Efficient computation of the one-step robust sets for piecewise affine systems with polytopic additive uncertainties

Mario Vašak, Mato Baotić and Nedjeljko Perić

Abstract— **We propose a modification of the existing algorithm for computing the one-step robust sets for the discrete-time piecewise affine (DTPWA) systems subject to additive polytope-bounded uncertainties. In the old algorithm the Minkowski set difference between a union of polytopes (the so-called P-collection) and a polytope needs to be computed, which in turn calls twice for the computation of the set difference between a polytope and a P-collection. These set operations become computationally very demanding with the increasing number of polytopes in the P-collection and thus limit the practical applicability of the algorithm. In this paper we propose a more efficient procedure that avoids explicit evaluation of the Minkowski set difference. Our algorithm, on average, significantly reduces the complexity of the computation. We illustrate this on several examples of DTPWA systems for which the maximal robust positively invariant set is computed.**

I. Introduction

Discrete-time piecewise affine (DTPWA) models [1] are a class of hybrid models [2] that are particulary useful in practice due to their ability to approximate dynamic behavior of nonlinear processes arbitrarily well. A DTPWA model comprises several affine state-update equations (dynamics) each of which is valid in a different polyhedral part of the state-input space. In recent years these models have been intensively studied and very useful tools (see Multi-Parametric Toolbox (MPT) [3]) have been developed for the off-line optimal controller synthesis and the DTPWA model identification. However, some of the implemented algorithms are not yet mature and there is still much space (and opportunity) for improving the efficiency of the overall computations. The overall objectives are to speedup the computation and to obtain controllers which are suitable for practical applications.

One of the most important objectives in control practice is to keep the output of the system close to the desired value under the presence of constraints and process-model discrepancies, i.e. to synthesize a robust controller for a given model. For DTPWA models, this is from a computational point of view a very hard task. To simplify the computation, the uncertainty is usually modeled as an additive polytope-bounded entry in the state-update equation. One way of guaranteeing robustness of the closed loop system (in a receding horizon policy) is to solve the optimal control problem with the robust controlled invariant set [4] included as a terminal set constraint [5]. A robust controlled invariant set [6] for a DTPWA system subject to polytopic disturbances is computed in an iterative procedure. At every iteration step the one-step robust set [7] has to be computed. Because the computation of the one-step robust sets has a direct impact on the efficiency of the overall computation we looked into the ways of improving its efficiency.

Two algorithms for the computation of the one-step robust sets for a given set of (target) states Ω are reported in the literature. One algorithm is based on projections and set differences [8], [9] and the other is based on the Minkowski set (Pontryagin) differences [6], [10]. The algorithm based on projections allows the disturbance to be state-input dependent, while the algorithm based on Minkowski set differences is tailored for the additive polytope-bounded uncertainty. We consider the case when Ω is given as a union of a finite number of non-overlapping polyhedra, i.e. Ω is a P-collection. Such set shapes generally arise in the computation of robust controlled/positively invariant sets for PWA systems when the initial set for iterations is a polyhedron. Although both proposed algorithms for obtaining one-step robust sets can be compactly formulated, in the higher dimensional state-space the underlying computation grows rapidly with the increasing number of polyhedra in the P-collection Ω.

The main aim of this paper is to make those computations more efficient with a new approach in computing the one-step robust sets. Our method is based on the algorithm reported in [10] that uses the Minkowski set difference to obtain the appropriate target set for the one-step set computation of the nominal (non-perturbed) system. In [10] the authors show that in such a way computed one-step set is the same as the one-step robust set. In our algorithm the appropriate target set is not explicitly computed. It is rather presented through the so-called *admissible* set (in which the state can be) and *forbidden* set (in which the state must not be). The one-step sets are then computed for both admissible and forbidden sets and the set difference between them is computed for each dynamic, thus producing the one-step robust set. A comparative complexity analysis is performed for both methods which shows that in the worst-case they have the same complexity. However, we give arguments to explain superiority of our approach in practice (as confirmed in examples of computation of the maximal robust positively invariant sets for randomly generated DTPWA systems).

The paper is structured as follows. In Section II we introduce basic definitions and describe some of the set operations that are used for the computation of the one-step robust sets for DTPWA systems. In Section III we revise the existing algorithm for the one-step robust set computation

This work was supported by the Ministry of Science, Education and Sports of the Republic of Croatia.

Authors are with the Department of Control and Computer Engineering at the Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, HR-10000 Zagreb, Croatia mario.vasak@fer.hr

from [10] and analyze its computational bottlenecks, while in Section IV we introduce our modified algorithm and compare its worst-case complexity with the algorithm from [10]. Finally, in Section V we compare performances of both algorithms when computing the maximal robust positively invariant set for several randomly generated DTPWA systems.

II. Basic Definitions and Set Operations

In this section we introduce the notation and define some set operations used throughout the paper.

