ 0 \end{bmatrix} \text{ and } \mathcal{R} = \begin{bmatrix} 0 & 1 \end{bmatrix},$$

where $v_{01} = Ax + Bu_0 + C\hat{d} - \hat{x}_0$. We then obtain the following formula for $\text{Pre}(W^0)$:

$$\{x \in W^0 \mid \exists u_0 \in \mathbf{U}\ \exists \tau_0,\ \begin{bmatrix} \Omega_0 & v_{01} & C\Delta^{1/2} \\ * & 1 - \tau_0 & 0 \\ * & 0 & \tau_0 I \end{bmatrix} \succeq 0\}. \qquad (4)$$

We now compute an inner approximation $\mathcal{E}_2(E_1, \hat{x}_1)$ of $\text{Pre}(W^0)$. Such approximation must satisfy: $\forall x \in \mathcal{E}_2(E_1, \hat{x}_1) \Rightarrow x \in \text{Pre}(W^0)$, which is equivalent to $\forall z, ||z|| \leq 1,\ \exists u_0 \in \mathbf{U}\ \exists \tau_0 \mid$

$$\begin{bmatrix} \Omega_0 & \hat{x}_1 - \hat{x}_0 + E_1 z \\ * & 1 \end{bmatrix} \succeq 0, \begin{bmatrix} \Omega_0 & \hat{v}_{01} + AE_1 z & C\Delta^{1/2} \\ * & 1 - \tau_0 & 0 \\ * & 0 & \tau_0 I \end{bmatrix} \succeq 0,$$

where $\hat{v}_{01} = A\hat{x}_1 + Bu_0 + C\hat{d} - \hat{x}_0$. In order to apply Lemma 1, we first exchange the quantifiers[1]. Since $\forall z \bigwedge_{l=1}^{m} \psi^l(z) \equiv \bigwedge_{l=1}^{m} \forall z\, \psi^l(z)$, we can apply Lemma 1 to each LMI separately. Let $Z^1 = Z^2 = z$, $\rho^1 = \rho^2 = 1$, $\tau^1 = \gamma_0$, $\tau^2 = \gamma_1$,

$$\mathcal{F}^1 = \begin{bmatrix} \Omega_0 & \hat{x}_1 - \hat{x}_0 \\ * & 1 \end{bmatrix},\ \mathcal{L}^1 = \begin{bmatrix} E_1 \\ 0 \end{bmatrix}, \mathcal{R}^1 = \begin{bmatrix} 0\,1 \end{bmatrix},$$

$$\mathcal{F}^2 = \begin{bmatrix} \Omega_0 & \hat{v}_{01} & C\Delta^{1/2} \\ * & 1 - \tau_0 & 0 \\ * & 0 & \tau_0 I \end{bmatrix},\ \mathcal{L}^2 = \begin{bmatrix} AE_1 \\ 0 \\ 0 \end{bmatrix}, \mathcal{R}^2 = \begin{bmatrix} 0\,1\,0 \end{bmatrix}.$$

Then the ellipsoidal inner approximation of $\text{Pre}(W^0)$ can be obtained as the solution of the following CP:

$$\begin{array}{ll} \max & \log\det(E_1) \\ \text{s.t.} & E_1 \succ 0, \\ & \begin{bmatrix} \Omega_0 & \hat{x}_1 - \hat{x}_0 & E_1 \\ * & 1 - \gamma_0 & 0 \\ * & 0 & \gamma_0 I \end{bmatrix} \succeq 0, \begin{bmatrix} \Gamma & u_0 - \hat{u} \\ * & 1 \end{bmatrix} \succeq 0, \\ & \begin{bmatrix} \Omega_0 & \hat{v}_{01} & C\Delta^{1/2} & AE_1 \\ * & 1 - \tau_0 - \gamma_1 & 0 & 0 \\ * & 0 & \tau_0 I & 0 \\ * & 0 & 0 & \gamma_1 I \end{bmatrix} \succeq 0. \end{array}$$

This procedure can be generalized to obtain an ellipsoidal inner approximation $\mathcal{E}_2(E_m, \hat{x}_m)$ of $\text{Pre}^m(W^0)$ by solving the CP:

$$\begin{array}{l} \max \log\det(E_m) \\ \text{ s.t. } E_m \succ 0, \\ \quad \begin{bmatrix} \Omega_0 & \hat{x}_m - \hat{x}_0 & E_m \\ * & 1 - \gamma_0 & 0 \\ * & 0 & \gamma_0 I \end{bmatrix} \succeq 0, \begin{bmatrix} \Gamma & u_i - \hat{u} \\ * & 1 \end{bmatrix}_{i=0}^{m-1} \succeq 0, \\ \quad \begin{bmatrix} \Omega_0 & \hat{v}_{j-1,m} & C\Delta^{1/2} & \cdots & A^{j-1}C\Delta^{1/2} & A^j E_m \\ * & \delta_{j,m} & 0 & \cdots & 0 & 0 \\ * & 0 & \tau_{m-j} I & & 0 & 0 \\ \vdots & \vdots & & \ddots & \vdots & \vdots \\ * & 0 & 0 & \cdots & \tau_{m-1} I & 0 \\ * & 0 & 0 & \cdots & 0 & \gamma_j I \end{bmatrix}_{j=1}^{m} \succeq 0, \end{array}$$

with $\hat{v}_{j-1,m} = A^j \hat{x}_m + \sum_{i=0}^{j-1} A^i (Bu_{m+i-j} + C\hat{d}) - \hat{x}_0$, for all $j = 1 \ldots m$, and $\delta_{j,m} = 1 - \sum_{i=m-j}^{m-1} \tau_i - \gamma_j$.

