Coupling Policy Iterations with Piecewise Quadratic Lyapunov Functions

Assalé Adjé
Université of Perpignan Via Domitia
LAMPS
France
assale.adje@univ-perp.fr

ABSTRACT

We recently constructed piecewise quadratic Lyapunov functions to compute overapproximations over the reachable values set of piecewise affine discrete-time systems. The overapproximations can be viewed as the solutions of an inverse problem. However these overapproximations can be loose. In this paper, we refine the latter overapproximations extending previous works combining policy iterations with quadratic Lyapunov functions.

1. INTRODUCTION

Several catastrophic events showed the importance of the formal verification of programs. Some of these failures are caused by overflows. A method to prove the absence of overflows in numerical programs consists in providing precise safe bounds over the values taken by the variable of the analyzed program.

In this paper, we are interesting in a particular class of numerical programs: single while loop programs with a switch-case structure inside the loop body. Moreover, we suppose that test and assignment functions are affine. Such a program can be represented as a piecewise affine discrete-time system. To compute bouns over the values taken by the variable of the analyzed program is thus reduced to overapproximate the reachable values set of a piecewise affine discrete-time system. Hence, we propose to compute *automatically* precise bounds over piecewise affine discrete-time systems using policy iterations and piecewise quadratic Lyapunov functions.

Initially the policy iterations algorithm solves stochastic control problems [16] which are equivalent to solve fixed point problems involving maxima of affine functions. The policy iteration algorithm was then extended to zero-sum two-player stochastic games [15], this extension allows the computation of the unique fixed point of a contractive piecewise affine function. The very first extension of the policy iterations algorithm in program analysis was in 2005 by Costan et al [9]. Since then, the use of policy iterations in various verification problems greatly increases: in [3, 14], the authors describe a policy iteration algorithm to overapproximate the reachable values set of numerical programs with affine assignments; in [19], the author proves termination using policy iterations; in [26, 28] the authors propose to embed policy iterations for programs dealing with both numerical and boolean variables.

The method developed in [2] allows to compute bounds over the state-variable of a piecewise affine system. The method relies on the synthesis of a piecewise quadratic Lyapunov function for the considered system. The optimal value of the formulated maximization problem furnishes an upper bound on the maximal value of the Euclidian norm of the state variable. This upper bound can be very loose since it combines all the coordinates together. To obtain tigher results, we propose to use a templates based method. A templates method consists in representing sets as sublevel sets of *given* functions called *templates*. Then an overapproximation in this context is computed from a vector of bounds over the templates. The most precise overapproximation with respect to these templates is provided by the vector of bounds satisfying a smallest fixed point equation. In our context, the generated piecewise quadratic Lyapunov function is used as a template. We complete the templates basis by the square of variables. Finally we use policy iterations to solve the (smallest) fixed point equation. Thus, the developed policy iterations algorithm leads to tighter bounds over the reachable values set.

Related Works. The use of a quadratic Lyapunov function as a quadratic template was explicitly done in [25] but one quadratic Lyapunov function is not sufficient to prove the boundedness of reachable values set of a piecewise affine system unless it exists a common quadratic Lyapunov function. Then, we deal with piecewise quadratic Lyapunov functions. Hence the works on the computation of piecewise quadratic Lyapunov functions [11, 12, 17, 21] are also related to this paper. Their authors are interested in proving stability of piecewise affine systems. However, as classical quadratic Lyapunov functions, piecewise quadratic Lyapunov functions provide sublevel invariant sets for the considered system. We use this latter interpretation for a verification purpose to compute an overapproximation of the reachable values set.

In this paper, we aboard the reachability problem as we are interested in reachable values sets. In [6], the authors consider the complexity of some reachability problem such as: Is a certain value can be attained? When this value is at-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

HSCC'17, April 18-20, 2017, Pittsburgh, PA, USA

© 2017 ACM. ISBN 978-1-4503-4590-3/17/04...$15.00

DOI: http://dx.doi.org/10.1145/3049797.3049825

tained?...In our work, we do not consider complexity issues but we want to represent a reachable value set in a computable way. So we avoid complexity issues by using computable representations which present some conservatism. We can cite the work of Rakovic et al [23]. The authors propose to deal with set invariance and present some results about sets that are invariant according to certain piecewise affine dynamics and some control. In our work, we are interesting in computing invariants set that contain initial conditions and thus the whole reachable value sets.

Another interesting tool could be the tropical polyhedra domain [4]. It generates disjunctions of zones as invariants. Nevertheless, the latter invariants did not encode quadratic relations between variables.

The works about verifying hybrid systems (see [27] and references therein) could give us some inspirations. First, the studies concern continuous dynamics whereas we are interested in discrete-time systems. Up to some adaptation, this first inconvenient could be overcomed. Second, tools to study safety problems are mainly limited to a bounded time analysis and perform a huge number of steps to obtain a sufficient precision. We propose to compute a safe guaranteed overapproximation which is valid whatever the time in a small number of iterations.

Policy iteration algorithms in templates domain proposed in [3] used quadratic templates and did not handle piecewise quadratic templates. In this paper, we adapt policy iteration based on Lagrange duality [1] to piecewise quadratic functions.

Contributions. The first contribution of the paper is the formalisation of piecewise quadratic Lyapunov functions to compute an overapproximation of the reachable values set of a piecewise affine discrete-time dynamical system. This formalisation uses the theory of cone-copositive matrices which is also an original contribution in this context.

The main contribution of the article is the extension of the policy iterations algorithm to the piecewise quadratic templates. Indeed, a policy iterations algorithm has just been constructed in the case of quadratic functions.

Notations. Numbers. $\mathbb{N}$ denotes the set of nonnegative integers, then for $d \in \mathbb{N}$, $[d] = \{1, \ldots, d\}$. $\mathbb{R}$ is the set of reals, $\mathbb{R}_+$ the set of nonnegative reals and $\mathbb{R}^d$ denotes the set of vectors of d reals. We denote by $\wp(\mathbb{R}^d)$ the set of subsets of $\mathbb{R}^d$.

Inequalities. For $y, z \in \mathbb{R}^d$, $y < z$ (resp. $y \leq z$) means $\forall\, l \in [d]$, $y_l < z_l$, (resp. $\forall\, l \in [d]$, $y_l \leq z_l$) and $y \leq_{w,s} z$ is a mix of weak and strict inequalities.