A hyperplane, open half-space and closed half-space are sets of the form $\{x \in \mathbb{R}^n |\ h^T x = k\}$, $\{x \in \mathbb{R}^n |\ h^T x < k\}$ and $\{x \in \mathbb{R}^n |\ h^T x \leq k\}$, respectively, where $h \in \mathbb{R}^n$, $k \in \mathbb{R}$. A set $\mathcal{A}$ that can be described as an intersection of a finite number of half-spaces is called *polyhedron*. A bounded and closed polyhedron is called *polytope*. Throughout this paper a polytope $\mathcal{A}$ will be described in the so-called H-representation

$$\mathcal{A} = \{x |\ H_{\mathcal{A}} x \leq K_{\mathcal{A}}\},$$

where each row of $H_{\mathcal{A}} x \leq K_{\mathcal{A}}$, with $H_{\mathcal{A}} \in \mathbb{R}^{m \times n}$ and $K_{\mathcal{A}} \in \mathbb{R}^m$, defines a single closed half-space (also known as *constraints*). We say that the polytope $\mathcal{A}$ is in its minimal representation if none of the constraints in $H_{\mathcal{A}}$ and $K_{\mathcal{A}}$ are redundant (i.e. removal of any of them from $H_{\mathcal{A}}$ and $K_{\mathcal{A}}$ would change the polytope). A *P-collection* is a union of a finite number of polytopes.

Let $\mathcal{A} \subseteq \mathbb{R}^n$ and $\mathcal{B} \subseteq \mathbb{R}^n$ be two sets. The set complement of $\mathcal{A}$ is a set $\mathcal{A}^c := \{x \in \mathbb{R}^n |\ x \notin \mathcal{A}\}$, set intersection $\mathcal{A} \cap \mathcal{B} := \{x \in \mathbb{R}^n |\ x \in \mathcal{A},\ x \in \mathcal{B}\}$, set difference $\mathcal{A} \setminus \mathcal{B} := \{x \in \mathcal{A} |\ x \notin \mathcal{B}\}$, Minkowski set addition $\mathcal{A} \oplus \mathcal{B} := \{x + y |\ x \in \mathcal{A},\ y \in \mathcal{B}\}$ and Minkowski set (Pontryagin) difference $\mathcal{A} \ominus \mathcal{B} := \{x \in \mathbb{R}^n |\ x + y \in \mathcal{A},\ \forall y \in \mathcal{B}\}$. If $0 \in \mathcal{B}$, then $\mathcal{A} \ominus \mathcal{B} = \{x \in \mathcal{A} |\ x + y \in \mathcal{A},\ \forall y \in \mathcal{B}\}$. The set of interior points of $\mathcal{A}$ is denoted with $\mathrm{int}(\mathcal{A})$. The convex hull of a set $\mathcal{A}$, denoted with $\mathrm{convh}(\mathcal{A})$, is the smallest convex set that contains all the elements of $\mathcal{A}$. The envelope of a P-collection $\mathcal{P}$, $\mathrm{env}(\mathcal{P})$, is a polyhedron obtained by collecting (i.e. intersecting) all half-spaces $\mathcal{H}_i$ from polyhedra that define $\mathcal{P}$ for which $\mathcal{H}_i^c \cap \mathcal{P} = \emptyset$. From definitions it follows that $\mathrm{convh}(\mathcal{P}) \subseteq \mathrm{env}(\mathcal{P})$. For the set $\mathcal{C} \subseteq \mathbb{R}^n \times \mathbb{R}^m$ the projection of $\mathcal{C}$ onto $\mathbb{R}^n$ is $\mathrm{proj}_{\mathbb{R}^n}(\mathcal{C}) = \{x \in \mathbb{R}^n | \exists u \in \mathbb{R}^m :\ [x^T\ u^T]^T \in \mathcal{C}\}$.

A. Set Difference With Polyhedral Sets

In the following section it will be shown that the set difference computation between a polytope and a P-collection plays an important role in the overall complexity of the one-step robust set computation. We briefly investigate complexity of a set difference computation between a polytope $\mathcal{A}$ and a P-collection $\mathcal{B} := \bigcup_{i=1}^{N_B} \mathcal{B}_i$ in $\mathbb{R}^n$ for the set difference algorithm reported in [11] and currently used in MPT [3].

Let

$$\mathcal{A} = \{x \in \mathbb{R}^n |\ H_{\mathcal{A}} x \leq K_{\mathcal{A}}\},$$

and

$$\mathcal{B}_i = \{x \in \mathbb{R}^n |\ H_{\mathcal{B}_i} x \leq K_{\mathcal{B}_i}\},\ i = 1, \ldots, N_B.$$

be the minimal H-representations of the polytopes involved. Without loss of generality we can assume that $\mathcal{B}_i \subseteq \mathcal{A}$, for all $i = 1, \ldots, N_B$. By borrowing the notation from [11], [3] all inequalities from $\mathcal{B}_i$ that split $\mathcal{A}$ in two full-dimensional sets are called *active constraints*. For the case when $\mathcal{B}_i \subseteq \mathcal{A}$ active constraints are all those constraints that define $\mathcal{B}_i$ and do not participate in H-representation of $\mathcal{A}$. Let c be the total number of active constraints. An upper bound on the worst-case complexity of computing $\mathcal{A} \setminus \mathcal{B}$ is (for more details see [11])

$$\mathcal{O}_1\left(\left(\frac{c}{N_B}\right)^{N_B}\right), \tag{1}$$

while the number of regions that describe the set difference is bounded by

$$\sum_{i=1}^{n} \left(\begin{array}{c} c \\ i \end{array} \right) \sim \mathcal{O}_2(c^n). \tag{2}$$

B. Minkowski Set Difference With Polyhedral Sets

In [10] the authors describe a procedure for computing the Minkowski set difference between a P-collection $\mathcal{P}$ and a polytope $\mathcal{Q}$, $\mathcal{R} = \mathcal{P} \ominus \mathcal{Q}$, that follows the general relation $\mathcal{R} = [\mathcal{P}^c \oplus (-\mathcal{Q})]^c$ [6]. We repeat the main steps of that procedure in the following algorithm.