5.2 A "heuristic" CP approach

Here we combine inner and outer approximations heuristically. We first compute a polytope containing $\text{Pre}(W^0)$ in (4) by solving a series of CPs of the type:

$$\begin{array}{ll} \max & w_i^T x \\ \text{s.t.} & \begin{bmatrix} \Omega_0 & x - \hat{x}_0 \\ * & 1 \end{bmatrix} \succeq 0, \begin{bmatrix} \Gamma & u_0 - \hat{u} \\ * & 1 \end{bmatrix} \succeq 0, \\ & \begin{bmatrix} \Omega_0 & v_{01} & C\Delta^{1/2} \\ * & 1 - \tau_0 & 0 \\ * & 0 & \tau_0 I \end{bmatrix} \succeq 0, \end{array}$$

where the vectors $w_i \in \mathbb{R}^n, i = 1 \ldots p$ are chosen arbitrarily. Then an outer approximation of $\text{Pre}(W^0)$ is given by $\{x \mid Mx \leq \beta\}$ where w_i^T is the i-th row of M and β_i is the optimal value of the convex program associated to w_i, $i = 1 \ldots p$. Then we compute an ellipsoidal inner approximation $\mathcal{E}_2(E_1, \hat{x}_1)$ of the obtained polytope by solving the CP:

$$\begin{array}{l} \max \log\det(E_1) \\ \text{ s.t. } E_1 \succ 0,\ \begin{bmatrix} (\beta_i - w_i^T \hat{x}_1) I & E_1 w_i \\ (E_1 w_i)^T & \beta_i - w_i^T \hat{x}_1 \end{bmatrix}_{i=1}^{p} \succeq 0. \end{array} \qquad (5)$$

Letting $\Omega_1 = E_1^2$, we can compute an approximation of $\text{Pre}^2(W^0)$ by using the same two step procedure. Generalizing, we get an ellipsoidal approximation $\mathcal{E}_1(\Omega_m, \hat{x}_m)$ of $\text{Pre}^m(W^0)$.

[1] This may reduce the size of the set we want to approximate, hence the inner ellipsoidal approximation may be conservative

5.3 Experimental Results

We implemented the CP and heuristic CP algorithms in SDPSOL and MATLAB. Our results are compared with the ones given by the exact algorithm for LDTS [7] (which we assume as the correct solution) and with the ellipsoidal method by Bertsekas [2].

Example 3 *The LDTS is defined by* $\mathbf{X} = \mathbb{R}^2$,

$$A = \begin{bmatrix} 0 & 0.5 \\ -1 & 1 \end{bmatrix},\ B = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\ \text{and}\ C = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

The ellipsoids $\mathbf{U}$ *and* $\mathbf{D}$ *are defined by:*

$$\Gamma = \begin{bmatrix} 0.0625 & 0 \\ 0 & 0.1406 \end{bmatrix},\ \hat{u} = \begin{bmatrix} 0 \\ -0.125 \end{bmatrix},\ \Delta = 0.01,\ \hat{d} = 0.$$

The set F *is defined by*

$$\Omega_0 = \begin{bmatrix} 660.3 & 416.8 \\ 416.8 & 1324.7 \end{bmatrix} \quad \text{and} \quad \hat{x}_0 = \begin{bmatrix} 6.5 \\ 20.4 \end{bmatrix}.$$

Figure 2 shows a comparison of the different methods. The heuristic CP approach gives the largest ellipse, but the solution is not *a subset of the correct solution. The results of the CP method for* $m \geq 2$ *are the same as those for* $m = 2$*, and better than the corresponding ones for* $m = 1$*. The algorithm by Bertsekas gives the smallest ellipsoid.*

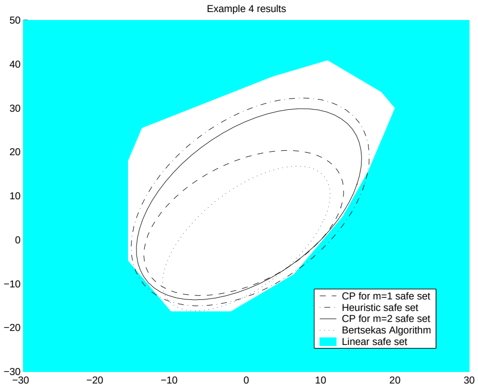

Figure 2: Comparison of CP, heuristic CP, exact linear, and Bertsekas algorithms for Example 3.