Matrices. $\mathbb{M}_{n\times m}$ is the set of matrices with n rows and m columns. $0_{n,m}$ and 0_n are respectively the null matrices of $\mathbb{M}_{n\times m}$ and $\mathbb{M}_{n\times n}$. Id_n is the identity matrix of $\mathbb{M}_{n\times n}$. $M^\intercal$ is the transpose of $M \in \mathbb{M}_{n\times m}$. $\mathbb{S}_n$ is the set of symmetric matrices of size $n \times n$. $A \succeq 0$ means that A is semi-definite positive i.e. $A \in \mathbb{S}_d$ and $\forall\, x \in \mathbb{R}^d$, $x^\intercal A x \geq 0$. $\mathbb{S}_d^+$ is the convex cone of semidefinite positive matrices.

2. PIECEWISE AFFINE DISCRETE-TIME SYSTEMS

Our very first motivation is the verification of programs. Indeed, we want to prove that the values taken by the variable of the program are bounded. To prove it, it suffices to compute bounds over the values taken by each variable. The considered programs consist of a single loop with possibly a complicated switch-case type loop body supposed to be written as a nested sequence of *if then else* (*ite* for short) statements, or a *switch* $c1 \to inst1; c2 \to instr2; c3 \to instr3$. Moreover, we assume that the analyzed programs are written in affine arithmetic: the assignments of the programs are of the form $v_j = \sum_{i=1}^{d} a_i v_i + c$ where a_i and c are scalars and v_j denotes a variable of the program. So, the programs analyzed here can be viewed as piecewise affine discrete-time systems as we can see at Example 1. In the rest of the paper, we only consider piecewise affine discrete-time systems. Finally, the verification problem boils down to compute an overapproximation of the reachable states of a piecewise affine discrete-time system.

Piecewise affine systems (PWA for short) are defined as systems the dynamic of which is piecewise affine. Thus the dynamic is characterized by a polyhedral partition and a family of affine maps relative to this partition. Here, a polyhedral partition is a family of convex polyhedra such that:

$$\bigcup_{i\in\mathcal{I}} X^i = \mathbb{R}^d \text{ and } \forall\, i, j \in \mathcal{I},\ i \neq j\ X^i \cap X^j = \emptyset\ . \qquad (1)$$

The convex polyhedron X^i can contain both strict and weak inequalities and is represented by $T^i \in \mathbb{M}_{n_i\times m}$ and $c^i \in \mathbb{R}^{n_i}$. We denote by T_s^i (resp. T_w^i) and c_s^i (resp. c_w^i) the parts of T^i and c^i corresponding to strict (resp. weak) inequalities:

$$\begin{aligned} X^i &= \left\{x \in \mathbb{R}^d \middle| T^i x \leq_{w,s} c^i\right\} \\ &= \left\{x \in \mathbb{R}^d \middle| T_s^i x < c_s^i,\ T_w^i x \leq c_w^i\right\} \end{aligned} \qquad (2)$$

Definition 1 (Piecewise Affine System). *A PWA is characterized by the triple $(X^0, \mathcal{X}, \mathcal{A})$ where:*

- *X^0 is the polytope of the initial conditions of the form* (2)*;*
- *$\mathcal{X} := \{X^i, i \in \mathcal{I}\}$ is a polyhedral partition i.e. satisfying* (1)*;*
- *$\mathcal{A} := \{x \mapsto f^i(x) = A^i x + b^i, i \in \mathcal{I}\}$ where $A^i \in \mathbb{M}_{d\times d}$ and $b^i \in \mathbb{R}^d$;*

And satisfies the following relation for all $k \in \mathbb{N}$:

$$x_0 \in X^0,\ \text{if } x_k \in X^i,\ x_{k+1} = f^i(x_k)\ . \qquad (3)$$

Let $P = (X^0, \mathcal{X}, \mathcal{A})$ be a PWA. We need some notations for the rest of the paper. First we define the reachable values set $\mathcal{R}$ of P:

$$\mathcal{R} := \bigcup_{k\in\mathbb{N}} \mathbb{A}^k(X^0),\ \text{where } \mathbb{A}(x) = f^i(x) \text{ if } x \in X^i \qquad (4)$$

We define the set of possible switches:

$$\begin{aligned} &\mathrm{Sw} := \{(i,j) \in \mathcal{I}^2 \mid \mathcal{R} \cap X^{ij} \neq \emptyset\} \\ &\text{where } X^{ij} = X^i \cap {f^i}^{-1}(X^j)\ . \end{aligned} \qquad (5)$$

Finally, we define the set of indices of polyhedra of $\mathcal{X}$ which meet the polyhedron of possible initial conditions:

$$\mathrm{In} := \{i \in \mathcal{I} \mid X^{i0} \neq \emptyset\} \text{ where } X^{i0} = X^i \cap X^0\ . \qquad (6)$$

We introduce for $i \in \mathcal{I}$, the following matrix of $\mathbb{M}_{(d+1)\times(d+1)}$:

$$F^i = \begin{pmatrix} 1 & 0_{1\times d} \\ b^i & A^i \end{pmatrix}\ . \qquad (7)$$

Eq. (3) can be rewritten as $(1, x_{k+1})^\intercal = F^i(1, x_k)$.