Algorithm 1 ([10]): **Minkowski set difference,** $\mathcal{R} = \mathcal{P} \ominus \mathcal{Q}$

1) $\mathcal{R}_1 = \mathrm{convh}(\mathcal{P})$ (or $\mathcal{R}_1 = \mathrm{env}(\mathcal{P})$);
2) $\mathcal{R}_2 = \mathcal{R}_1 \ominus \mathcal{Q}$;
3) $\mathcal{R}_3 = \mathcal{R}_1 \setminus \mathcal{P}$;
4) $\mathcal{R}_4 = \mathcal{R}_3 \oplus (-\mathcal{Q})$;
5) $\mathcal{R} = \mathcal{R}_2 \setminus \mathcal{R}_4$.

Note that the Minkowski set difference operation in step 2 of Algorithm 1 is not problematic, since both $\mathcal{R}_1$ and $\mathcal{Q}$ are polytopes and the Minkowski difference between two polytopes can be easily computed [3]. Note also that in the implementation of Algorithm 1 in [3] at all steps only closures of sets are computed. Since the set difference operation (such as the one in step 3 of Algorithm 1) in general generates sets that are neither open nor closed this may lead to the loss (or superfluous generation) of some lower-dimensional sets from the set $\mathcal{R}$.

In Algorithm 1 the set difference calculation between a polytope and a P-collection is carried out twice (step 3 and step 5). In most cases[1] these are the most time consuming parts of Algorithm 1. The set difference in step 5 of Algorithm 1 in particular can be extremely time consuming since the P-collection $\mathcal{R}_4$ is already a successor of a set difference operation in step 3 (see (1)–(2)).

[1]That is in those applications where vertex enumeration of the polytopes occuring within the Minkowski set addition is not problematic.

III. The one-step robust sets for DTPWA systems with polytopic uncertainties

A DTPWA model with polytope-bounded uncertainty is

$$\begin{aligned} x^+ &= A_i x + B_i u + f_i + w \\ &\quad \text{if } \left[\begin{smallmatrix} x \\ u \end{smallmatrix}\right] \in \mathcal{D}_i, \quad i = 1, \dots, s \ , \end{aligned} \tag{3}$$

where $x \in \mathbb{R}^n$ is the model state, x^+ is the successor state, $u \in \mathbb{R}^m$ is the control input, $\{\mathcal{D}_i\}_{i=1}^s$ is a non-overlapping polyhedral partition of the state-input space $\mathbb{R}^{n+m}$, $\mathcal{D}_i = \{\left[x^T \ u^T\right]^T \in \mathbb{R}^{n+m} | \ H_i x + L_i u \leq K_i\}$, and $w \in \mathcal{W} \subset \mathbb{R}^p$ is the uncertainty with polytope $\mathcal{W}$ that contains the origin. Note that constraints on states and inputs may be included in the descriptions of all $\mathcal{D}_i$. Equation (3) is compactly denoted with

$$x^+ = f(x, u, w). \tag{4}$$

If the control law is an a-priori fixed PWA function over polyhedra [11] we consider the DTPWA model of the closed-loop control system with the bounded uncertainty

$$x^+ = A_i^{\rm cl} x + f_i^{\rm cl} + w \quad \text{if} \quad x \in \mathcal{D}_i^{\rm cl}, \tag{5}$$

where $\{\mathcal{D}_i^{\rm cl}\}_{i=1}^{s^{\rm cl}}$ is a non-overlapping polyhedral partition of the state space, $\mathcal{D}_i^{\rm cl} = \{x \in \mathbb{R}^n | \ H_i^{\rm cl} x \leq K_i^{\rm cl}\}$. We write (5) in a more compact form

$$x^+ = f^{\rm cl}(x, w). \tag{6}$$

Let $\Omega \subset \mathbb{R}^n$ be a set represented with a non-overlapping P-collection. For the mapping f the *one-step robustly controllable set* of Ω in the space of interest $\mathbb{X} \subseteq \mathbb{R}^n$ is defined as

$$\mathrm{pre}(\Omega, f)_{\mathbb{X}} \triangleq \{x \in \mathbb{X} | \exists u, f(x, u, w) \in \Omega \ \forall w \in \mathcal{W}\}. \tag{7}$$

For the mapping $f^{\rm cl}$ the *one-step robust pre-set* of Ω in the space of interest $\mathbb{X} \subseteq \mathbb{R}^n$ is defined as

$$\mathrm{pre}(\Omega, f^{\rm cl})_{\mathbb{X}} \triangleq \{x \in \mathbb{X} | f^{\rm cl}(x, w) \in \Omega \ \forall w \in \mathcal{W}\}. \tag{8}$$