6 Conclusions

This paper studied classes of DTHS for which the CIA algorithm is computable and guaranteed to terminate. First, we proposed an implementation of the algorithm based on quantifier elimination, which led to a semi-decidability result for definable systems. Second, we found that for LDTS which are either nilpotent or controllable and have bounded disturbances, the CIP is decidable. Finally, we proposed two algorithms for solving the CIP for LDTS with ellipsoidal constraints and showed that they give better estimations than the ellipsoidal method by Bertsekas [2] and are more efficient than the exact one for linear constraints [7].

References

[1] D. Arnon, G. Collins, and S. McCallum. Cylindrical algebraic decomposition I: the basic algorithm. *SIAM Journal on Computing*, 13(4):865–877, 1984.

[2] D. Bertsekas and I. Rhodes. On the minimax reachability of target sets and target tubes. *Automatica*, 7:233–247, 1971.

[3] F. Blanchini. Ultimate boundedness control for discrete time uncertain systems via set induced Lyapunov functions. *IEEE Transactions on Automatic Control*, 39(2):428–433, 1994.

[4] F. Blanchini and M. Sznaier. Persistent disturbance rejection via static-state feedback. *IEEE Transactions on Automatic Control*, 40(5):1127–1131, 1995.

[5] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* Philadelphia: SIAM, 1994.

[6] A. Dolzmann and T. Sturm. REDLOG: Computer algebra meets computer logic. *ACM SIGSAM Bulletin*, 31(2):2–9, 1997.

[7] C. Dórea and J. Hennet. (A,B)-Invariant polyhedral sets of linear discrete time systems. *Journal of Optimization Theory and Applications*, 103(3):521–542, 1999.

[8] L. El Ghaoui and G. Calafiore. *Robustness in Identification and Control*, chapter Worst-Case Simulation of Uncertain Systems. Springer Verlag, 1999.

[9] J. Glover and F. Schweppe. Control of linear dynamic systems with set constrained disturbances. *IEEE Transactions on Automatic Control*, 16(5):411–423, 1971.

[10] A. Kurzhanski and P. Varaiya. Ellipsoidal techniques for reachability analysis. In *Hybrid Systems: Computation and Control*, volume 1790 of *LNCS*, pages 202–214. Springer Verlag, 2000.

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

[12] C. Lassez and J. Lassez. Quantifier elimination for conjunctions of linear constraints via a convex hull algorithm. In *Symbolic and Numeric Computation for Artificial Intelligence*, pages 103–122. Academic Press, 1992.

[13] J. Lygeros, K. Johansson, S. Sastry, and M. Egerstedt. On the existence of executions of hybrid automata. In *IEEE Conference on Decision and Control*, pages 2249–2254, 1999.

[14] J. Lygeros, C. Tomlin, and S. Sastry. Controllers for reachability specifications for hybrid systems. *Automatica*, pages 349–370, March 1999.

[15] G. Pappas. *Hybrid Systems: Computations and Abstractions.* PhD thesis, Department of EECS, UC Berkeley, 1998.

[16] S. Schaffert. *Controller Synthesis for Discrete Time Hybrid Systems with Safety Specifications.* MSc thesis, Department of EECS, UC Berkeley, 2001.

[17] A. Seidenberg. A new decision method for elementary algebra. *Annals of Mathematics*, 60:387–374, 1954.

[18] O. Shakernia, G. Pappas, and S. Sastry. Decidable controller synthesis for classes of linear systems. In *Hybrid Systems: Comp. and Cont.*, vol. 1790, pp. 407–420. Springer Verlag, 2000.

[19] O. Shakernia, G. Pappas, and S. Sastry. Semi-decidable controller synthesis for classes of linear hybrid systems. In *IEEE Conference on Decision and Control*, pp. 1834–1839, 2000.

[20] O. Shakernia, G. Pappas, and S. Sastry. Semi-decidable synthesis for triangular hybrid systems. In *Hybrid Systems: Comp. and Cont.*, vol. 2034, pp. 487–500. Springer Verlag, 2001.

[21] J. Shamma. Nonlinear state feedback for l^1 optimal control. *Systems and Control Letters*, 21:265-270, 1993.

[22] J. Shamma. Optimization of the l^∞induced norm under full state feedback. *IEEE Transactions on Automatic Control*, 41(4):533-544, 1996.

[23] A. Tarski. *A decision method for elementary algebra and geometry.* University of California Press, 1951.

[24] R. Vidal, S. Schaffert, J. Lygeros, and S. Sastry. Controlled invariance of discrete time systems. Technical Report UCB/ERL M99/65, ERL, UC Berkeley, 2001.

[25] R. Vidal, S. Schaffert, J. Lygeros, and S. Sastry. Controlled invariance of discrete time systems. In *Hybrid Systems: Comp. and Cont.*, vol. 1790, pp. 437–450. Springer Verlag, 2000.

[26] H. Witsenhausen. A minimax control problem for sampled linear systems. *IEEE Transactions on Automatic Control*, 13(1):5–20, 1968.