EXAMPLE 1. *Let us consider the following program : a single while loop with a ite instruction in the loop body.*

```
(x,y)∈ [0,3] × [0,2];
while(true){
    ox=x;
    oy=y;
    if (3*ox+8*oy < -3 ){
        x=0.4197*ox-0.2859*oy+2;
        y=-0.5029*ox+0.1679*oy+5;
    }
    else{ \\3*ox+8*oy>=-3
        x=-0.0575*ox-0.4275*oy-4;
        y=-0.3334*ox-2682*oy+4;
    }
}
```

The initial condition X^0 of the piecewise affine systems is $(x, y) \in [0, 3] \times [0, 2]$. We can rewrite this program as a piecewise affine discrete-time dynamical systems using our notations. To do so, we have to exhibit the matrices T_s^i and T_w^i and vectors c_s^i and c_w^i (see Eq. (2)) and the matrices F^i (see Eq. (7)):

$$F^1 = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 0.4197 & 0.2859 \\ 5 & 0.5029 & 0.1679 \end{pmatrix}, \begin{cases} T_s^1 = (3 \quad 8) \\ c_s^1 = -3 \end{cases}$$

$$F^2 = \begin{pmatrix} 1 & 0 & 0 \\ -4 & -0.0575 & 0.4275 \\ 4 & -0.3334 & -0.2682 \end{pmatrix}, \begin{cases} T_w^2 = (-3 \quad -8) \\ c_w^2 = 3 \end{cases}$$

We are interested in computing *automatically* precise over-approximation of $\mathcal{R}$. First, we propose to compute an over-approximation of $\mathcal{R}$ as a set $S \subseteq \mathbb{R}^d$ such that $X^0 \subseteq S$ and $\forall i \in \mathcal{I},\ x \in S \cap X^i \implies A^i x + b^i \in S$. From the latter invariance condition, a sublevel of a Lyapunov function containing the initial states can be such a set S.

From now on, we consider a fixed $P = (X^0, \mathcal{X}, \mathcal{A})$ following Def. 1.

3. WEAK PIECEWISE QUADRATIC LYAPUNOV FUNCTIONS

In this paper, we use weak piecewise quadratic Lyapunov functions to compute directly an overapproximation of reachable values set.

Let q be a quadratic form i.e. a function such that for all $y \in \mathbb{R}^d$, $q(y) = y^\intercal A_q y + b_q^\intercal y + c_q$ where $A_q \in \mathbb{S}_d$, $b_q \in \mathbb{R}^d$ and $c_q \in \mathbb{R}$. We define the lift-matrix of q, the matrix of $\mathbb{S}_{d+1}$ defined as follows:

$$\mathbf{M}(A_q, b_q, c_q) = \mathbf{M}(q) = \begin{pmatrix} c_q & (b_q/2)^\intercal \\ (b_q/2) & A_q \end{pmatrix} \tag{8}$$

It is obvious that the $q \mapsto \mathbf{M}(q)$ is linear. Let $A \in \mathbb{M}_{d\times d}$, $b \in \mathbb{R}^d$, and q be a quadratic form, we have, for all $x \in \mathbb{R}^d$:

$$q(Ax+b) = \begin{pmatrix} 1 \\ x \end{pmatrix}^\intercal \begin{pmatrix} 1 & 0_{1\times d} \\ b & A \end{pmatrix}^\intercal \mathbf{M}(q) \begin{pmatrix} 1 & 0_{1\times d} \\ b & A \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix} . \tag{9}$$

LEMMA 1. *Let $A \in \mathbb{S}_d$, $b \in \mathbb{R}^d$ and $c \in \mathbb{R}$. Then:* $(\forall y \in \mathbb{R}^d,\ y^\intercal Ay + b^\intercal y + c \geq 0) \iff \mathbf{M}(A, b, c) \in \mathbb{S}_{d+1}^+$

PROOF. It suffices to remark that for all $t \neq 0$, for all $y \in \mathbb{R}^d$, $t^2 q(t^{-1}y) = \begin{pmatrix} 1 \\ y \end{pmatrix}^\intercal M(q) \begin{pmatrix} 1 \\ y \end{pmatrix}$. $\square$

DEFINITION 2 ((CONE)-COPOSITIVE MATRICES). *Let $M \in \mathbb{M}_{m\times d}$. A matrix $Q \in \mathbb{S}_d$ is said to be M-copositive iff:*

$$My \geq 0 \implies y^\intercal Q y \geq 0$$

An Id_d-copositive matrix is called a copositive matrix. *We denote by $\mathbf{C}_d(M)$ the set of M-copositive matrices and $\mathbf{C}_d$ the set of copositive matrices.*

The notion of cone-copositive matrix involves conic polyhedra but can be easily extended to non-conic polyhedra by using $Mx \leq p \iff \begin{pmatrix} 1 & 0_{1\times d} \\ p & -M \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix} \geq 0$.

LEMMA 2. *Let $q : \mathbb{R}^d \to \mathbb{R}^d$ be a quadratic function. Let $M \in \mathbb{M}_{m\times d}$, $p \in \mathbb{R}^m$ and consider $C = \{x \mid Mx \leq p\}$. Then $\mathbf{M}(q) \in \mathbf{C}_{d+1}\left(\begin{pmatrix} 1 & 0_{1\times d} \\ p & -M \end{pmatrix}\right) \implies (q(x) \geq 0,\ \forall x \in C)$.*

We introduce the following matrices:

$$\forall\, i \in \mathcal{I},\ E^i = \begin{pmatrix} 1 & 0_{1\times d} \\ c^i & -T^i \end{pmatrix} , \tag{10a}$$

$$\forall\, (i,j) \in \mathcal{I}^2,\ E^{ij} = \begin{pmatrix} 1 & 0_{1\times d} \\ c^i & -T^i \\ c^j - T^j b^i & -T^j A^i \end{pmatrix} , \tag{10b}$$

$$\forall\, i \in \mathrm{In},\ E^{i0} = \begin{pmatrix} 1 & 0_{1\times d} \\ c^i & -T^i \\ c^0 & -T^0 \end{pmatrix} . \tag{10c}$$

We will denote by n_i the number of rows of E^i, n_{ij} the number of rows of E^{ij} and n_{i0} the number of rows of E^{i0}.