For the following exposition we also define the (nominal) one-step sets. A (nominal) *one-step controllable set* of Ω for f in the space of interest $\mathbb{X}$ is

$$\begin{aligned} &\mathrm{pre}(\Omega, f)^0_{\mathbb{X}} \triangleq \{x \in \mathbb{X} | \exists u, f(x, u, 0) \in \Omega\} = \\ &\textstyle\bigcup_{i=1}^s \{x \in \mathbb{X} | \exists u, \ [x^T \ u^T]^T \in \mathcal{D}_i, \ A_i x + B_i u + f_i \in \Omega\}, \end{aligned} \tag{9}$$

while the (nominal) *one-step pre-set* of Ω for $f^{\rm cl}$ in the space of interest $\mathbb{X}$ is

$$\begin{aligned} &\mathrm{pre}(\Omega, f^{\rm cl})^0_{\mathbb{X}} \triangleq \{x \in \mathbb{X} | f^{\rm cl}(x, 0) \in \Omega\} = \\ &\textstyle\bigcup_{i=1}^{s^{\rm cl}} \{x \in \mathbb{X} \cap \mathcal{D}_i^{\rm cl} | \ A_i^{\rm cl} x + f_i^{\rm cl} \in \Omega\}. \end{aligned} \tag{10}$$

Note that in this paper we use the term *one-step robust set* when we want to refer both to the one-step robustly controllable set and to the one-step robust pre-set. Similarly a term *one-step set* denotes both one-step controllable set and one-step pre-set.

We focus on the algorithms proposed in [10] for computing the one-step robustly controllable set of Ω and f (Algorithm 5.1. in [10]), and computing the one-step robust pre-set of Ω and $f^{\rm cl}$ (Algorithm 4.1 in [10]), which is also the current implementation in the MPT [3].

Algorithm 2 ([10]): **One-step robustly controllable set**

1) $\Omega^+ = \Omega \ominus \mathcal{W}$
2) $\mathrm{pre}(\Omega, f)_{\mathbb{X}} = \mathrm{pre}(\Omega^+, f)^0_{\mathbb{X}}$

Problem of finding the nominal one-step controllable set in step 2 of the algorithm is in most cases of negligible computational complexity compared to the step 1 where the Minkowski set difference needs to be computed. Step 2 takes only several Linear Programs (LP) and one projection computation.

Algorithm 3 ([10]): **One-step robust pre-set**

1) $\Omega^+ = \Omega \ominus \mathcal{W}$
2) $\mathrm{pre}(\Omega, f^{\rm cl})_{\mathbb{X}} = \mathrm{pre}(\Omega^+, f^{\rm cl})^0_{\mathbb{X}}$

Remark 1: Note that the resulting P-collection obtained in step 2 of Algorithm 3 is non-overlapping, while the P-collection obtained at step 2 of Algorithm 2 may in general contain overlaps which possibly need to be eliminated for simplification of further computations with the obtained set.

IV. Efficient Computation of the One-step Robust Sets

Since Minkowski set difference between a P-collection and a polytope is computationally demanding, we propose a procedure to avoid its explicit computation, aiming at the simplification of the computationally demanding set differentiation in step 5 of Algorithm 1.

For the Minkowski set difference $\mathcal{P} \ominus \mathcal{Q}$, where $\mathcal{P}$ is a P-collection and $\mathcal{Q}$ a polytope we define an admissible polytope $\mathcal{P}^a$ and a forbidden P-collection $\mathcal{P}^f$ such that $\mathcal{P} \ominus \mathcal{Q} = \mathcal{P}^a \setminus \mathcal{P}^f$. Sets $\mathcal{P}^a$ and $\mathcal{P}^f$ are obtained by running steps 1-4 of Algorithm 1. The proposed procedure for one-step robustly controllable set computation is given in the following algorithm.

Algorithm 4: **One-step robustly controllable set**

1) run steps 1–4 of Algorithm 1 for computing $\Omega \ominus \mathcal{W}$. Let $\Omega^a = \mathcal{R}_2$, $\Omega^f = \mathcal{R}_4 = \bigcup_{j=1}^{N^f} \Omega_j^f$, where Ω^a and Ω_j^f, $j = 1, \dots, N^f$ are polytopes with the property
$$\Omega^a \setminus \Omega^f = \Omega \ominus \mathcal{W}; \tag{11}$$
2) for each dynamics $i = 1, \dots, s$ compute
$$\begin{aligned} \Omega^{a+}_{xu,i} &= \{[x^T \ u^T]^T \in \mathcal{D}_i | \\ &\quad x \in \mathbb{X}, \ A_i x + B_i u + f_i \in \Omega^a\}, \\ \Omega^{f+}_{xu,i,j} &= \{[x^T \ u^T]^T \in \mathcal{D}_i | \\ &\quad x \in \mathbb{X}, \ A_i x + B_i u + f_i \in \Omega_j^f\}, \\ j &= 1, \dots, N^f, \\ \Omega^{f+}_{xu,i} &= \textstyle\bigcup_{j=1}^{N^f} \Omega^f_{xu,i,j}; \end{aligned}$$
3) for each dynamics $i = 1, \dots, s$ compute
$$\Psi_{xu,i} = \Omega^{a+}_{xu,i} \setminus \Omega^{f+}_{xu,i} \tag{12}$$
to finally obtain
$$\Psi_{xu} = \bigcup_{i=1}^{s} \Psi_{xu,i} \tag{13}$$
$$\Psi = \mathrm{proj}_{\mathbb{R}^n} \Psi_{xu} \tag{14}$$

In the following lemma we prove that the results of Algorithm 2 and Algorithm 4 are the same.