LEMMA 3. *For all $i \in \mathcal{I}$, $X^i \subseteq \{x \mid E^i(1\ x^\intercal)^\intercal \geq 0\}$, for all $(i,j) \in \mathrm{Sw}$, $X^{ij} \subseteq \{x \mid E^{ij}(1\ x^\intercal)^\intercal \geq 0\}$ and for all $i \in \mathrm{In}$, $X^{i0} \subseteq \{x \mid E^{i0}(1\ x^\intercal)^\intercal \geq 0\}$.*

DEFINITION 3 (WPQL FUNCTIONS). *A function L is a weak piecewise quadratic Lyapunov function (wPQL for short) for P if and only if there exist a family $\{(P^i, q^i), P^i \in \mathbb{S}_d, q^i \in \mathbb{R}^d,\ i \in \mathcal{I}\}$ and two reals α and β such that:*

1. *$\forall\, i \in \mathcal{I}$, $\forall\, x \in X^i$, $L(x) = L^i(x) = x^\intercal P^i x + 2x^\intercal q^i$;*
2. *$\forall\, i \in \mathcal{I}$:*

$$\mathbf{M}(P^i, 2q^i, -\alpha) - \mathbf{M}(\mathrm{Id}, 0, -\beta) \in \mathbf{C}_{d+1}\left(E^i\right) \ ; \tag{11}$$

3. *$\forall\, (i,j) \in \mathrm{Sw}$:*

$$\mathbf{M}(P^i, 2q^i, 0) - F^{i\intercal}\mathbf{M}(P^j, 2q^j, 0)F^i \in \mathbf{C}_{d+1}\left(E^{ij}\right) \ ; \tag{12}$$

4. *$\forall\, i \in \mathrm{In}$:*

$$-\mathbf{M}(P^i, 2q^i, -\alpha) \in \mathbf{C}_{d+1}\left(E^{i0}\right) . \tag{13}$$

PROPOSITION 1 (BOUNDED TRAJECTORIES). *Suppose that P admits a wPQL function represented by $\{(P^i, q^i), P^i \in \mathbb{S}_d, q^i \in \mathbb{R}^d,\ i \in \mathcal{I}\}$ and reals α and β. Let $i \in \mathcal{I}$, $S_\alpha^i = \{x \in X^i \mid L^i(x) \leq \alpha\} = \{x \in X^i \mid x^\intercal P^i x + 2x^\intercal q^i \leq \alpha\}$ and $S = \cup_{i\in\mathcal{I}} S_\alpha^i$. Then, $\mathcal{R} \subseteq S \subseteq \{x \in \mathbb{R}^d \mid \|x\|_2^2 \leq \beta\}$.*

PROOF. First, we prove that $S \subseteq \{x \in \mathbb{R}^d \mid \|x\|_2^2 \leq \beta\}$. Let $i \in \mathcal{I}$ and $x \in X^i$. From Eq. (11), Lemma 2 and Lemma 3, $x^\intercal P^i x + 2x^\intercal q^i - \alpha - \|x\|_2^2 + \beta \geq 0$ which implies that $S \subseteq \{x \in \mathbb{R}^d \mid \|x\|_2^2 \leq \beta\}$.

From Eq. (4), to prove the first inclusion, we have to show that for all $k \in \mathbb{N}$, $\mathbb{A}^k(X^0) \subseteq S$. We prove it by induction on k. Let $x \in X^0$. Since $\mathcal{X}$ satisfies (1), there exists a unique $i \in \mathrm{In}$ such that $x \in X^{i0}$. From Eq. (13), Lemma 2 and Lemma 3, $L^i(x) \leq \alpha$. Now suppose $\mathbb{A}^k(X^0) \subseteq S$ for some $k \in \mathbb{N}$. Let $y \in \mathbb{A}^{k+1}(X^0)$. Then $y = \mathbb{A}(x)$ for some $x \in \mathbb{A}^k(X^0)$. Since $\mathcal{X}$ satisfies (1), there exists an unique $(i,j) \in \mathrm{Sw}$ such that $x \in X^{ij}$ (hence $y \in X^j$). As $x \in X^i$ and $x \in S$, then $x \in S^i_\alpha$. From Eq. (12), Lemma 2 and Lemma 3, $0 \leq L^i(x) - L^j(y) = L^i(x) - \alpha - (L^j(y) - \alpha)$. Finally $y \in S^j_\alpha \subseteq S$ as $x \in S^i_\alpha$. $\square$

3.1 Computational issues

To construct wPQL functions, we are faced with two issues. First, we must know the sets of indices Sw and In. Second we have to manipulate cone-copositive constraints.

3.1.1 The computation of sets Sw and In

To set Sw relies on $\mathcal{R}$, the set we want approximate. To overcome this issue, we just remove the intersection with $\mathcal{R}$:

$$\overline{\mathrm{Sw}} := \{(i,j) \in \mathcal{I}^2 \mid X^{ij} \neq \emptyset\} \ . \tag{14}$$

The polyhedra X^i and X^j can contain strict inequalities. Hence to compute $\overline{\mathrm{Sw}}$ we need Motzkin's theorem [22]. This alternative theorem permits to compute exactly the set In. The direct application of Motzkin's transposition theorem [22] yields to the next proposition.

PROPOSITION 2. *The couple $(i,j) \in \overline{\mathrm{Sw}}$ if and only if:*

$$\begin{cases} \begin{pmatrix} 1 & 0_{1\times d} \\ c_s^i & -T_s^i \\ c_s^j - T_s^j b^i & -T_s^j A^i \end{pmatrix}^\intercal p^s + \begin{pmatrix} c_w^i & -T_w^i \\ c_w^j - T_w^j b^i & -T_w^j \end{pmatrix}^\intercal p = 0 \\ \displaystyle\sum_k p_k^s = 1,\ p^s \geq 0,\ p \geq 0 \end{cases}$$

has no solution.

The index $i \in \mathrm{In}$ if and only if:

$$\begin{cases} \begin{pmatrix} 1 & 0_{1\times d} \\ c_s^i & -T_s^i \\ c_s^0 & -T_s^0 \end{pmatrix}^\intercal p^s + \begin{pmatrix} c_w^i & -T_w^i \\ c_w^0 & -T_w^0 \end{pmatrix}^\intercal p = 0 \\ \displaystyle\sum_k p_k^s = 1,\ p^s \geq 0,\ p \geq 0 \end{cases}$$

has no solution.

3.1.2 Cone-copositive constraints

The interested reader can refer to [7] a list of exciting papers about the representation of cone-copositive matrices.

PROPOSITION 3 (TH. 2.1 OF [18]). *For all $M \in \mathbb{M}_{m\times d}$, we have:*

$$\{M^\intercal C M + S \mid C \in \mathbf{C}_d \text{ and } S \in \mathbb{S}_d^+\} \subseteq \mathbf{C}_d(M) \tag{Δ}$$

If the rank of M is equal to m, then (Δ) is an equality.

The next proposition discusses a simple characterization of copositive matrices.

PROPOSITION 4 ([10, 20]). *We have: $\forall\, d \in \mathbb{N}$: $\mathbb{S}_d^{\geq 0} + \mathbb{S}_d^+ \subseteq \mathbf{C}_d$. If $d \leq 4$ then $\mathbf{C}_d = \mathbb{S}_d^{\geq 0} + \mathbb{S}_d^+$.*

COROLLARY 1. *Let $M \in \mathbb{M}_{m\times d}$. Then:*

$$\mathbf{C}_d(M) \supseteq \left\{ Q \in \mathbb{S}_d \;\middle|\; \begin{array}{c} \exists\, W_p \in \mathbb{S}_m^{\geq 0},\ W_+ \in \mathbb{S}_m^+,\ \text{s.t.} \\ Q - M^\intercal (W_p + W_+) M \succeq 0 \end{array} \right\} \tag{$\star$}$$

If M has full row rank and $d \leq 4$, then ($\star$) is actually an equality.

The computation of copositive constraints is a quite recent field of research. Algorithms exist (e.g. [8]) but for the knowledge of the author no tools are available. In this paper, in practice, we use Corollary 1 and we replace $\mathbf{C}_d(M)$ by the right-hand side of Eq. ($\star$).

3.1.3 Computation of weak piecewise quadratic Lyapunov functions using SDP solvers

Finally, we construct wPQL functions using semidefinite programming. We define the notion of computable wPQL functions.

DEFINITION 4 (COMPUTABLE WPQL FUNCTIONS). *A function L is a computable wPQL for P if and only if there exist two reals α and β and four families:*

- $\mathcal{P} := \{(P^i, q^i), P^i \in \mathbb{S}_d, q^i \in \mathbb{R}^d,\ i \in \mathcal{I}\}$
- $\mathcal{W} := \{\left(W_p^i, W_+^i\right) \in \mathbb{S}_{n_i+1}^{\geq 0} \times \mathbb{S}_{n_i+1}^+, i \in \mathcal{I}\}$,
- $\mathcal{U} := \{\left(U_p^{ij}, U_+^{ij}\right) \in \mathbb{S}_{n_{ij}}^{\geq 0} \times \mathbb{S}_{n_{ij}}^+, (i,j) \in \overline{\mathrm{Sw}}\}$
- $\mathcal{Z} := \{\left(Z_p^{i0}, Z_+^{i0}\right) \in \mathbb{S}_{n_{i0}}^{\geq 0} \times \mathbb{S}_{n_{i0}}^+, i \in \mathrm{In}\}$

such that:

1. *$\forall\, i \in \mathcal{I}$, $\forall\, x \in X^i$, $L(x) = L^i(x) = x^\intercal P^i x + 2x^\intercal q^i$;*
2. *$\forall\, i \in \mathcal{I}$:*

$$\begin{aligned} \mathbf{M}(P^i, 2q^i, -\alpha) - \mathbf{M}(\mathrm{Id}, 0, -\beta) & \\ - E^{i\intercal}\left(W_p^i + W_+^i\right) E^i & \succeq 0 \ ; \end{aligned} \tag{15}$$

3. *$\forall\, (i,j) \in \overline{\mathrm{Sw}}$:*

$$\begin{aligned} \mathbf{M}(P^i, 2q^i, 0) - F^{i\intercal}\mathbf{M}(P^j, 2q^j, 0) F^i & \\ - E^{ij\intercal}\left(U_p^{ij} + U_+^{ij}\right) E^{ij} & \succeq 0 \ ; \end{aligned} \tag{16}$$

4. *$\forall\, i \in \mathrm{In}$:*

$$-\mathbf{M}(P^i, 2q^i, -\alpha) - E^{i0\intercal}\left(Z_p^{0i} + Z_+^{0i}\right) E^{i0} \succeq 0; \tag{17}$$

Let us consider the problem:

$$\begin{aligned} \inf_{\substack{\mathcal{P},\mathcal{W},\mathcal{U},\mathcal{Z},\\ \alpha,\beta}} \quad & \alpha + \beta \\ \text{s.t.} \quad & \begin{cases} (\mathcal{P},\mathcal{W},\mathcal{U},\mathcal{Z},\alpha,\beta) \text{ satisfies (15), (16) and (17)} \\ \alpha \geq 0,\ \beta \geq 0 \end{cases} \end{aligned} \tag{PSD}$$

Problem (PSD) is thus a semi-definite program. The use of the sum $\alpha + \beta$ as objective function enforces the functions L^is to provide a minimal bound β and a minimal ellispoid containing the initial conditions. The constraint $\beta \geq 0$ is obvious since β represents a norm. However, $\alpha \geq 0$ is less natural but ensures that the objective function is bounded from below. The presence of the constraint $\alpha \geq 0$ does not affect the feasibility. Note that to reduce the size of the problem, we can take $q^i = 0$ and get an homogeneous wPQL function.

We remark the presence of $(1, 0_{1\times d})$ in the contruction of matrices E^i, E^{ij} and E^{i0} (see Eqs (10)). It would be more natural to express them without this vector. However, when we replace the cone-copositivity constraints by right-hand-side of Eq. ($\star$), we allow symmetry as it is shown in Example 2. The vector $(1, 0_{1\times d})$ aims to break it.

Example 2 (Why $(1, 0_{1\times d})$ in Eqs (10)?). *Let us consider* $X = \{x \in \mathbb{R} \mid x \leq 1\}$. *Let* $u(x) = (1, x)$, *and* $M = (1\ \ -1)$. *Then* $X = \{x \mid Mu(x)^\intercal \geq 0\}$.

Now let $W \geq 0$ *and define* $X' = \{x \mid u(x)M^\intercal W M u(x)^\intercal \geq 0\}$. *Since* $u(x)M^\intercal WMu(x)^\intercal = Wu(x)M^\intercal Mu(x)^\intercal = 2W(1-x)^2$, $X' = \mathbb{R}$ *for all* $W \geq 0$.