Lemma 1:

$$\Psi = \mathrm{pre}(\Omega, f)_{\mathbb{X}}. \tag{15}$$

Proof: Since regions $\mathcal{D}_i$ do not intersect, polytope $\Omega^{a+}_{xu,i}$ does not intersect $\Omega^{f+}_{xu,j}$ for $i \neq j$ and the following holds:

$$\Psi_{xu} = \bigcup_{i=1}^{s} \Omega^{a+}_{xu,i} \setminus \Omega^{f+}_{xu,i} = \bigcup_{i=1}^{s} \Omega^{a+}_{xu,i} \setminus \bigcup_{i=1}^{s} \Omega^{f+}_{xu,i}.$$

From respective definitions we see that

$$\begin{array}{l} \bigcup_{i=1}^{s} \Omega^{a+}_{xu,i} \setminus \bigcup_{i=1}^{s} \Omega^{f+}_{xu,i} = \\ \{[x^T \ u^T]^T |\ x \in \mathbb{X},\ f(x,u,0) \in \Omega^a\} \setminus \\ \setminus \{[x^T \ u^T]^T |\ x \in \mathbb{X},\ f(x,u,0) \in \Omega^f\} = \\ \{[x^T \ u^T]^T |\ x \in \mathbb{X},\ f(x,u,0) \in \Omega^a \setminus \Omega^f\} = \\ \{[x^T \ u^T]^T |\ x \in \mathbb{X},\ f(x,u,0) \in \Omega^+\}. \end{array}$$

$$\begin{array}{l} \Psi = \mathrm{proj}_{\mathbb{R}^n}(\Psi_{xu}) = \\ \{x \in \mathbb{X} |\ \exists u\ f(x,u,0) \in \Omega^+\} = \mathrm{pre}(\Omega, f)_{\mathbb{X}}. \end{array}$$

□

Analogously we pose the algorithm for computation of the one-step robust pre-set.

Algorithm 5: **One-step robust pre-set**

1) execute step 1 of Algorithm 4
2) for each dynamics $i = 1, \ldots, s^{\mathrm{cl}}$ compute

$$\Omega^{a+}_i = \{x \in \mathcal{D}^{\mathrm{cl}}_i \cap \mathbb{X} \mid A^{\mathrm{cl}}_i x + f^{\mathrm{cl}}_i \in \Omega^a\}, \tag{16}$$

$$\Omega^{f+}_{i,j} = \{x \in \mathcal{D}^{\mathrm{cl}}_i \cap \mathbb{X} \mid A^{\mathrm{cl}}_i x + f^{\mathrm{cl}}_i \in \Omega^f_j\}, j = 1, \ldots, N^f \tag{17}$$

$$\Omega^{f+}_i = \bigcup_{j=1}^{N^f} \Omega^{f+}_{i,j}; \tag{18}$$

3) for each dynamics $i = 1, \ldots, s^{\mathrm{cl}}$ compute

$$\Xi_i = \Omega^{a+}_i \setminus \Omega^{f+}_i \tag{19}$$

to finally obtain

$$\Xi = \bigcup_{i=1}^{s^{\mathrm{cl}}} \Xi_i; \tag{20}$$

Similarly as before we can show that the following holds.

Lemma 2:

$$\Xi = \mathrm{pre}(\Omega, f^{\mathrm{cl}})_{\mathbb{X}}. \tag{21}$$

Proof: Analogous to the proof of Lemma 1.

Now, let us compare the computational complexities of Algorithm 3 and Algorithm 5. Here the comparison is analogous, but somewhat simpler than between Algorithm 2 and Algorithm 4. Given Ω and $\mathcal{W}$, both algorithms enter Algorithm 1 for Minkowski set difference calculation. Algorithm 3 passes all the steps 1-5 to compute $\Omega \ominus \mathcal{W}$ while Algorithm 5 computes Ω^a in step 2 and Ω^f in step 4 of Algorithm 1.

Consider now the situation after step 4 of Algorithm 1 when Ω^a and Ω^f are computed. Until that moment, both algorithms run identically and after it they discourse. Ω^a is a polytope, and Ω^f a P-collection consisting of polytopes Ω^f_j, $j = 1, \ldots, N^f$, all the polytopes are in the minimal representation:

$$\Omega^a = \{x |\ H_{\Omega^a} x \leq K_{\Omega^a}\}, \tag{22}$$

$$\Omega^f_j = \{x |\ H_{\Omega^f_j} x \leq K_{\Omega^f_j}\}. \tag{23}$$

For the set difference $\Omega^a \setminus \Omega^f$ suppose that there are c active constraints among those defining Ω^f_j. The worst-case complexity of the set difference algorithm in step 5 of Algorithm 1 is thus $\mathcal{O}_1\left(\left(\frac{c}{N^f}\right)^{N^f}\right)$. Moreover, in the worst case the number of polytopes in the resulting P-collection $\Omega \ominus \mathcal{W} = \Omega^a \setminus \Omega^f$ is proportional to c^n, where n is the dimension of the state-space. Finally, to compute the set $\mathrm{pre}(\Omega, f^{\mathrm{cl}})_{\mathbb{X}}$, one needs to find the nominal one-step pre-set for $\Omega \ominus \mathcal{W}$ which is generally not problematic, but still the number of polyhedra in $\Omega \ominus \mathcal{W}$ could be large and complicate the computation. Thus, the worst-case computational complexity of Algorithm 3 after the proposed and the existing algorithm discourse is

$$\mathcal{O}_1\left(\left(\frac{c}{N^f}\right)^{N^f}\right) + s^{\mathrm{cl}} \mathcal{O}_3\left(c^n\right), \tag{24}$$

where $\mathcal{O}_3(a)$ denotes the complexity of computation of the nominal one-step pre-set of a polyhedra for an affine model.

After finishing step 4 of Algorithm 1, the proposed algorithm proceeds directly to compute the nominal one-step pre-sets Ω^{a+} and Ω^{f+} . According to (16) and (17), for each dynamics i polytopes Ω^{a+}_i and $\Omega^{f+}_{i,j}$, $j = 1, \ldots, N^f$ are computed at the cost of $\mathcal{O}_3(N^f + 1)$ number of LPs:

$$\begin{array}{l} \Omega^{a+}_i = \left\{x |\ \begin{bmatrix} H^{\mathrm{cl}}_i \\ H_{\Omega^a} A^{\mathrm{cl}}_i \end{bmatrix} x \leq \begin{bmatrix} K^{\mathrm{cl}}_i \\ K_{\Omega^a} - H_{\Omega^a} f^{\mathrm{cl}}_i \end{bmatrix}\right\}, \\ \Omega^{f+}_{i,j} = \left\{x |\ \begin{bmatrix} H^{\mathrm{cl}}_i \\ H_{\Omega^f_j} A^{\mathrm{cl}}_i \end{bmatrix} x \leq \begin{bmatrix} K^{\mathrm{cl}}_i \\ K_{\Omega^f_j} - H_{\Omega^f_j} f^{\mathrm{cl}}_i \end{bmatrix}\right\}, \\ \Omega^{f+}_i = \bigcup_{j=1}^{N^f} \Omega^{f+}_{i,j}, \end{array}$$

and the set $\mathrm{pre}(\Omega, f^{\mathrm{cl}})_{\mathbb{X}}$ is according to Lemma 1:

$$\mathrm{pre}(\Omega, f^{\mathrm{cl}})_{\mathbb{X}} = \bigcup_{i=1}^{s^{\mathrm{cl}}} \Omega^{a+}_i \setminus \Omega^{f+}_i. \tag{25}$$

In the new algorithm after the discourse s^{cl} polytope–P-collection set differences need to be computed to finally obtain $\mathrm{pre}(\Omega, f^{\mathrm{cl}})_{\mathbb{X}}$ as their union. It is trivial to show that for the number of full-dimensional $\Omega^{f+}_{i,j}$, N^{f+}_i, the following holds:

$$N^{f+}_i \leq N^f,\ \forall i. \tag{26}$$

For the number of active constraints in the i-th set difference of (25), c_i, the following lemma holds for each i.

Lemma 3:

$$c_i \leq c. \tag{27}$$

Proof: Consider relations (22) and (23) defining polytopes Ω^a and Ω^f_j. Recall that all H-representations are minimal. Without loss of generality, suppose that $\Omega^f_j \subseteq \Omega^a$

(i.e. every Ω_j^f is already intersected with Ω^a), such that active constraints are those constraints in Ω_j^f, $j = 1, \ldots, N^f$ that are not equal to any constraints defining Ω^a. Their overall number is c. Now consider the number of active constraints in $\Omega_i^{a+} \setminus \Omega_i^{f+}$. Note that $\Omega_j^f \subseteq \Omega^a$ causes

$$\Omega_{i,j}^{f+} \subseteq \Omega_i^{a+}, \ \forall j. \tag{28}$$

In the following with $[H, K]$ we denote the set of constraints in $Hx \leq K$. None of the constraints in $[H_i^{\mathrm{cl}}, K_i^{\mathrm{cl}}]$ cannot be active for $\Omega_{i,j}^{f+}$. Namely, since those constraints are common for $\Omega_{i,j}^{f+}$ and Ω_i^{a+}, those that are redundant in describing Ω_i^{a+} are definitely redundant for $\Omega_{i,j}^{f+}$ due to the subset property given in (28). Following this, only those constraints in $[H_i^{\mathrm{cl}}, K_i^{\mathrm{cl}}]$ that enter the Ω_i^{a+} minimal description may enter $\Omega_{i,j}^{f+}$ minimal description – none of them is thus active.

Now consider constraints in $[H_{\Omega_j^f} A_i^{\mathrm{cl}}, K_{\Omega_j^f} - H_{\Omega_j^f} f_i^{\mathrm{cl}}]$. For $\Omega^a \backslash \Omega^f$, c constraints among $[H_{\Omega_j^f}, K_{\Omega_j^f}]$, $j = 1, \ldots, N^f$, are different compared to $[H_{\Omega^a}, K_{\Omega^a}]$ (active constraints). Now, both $[H_{\Omega^a}, K_{\Omega^a}]$ and $[H_{\Omega_j^f}, K_{\Omega_j^f}]$ are transformed using the same transformation with A_i^{cl} and f_i^{cl}, and thus again exactly c of them are different. Some of those c constraints in $[H_{\Omega_j^f} A_i^{\mathrm{cl}}, K_{\Omega_j^f} - H_{\Omega_j^f} f_i^{\mathrm{cl}}]$ may have become redundant making c_i drop. Additionally those constraints from $[H_{\Omega^a}, K_{\Omega^a}]$ that were equal to some constraints in $[H_{\Omega_j^f}, K_{\Omega_j^f}]$ and that became redundant after the transformation are definitely redundant in $[H_{\Omega_j^{f+}}, K_{\Omega_j^{f+}}]$ due to the subset property (28) and thus c_i cannot become larger than c. □