Now let us take $E = \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix}$ *and let* $W = \begin{pmatrix} w_1 & w_3 \\ w_3 & w_2 \end{pmatrix}$ *with* $w_1, w_2, w_3 \geq 0$ *and define* $\overline{X} = \{x \mid u(x)E^\intercal W E u(x)^\intercal \geq 0\}$. *Hence,* $u(x)E^\intercal \begin{pmatrix} w_1 & w_3 \\ w_3 & w_2 \end{pmatrix} Eu(x)^\intercal = w_1 + 2w_3(1-x) + w_2(1-x)^2$. *Taking for example* $w_2 = w_1 = 0$ *and* $w_3 > 0$ *implies that* $\overline{X} = X$.

Proposition 5. *Assume that Problem* (PSD) *has a feasible solution* $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$. *Then:*

1. The family $\mathcal{P}$ *defines a wPQL function* L*;*

2. There exists $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$ *satisfiying* (15), (16) *and* (17) *if and only if Problem* (PSD) *is feasible;*

3. Let $(i, j) \in \overline{\mathrm{Sw}}$. *Let us define the permutation* σ *as follows:* $\sigma(k) = n_i$ *if* $k = 1$, $k - 1$ *if* $2 \leq k \leq n_i$ *and* k *otherwise. and* P_σ *the associated permutation matrix*[1]. *Then, we have:*

$$\begin{array}{ll} & F^{i\intercal}\mathbf{M}(\mathrm{Id}, 0, 0)F^i \\ \preceq & \mathbf{M}(P^i, 2q^i, -\alpha) + \mathbf{M}(0, 0, \beta) \\ & -E^{ij\intercal}\left(P_\sigma^\intercal \begin{pmatrix} 0_{n_i-1} & 0_{n_i-1,n_j} \\ 0_{n_j,n_i-1} & W_p^j + W_+^j \end{pmatrix} P_\sigma + U_p^{ij} + U_+^{ij}\right) E^{ij} \ ; \end{array}$$

4. We have $\sup_{x\in X^0} \|x\|_2^2 \leq \beta$*;*

5. Assume that $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$ *is optimal with* $\alpha > 0$. *Then,* $\sup_{x\in X^0} L(x) = \alpha$.

Proof. *1.* This follows readily from Corollary 1.

2. The "if" part is obvious. Let us focus on the "only if" part and let $S^1 := (\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$ satisfiying (15), (16) and (17). From Th. 1, $\beta \geq 0$. Let us suppose that $\alpha < 0$ otherwise the proof is finished. Let us prove that $S^2 := (\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, 0, \beta - \alpha)$ is feasible for Problem (PSD). First $\beta - \alpha \geq 0$ since $\beta \geq 0$ and $\alpha < 0$. Second, $\mathbf{M}(P^i, 2q^i, 0) - \mathbf{M}(\mathrm{Id}, 0, -(\beta-\alpha)) - E^{i\intercal}\left(W_p^i + W_+^i\right)E^i = \mathbf{M}(P^i, 2q^i, -\alpha) - \mathbf{M}(\mathrm{Id}, 0, -\beta) - E^{i\intercal}\left(W_p^i + W_+^i\right)E^i$ thus S^2 satisfies (15) as S^1 does. Since α and β do not appear in (16), S^2 satisfies (16). Finally, $-\mathbf{M}(P^i, 2q^i, 0) - E^{i0\intercal}\left(Z_p^{0i} + Z_+^{0i}\right)E^{i0} = -\mathbf{M}(P^i, 2q^i, \alpha - \alpha) - E^{i0\intercal}\left(Z_p^{0i} + Z_+^{0i}\right)E^{i0} = \mathbf{M}(0, 0, -\alpha) - \mathbf{M}(P^i, 2q^i, -\alpha) - E^{i0\intercal}\left(Z_p^{0i} + Z_+^{0i}\right)E^{i0}$. We conclude that S^2 satisfies (17) as S^1 does.

3. Let $(i, j) \in \overline{\mathrm{Sw}}$. We have, $\mathbf{M}(P^j, 2q^j, -\alpha) - \mathbf{M}(\mathrm{Id}, 0, -\beta) - E^{j\intercal}\left(W_p^j + W_+^j\right)E^j \succeq 0$. Hence, $F^{i\intercal}\big(\mathbf{M}(P^j, 2q^j, -\alpha) - \mathbf{M}(\mathrm{Id}, 0, -\beta) - E^{j\intercal}\left(W_p^j + W_+^j\right)E^j\big)F^i \succeq 0$. Thus, $F^{i\intercal}\mathbf{M}(P^j, 2q^j, -\alpha)F^i - F^{i\intercal}E^{j\intercal}\left(W_p^j + W_+^j\right)E^jF^i \succeq F^{i\intercal}\mathbf{M}(\mathrm{Id}, 0, -\beta)F^i$. Hence, $-F^{i\intercal}E^{j\intercal}\left(W_p^j + W_+^j\right)E^jF^i + \mathbf{M}(P^i, 2q^i, 0) - E^{ij\intercal}\left(U_p^{ij} + U_+^{ij}\right)E^{ij} \succeq F^{i\intercal}\mathbf{M}(\mathrm{Id}, 0, -\beta)F^i$. By using $F^{i\intercal}\mathbf{M}(0, 0, -\beta)F^i = \mathbf{M}(0, 0, -\beta)$, we get that $F^{i\intercal}\mathbf{M}(\mathrm{Id}, 0, 0)F^i \preceq -F^{i\intercal}E^{j\intercal}\left(W_p^j + W_+^j\right)E^jF^i + \mathbf{M}(P^i, 2q^i, 0) - E^{ij\intercal}\left(U_p^{ij} + U_+^{ij}\right)E^{ij} + \mathbf{M}(0, 0, \beta)$. To end with the proof, it suffices to remark that $E^{ij} = P_\sigma^\intercal \begin{pmatrix} c^i & -T^i \\ \multicolumn{2}{c}{E^jF^i} \end{pmatrix}$.

4. Since $\mathcal{P}$ is wPQL, then from Th. 1 $\mathcal{R} \subseteq \{x \in \mathbb{R}^d \mid \|x\|_2^2 \leq \beta\}$ and since $X^0 \subseteq \mathcal{R}$, $\sup_{x\in X^0} \|x\|_2^2 \leq \beta$.