The worst-case complexity of the proposed algorithm after the discourse is

$$s^{\mathrm{cl}} \mathcal{O}_3 \left(N^f + 1\right) + \sum_{i=1}^{s^{\mathrm{cl}}} \mathcal{O}_1 \left(\left(\frac{c_i}{N_i^{f+}} \right)^{N_i^{f+}} \right) \tag{29}$$

and is thus similar to the old algorithm if all c_i equal c and all N_i^{f+} equal N^f. Still, our motivation is that this rarely happens, especially in computationally demanding higher-order problems with bigger s^{cl}, since for certain dynamics i (i) many of the regions $\Omega_{i,j}^{f+}$ are empty, i.e. Ω_j^f are not one-step reachable using dynamics i making both N_i^{f+} and c_i drop and (ii) some active constraints in Ω_j^f become redundant after their linear transformation using A_i^{cl} and f_i^{cl}. The strategy we propose has thus a 'divide-and-conquer' nature where the problem of finding a set difference in target sets is moved to nominal pre-sets and divided by dynamics with benefits in number of polytopes and active constraints in those set differences.

Similar computational benefits may be observed in Algorithm 4 compared to 2. It should only be noted that the set differences between pre-sets are this time for the proposed algorithm carried out in $\mathbb{R}^{n+m}$ and for the existing one in $\mathbb{R}^n$. Analogous claims on the dropping number of regions N_i^{f+} and active constraints c_i still hold.

V. Application to the Robust Invariant Set Computation – A Comparative study

The computation of the one-step robust set is the atomic part of the iterative procedure to compute a robust invariant set for the given DTPWA system. The computation of maximal robust controlled invariant set inside a predefined set $\mathbb{X} \subset \mathbb{R}^n$ [4] for the system f given in (4) follows [10].

Algorithm 6: **Maximal robust controlled invariant set inside** $\mathbb{X}$

1) $\Omega_0 = \mathbb{X}$, $k = 0$;
2) $\Omega_{k+1} = \mathrm{pre}(\Omega_k, f)_{\mathbb{X}}$;
3) if $\Omega_k = \Omega_{k+1}$ return Ω_k, else $k = k + 1$ and goto 2.

For the given closed-loop DTPWA model the maximal robust positively invariant (MRPI) set inside a presumed set $\mathbb{X} \subset \mathbb{R}^n$ [4] for the system f^{cl} given in (6) can be computed in the following way.

Algorithm 7: **Maximal robust positively invariant set inside** $\mathbb{X}$

1) $\Omega_0 = \mathbb{X}$, $k = 0$;
2) $\Omega_{k+1} = \mathrm{pre}(\Omega_k, f^{\mathrm{cl}})_{\mathbb{X}}$;
3) if $\Omega_k = \Omega_{k+1}$ return Ω_k, else $k = k + 1$ and goto 2.

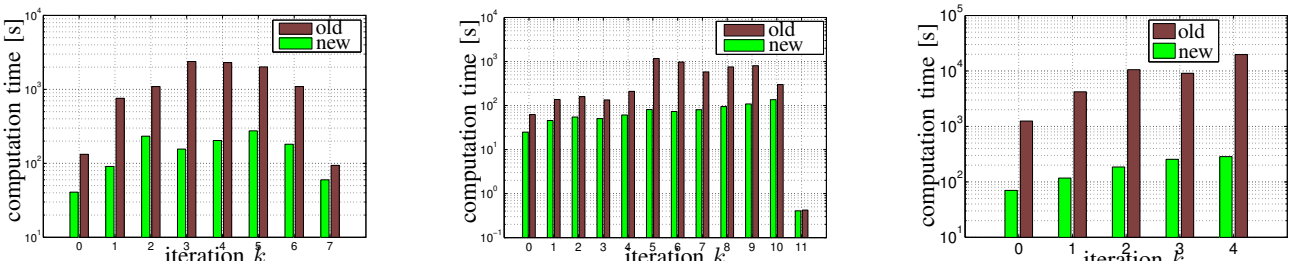
We compare the performances of the old and the proposed method for one-step robust pre-sets computation on several examples of the computation of the MRPI sets for autonomous PWA systems subject to additive uncertainty. The autonomous PWA system is formed in the following way. For the randomly generated PWA system in each dynamics an mpQP is solved to find a PWA control law [11] that minimizes the offset of the system state from the origin at the next sampling instant.

A. Example 1

We test the new and the existing method on the problem of computing the MRPI set for the autonomous DTPWA system. The computation times[2] for both methods for 3 randomly generated DTPWA systems in 3D (with 72, 65 and 90 affine dynamics, respectively) are reported in Figure 1.

DTPWA systems were generated in $\mathbb{X} = \{x \mid \|x\|_\infty \leq 10\}$ box. In examples 1.1. and 1.2. the uncertainty polytope was chosen $\|w\|_\infty \leq 0.5$ and the resulting MRPI set is an empty set. This explains why the computation times in the last iteration for those two examples are similar for both methods. In earlier iterations the computational gain (speedup) of the new algorithm is in the range from 2 to 10. In example 1.3. we use a smaller uncertainty polytope ($\|w\|_\infty \leq 0.05$) and the full-dimensional MRPI set is generated. Computational gain through iterations in this case ranges from 10 to 50. To get a better idea of the overall speedup for this particular example the MRPI set is computed within 15 minutes with the new method, while the old method requires more than 11 hours of computation.