5. Assume that $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$ is an optimal solution with $\alpha > 0$ and suppose that $\sup_{x\in X^0} L(x) \neq \alpha$. From Constraint (17), we have for all $i \in \mathrm{In}$, $X^i \cap X^0 \subseteq \{x \mid L^i(x) \leq \alpha\}$ and thus for all $i \in \mathrm{In}$, $\sup_{x\in X^i\cap X^0} L^i(x) \leq \alpha$. Since $\sup_{x\in X^0} L(x) = \sup_{i\in\mathrm{In}} \sup_{x\in X^i\cap X^0} L^i(x)$, we get $\sup_{x\in X^0} L(x) \leq \alpha$. Now let $\epsilon > 0$ such that $\gamma = \alpha - \epsilon \geq 0$ and $\sup_{x\in X^0} L(x) \leq \gamma$. Let us define the matrix N by $N_{l,m} = 1$ if $l = m = 1$ and 0 otherwise. We have $-\mathbf{M}(P^i, 2q^i, -\gamma) - E^{i0\intercal}\left(Z_p^{0i} + Z_+^{0i}\right)E^{i0} = -\mathbf{M}(L^i) + \gamma N - E^{i0\intercal}\left(Z_p^{0i} + Z_+^{0i}\right)E^{i0}$. From $E_{1,1}^{i0} = 1$, we get $E^{i0\intercal}NE^{i0} = N$. So, $-\mathbf{M}(P^i, 2q^i, -\gamma) - E^{i0\intercal}\left(Z_p^{0i} + Z_+^{0i}\right)E^{i0} = -\mathbf{M}(L^i) + \alpha N - E^{i0\intercal}\left(Z_p^{0i} + \epsilon N + Z_+^{0i}\right)E^{i0}$. Now, $\mathbf{M}(P^i, 2q^i, -\gamma) - \mathbf{M}(\mathrm{Id}, 0, -\beta) - E^{i\intercal}\left(W_p^i + W_+^i\right)E^i = \mathbf{M}(P^i, 2q^i, -\alpha) - \mathbf{M}(\mathrm{Id}, 0, -\beta) - E^{i\intercal}\left(W_p^i + W_+^i\right)E^i + \epsilon N$. From Constraint (15), $\mathbf{M}(P^i, 2q^i, -\gamma) - \mathbf{M}(\mathrm{Id}, 0, -\beta) - E^{i\intercal}\left(W_p^i + W_+^i\right)E^i$ is positive semidefinite. We conclude that $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}', \gamma, \beta)$ with $\mathcal{Z}' = \{\left(Z_p^{i0} + \epsilon N, Z_+^{i0}\right) \in \mathbb{S}_{n_{i0}}^{\geq 0} \times \mathbb{S}_{n_{i0}}^{+}, i \in \mathrm{In}\}$ is feasible and $\gamma + \beta = \alpha + \beta - \epsilon$ thus $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$ cannot be optimal. $\square$

[1] $[P_\sigma]_{lk} = 1$ if $l = \sigma(k)$; 0 otherwise and $P_\sigma^{-1} = P_\sigma^\intercal$.

4. POLICY ITERATION ALGORITHM

In this section, we give details about the policy iteration algorithm which aims to make more accurate the overapproximation found directly from the computation of a wPQL function.

4.1 Sublevel Modelisation

In Def. 4, β is an upper bound on the Euclidian norm of the state variable. We do not have a precise upper bound on each coordinate considered separetely neither a precise upper bound on the state variable considering a specific cell. To obtain tigher bounds, we intersect S_α with other sublevel sets. In [25], the authors propose to combine quadratic Lyapunov functions with the square of coordinate. In this paper, we apply this technique replacing quadratic Lyapunov functions by wPQL functions. Thus we are interested in a set V of the form $S_\alpha \cap \cup_{i\in\mathcal{I}}\{y \in X^i \mid y_l^2 \leq \beta_l^i, l = 1, \ldots, d\}$. The computation of V is thus reduced to compute β_l^i. In verification of programs, the method is called a *templates domain abstraction* (for more background [3]).

We can deduce from Eq. (4) that $\mathcal{R} = \mathbb{A}(\mathcal{R}) \cup X^0$. We introduce the map $F : \wp(\mathbb{R}^d) \mapsto \wp(\mathbb{R}^d)$ defined by

$$C \mapsto F(C) := \mathbb{A}(C) \cup X^0 \ .$$

Hence, $\mathcal{R}$ is the smallest fixed point of F in the sense of if $C = F(C)$ then $\mathcal{R} \subseteq C$. From Tarski's theorem [29], since F is monotone on $\wp(\mathbb{R}^d)$, then:

$$\mathcal{R} = \inf\{C \in \wp(\mathbb{R}^d) \mid F(C) \subseteq C\}; \tag{18}$$

Any set C such that $F(C) \subseteq C$ satisfies $\mathcal{R} \subseteq C$. We propose to consider a restricted family of such sets C parameterized by $\omega \in \mathbb{R}^{d+1}$:

$$C(\omega) := \{x \in \mathbb{R}^d \mid \forall\, k \in [d],\ x_k^2 \leq \omega_k, L(x) \leq \omega_{d+1}\}$$

where L is a wPQL function for the PWA P. A set $C(\omega)$ is just the intersection of a sublevel of a wPQL function with a cartesian product of intervals. We define:

$$\forall\, k \in [d],\ X_k^0 = \sup_{y\in X^0} y_k^2 \text{ and } X_{d+1}^0 = \sup_{y\in X^0} L(y)$$

The numbers X_k^0 allows to construct interval bounds for the k-th coordinate of an initial value whereas X_{d+1}^0 refers to the maximum value of the wPQL function L over X^0.

We also define for all $(i,j) \in \overline{\mathrm{Sw}}$ and for all $\omega \in \mathbb{R}^{d+1}$:

$$\forall\, k \in [d], \qquad F_{ij,k}^{\sharp}(\omega) = \sup_{\substack{\forall\, k \in [d],\ x_k^2 \leq \omega_k,\\ L^i(x) \leq \omega_{d+1},\ x \in X^{ij}}} (A_{k\cdot}^i x + b_k^i)^2$$

and

$$F_{ij,d+1}^{\sharp}(\omega) = \sup_{\substack{\forall\, k \in [d],\ x_k^2 \leq \omega_k,\\ L^i(x) \leq \omega_{d+1},\ x \in X^{ij}}} L^j(A^i x + b^i)$$

and finally, we define for all $\omega \in \mathbb{R}^{d+1}$:

$$\forall\, l \in [d+1],\ F_l^{\sharp}(\omega) = \sup\{\sup_{(i,j) \in \overline{\mathrm{Sw}}} F_{ij,l}^{\sharp}(\omega), X_l^0\}$$

and $F^{\sharp}(\omega) = (F_1^{\sharp}(\omega), \ldots, F_{d+1}^{\sharp}(\omega))$. The map F acts on sets whereas $F^{\sharp}$ acts on vectors of $\mathbb{R}^{d+1}$. Prop. 6 highlights the link between the two maps.