[2]Pentium IV, 2.4 GHz, using MPT 2.5 with NAG LP solver within Matlab 7.1 on Windows XP.



(a) Example 1.1. (b) Example 1.2. (c) Example 1.3.

Fig. 1. MRPI set computation times through iterations for three randomly generated 3D PWA systems.

B. Example 2

We compare two methods of the one-step robust set computation on a 4D DTPWA system consisting of 13 affine dynamics, with $\mathbb{X} = \{x \mid \|x\|_\infty \leq 10\}$ and $\|w\|_\infty \leq 0.01$. Computation times are reported in Figure 2. The iterations with $k = 0$ have similar computation times since the initial partition $\mathbb{X}$ of the model is convex resulting in $\mathbb{X} \ominus \mathcal{W}$ being a single polytope and $\Omega^f = \emptyset$. The data for the old method are not available for $k \geq 2$ because the algorithm did not finish computation with $k = 2$ even after 7 days. The new algorithm ended in about 13 minutes with $k = 3$, resulting in the MRPI set consisting of 115 polytopes. Clearly, the

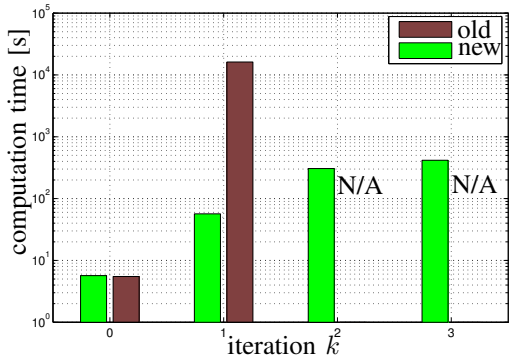


Fig. 2. Computation times in Example 2.

computational benefits of the new algorithm become more and more obvious as the problem gets more complicated, i.e. as the state-space dimension and/or the number of affine dynamics grow.

VI. Conclusion

We consider discrete-time piecewise affine (DTPWA) systems subject to polytope-bounded uncertainty. For such systems we propose an efficient procedure for computation of the one-step robust sets for target sets that can be represented as union of polytopes (P-collection). Unlike the existing algorithm our procedure avoids explicit computation of the Minkowski set difference between a P-collection and a polytope – a computationally demanding task comprising two polytope–P-collection set difference computations in a row. The worst-case computational complexities of the existing and the new method are comparable. However, we show that the number of polytopes and active constraints for the second set difference may only decrease with the proposed method, thus making it superior in practice. The computational benefits get more and more obvious as the state-space dimension and/or the number of affine dynamics in the model get higher. We illustrate this by comparing both methods on several examples of the maximal robust positively invariant set computation for randomly generated DTPWA systems.

References

[1] E. D. Sontag, "Nonlinear regulation: The piecewise linear approach," *IEEE Transactions on Automatic Control*, vol. 26, no. 2, pp. 346–358, April 1981.

[2] W. P. M. H. Heemels, B. De Schutter, and A. Bemporad, "Equivalence of hybrid dynamical models," *Automatica*, vol. 37, no. 7, pp. 1085–1091, July 2001.

[3] M. Kvasnica, P. Grieder, M. Baotić, and M. Morari, *Multi-Parametric Toolbox (MPT)*, 2004. [Online]. Available: http://control.ee.ethz.ch/~mpt/

[4] F. Blanchini, "Set invarinace in control — A survey," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[5] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, June 2000.

[6] E. C. Kerrigan, "Robust constraint satisfaction: Invariant sets and predictive control," Ph.D. dissertation, Department of Engineering, University of Cambridge, UK, Trumpington Street, Cambridge CB2 1PZ, UK, 2000.

[7] D. P. Bertsekas, "Infinite-time reachability of state-space regions by using feedback control," *IEEE Transactions on Automatic Control*, vol. 17, no. 5, pp. 604–613, May 1972.

[8] S. V. Raković, E. C. Kerrigan, and D. Q. Mayne, "Reachability computations for constrained discrete-time systems with state- and input-dependent disturbances," in *Proceedings of the 42nd IEEE Conference on Decisionm and Control*, Maui, Hawaii, USA, December 2003, pp. 3905–3910.

[9] S. V. Raković, E. C. Kerrigan, D. Q. Mayne, and J. Lygeros, "Reachability analysis of discrete-time systems with disturbances," *IEEE Transactions on Automatic Control*, vol. 51, no. 4, pp. 546–561, April 2006.

[10] S. V. Raković, P. Grieder, M. Kvasnica, D. Q. Mayne, and M. Morari, "Computation of invariant sets for piecewise affine discrete time systems subject to bounded disturbances," *Proceedings of the 43rd IEEE Conference on Decision and Control, Atlantis, Bahamas*, pp. 1418–1423, December 2004.

[11] M. Baotić, "Optimal Control of Piecewise Affine Systems – a Multi-parametric Approach," Ph.D. dissertation, Automatic Control Laboratory, Swiss Federal Institute of Technology, Zurich, Switzerland, Physikstrasse 3, CH-8092 Zurich, Mar. 2005. [Online]. Available: http://control.ee.ethz.ch/index.cgi?page=publications