PROPOSITION 6. *The following statements hold:*

1. $F(C(\omega)) \subseteq C(\omega) \iff F^{\sharp}(\omega) \leq \omega$*;*

2. $\mathcal{R} \subseteq \inf\{C(\omega) \mid \omega \in \mathbb{R}^{d+1}$ *s.t.* $F^{\sharp}(\omega) \leq \omega\}$*;*

3. For all $\omega \in \mathbb{R}^{d+1}$ *and all* $l \in [d+1]$*,* $F_{ij,l}^{\sharp}(\omega)$ *is the optimal value of quadratic program;*

4. For all $k \in [d]$*,* $X_k^0 = \max\{(\inf_{x \in X^0} x_k)^2, (\sup_{x \in X^0} x_k)^2\}$ *and if L is a computable wPQL function constructed from an optimal solution* $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$ *of* (PSD) *with* $\alpha > 0$*, then* $X_{d+1}^0 = \alpha$*.*

PROOF. *1.* We remark that $F(C(\omega)) \subseteq C(\omega)$ iff for all $k \in [d]$, $\sup_{y \in F(C(\omega))} y_k^2 \leq \omega_k$ and $\sup_{y \in F(C(\omega))} L(y) \leq \omega_{d+1}$. We only give details for the first inequality, the proof of the second follows the same idea. For all $k \in [d]$:

$$\begin{aligned}\sup_{y \in F(C(\omega))} y_k^2 &= \sup\{\sup_{y \in \mathbb{A}(C(\omega))} y_k^2, \sup_{y \in X^0} y_k^2\}\\ &= \sup\{\sup_{(i,j) \in \overline{\mathrm{Sw}}} \sup_{\substack{y = A^i x + b^i,\\ x \in C(\omega),\\ x \in X^{ij}}} y_k^2, \sup_{y \in X^0} y_k^2\} = F_k^{\sharp}(\omega)\end{aligned}$$

2. From Eq. (18), $\mathcal{R} \subseteq \inf\{C(\omega) \mid \omega \in \mathbb{R}^{d+1},\ F^{\sharp}(C(\omega)) \subseteq C(\omega)\}$. We conclude using the first point.

3. Direct from the definition of $F_{ij,l}^{\sharp}(\omega)$.

4. Let $k \in [d]$. Since X^0 is compact and $x \mapsto x_k$ is continuous, there exist $u, z \in X^0$ such that $u_k = \inf_{x \in X^0} x_k$ and $z_k = \sup_{x \in X^0} x_k$. Hence for all $x \in X^0$, $x_k^2 \leq \max(z_k^2, u_k^2)$. Since z and u belong to X^0, then $X_k^0 = \max(z_k^2, u_k^2)$. Since $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$ is an optimal solution of Problem (PSD) and $\alpha > 0$ then $X_{d+1}^0 = \alpha$ from Prop. 5. □

Now, we assume that Problem (PSD) has an optimal solution $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$ with $\alpha > 0$ and we denote by L the associated wPQL function.

From Prop. 6, to compute $F_{ij,l}^{\sharp}(\omega)$ is reduced to solve a quadratic maximization known to be NP-Hard [30]. So we propose to compute instead a safe overapproximation using Lagrange duality and semi-definite programming.

4.2 Relaxed functional

In this subsection, we define the function on which we compute a fixed point.

For this subsection, we fix $(i,j) \in \overline{\mathrm{Sw}}$ and $\omega \in \mathbb{R}^{d+1}$. For all $k \in [d]$, we write $\mathbf{M}_k$ for $\mathbf{M}(x \mapsto x_k^2)$ and for all $i \in \mathcal{I}$, $\mathbf{M}_L^i$ for $\mathbf{M}(L^i)$. The matrix $\mathbf{N} \in \mathbb{M}_{(d+1)\times(d+1)}$ is defined by $\mathbf{N}_{l,m} = 1$ if $l = m = 1$ and 0 otherwise.

Let $\lambda \in \mathbb{R}_+^{d+1}$, $Y \in \mathbb{S}_{n_{ij}}^{\geq 0}$ and $Z \in \mathbb{S}_{n_{ij}}^{+}$. We construct the auxiliary matrices:

$$\begin{aligned}&\Phi_{ij,k}(\lambda, Y, Z) =\\ &{F^i}^{\intercal}\mathbf{M}_k F^i - \sum_{l=1}^{d} \lambda_l \mathbf{M}_l - \lambda_{l+1}\mathbf{M}_L^i + {E^{ij}}^{\intercal}(Y+Z)E^{ij}\\ &\Phi_{ij,d+1}(\lambda, Y, Z) =\\ &{F^i}^{\intercal}\mathbf{M}_L^j F^i - \sum_{l=1}^{d} \lambda_l \mathbf{M}_l - \lambda_{l+1}\mathbf{M}_L^i + {E^{ij}}^{\intercal}(Y+Z)E^{ij}\end{aligned} \tag{19}$$

For all $l \in [d+1]$, for all $\omega \in \mathbb{R}_+^{d+1}$:

$$\begin{aligned}&F_{ij,l}^{\mathcal{R}}(\omega) =\\ &\inf_{\lambda,\eta,Y,Z} \quad \eta\\ &\text{s.t.} \quad \begin{cases}(\eta - \sum_{k=1}^{d+1}\lambda_k\omega_k)\mathbf{N} - \Phi_{ij,l}(\lambda, Y, Z) \succeq 0,\\ \lambda \in \mathbb{R}_+^{d+1},\ \eta \in \mathbb{R},\ Y \geq 0,\ Z \succeq 0\end{cases}\end{aligned} \tag{20}$$

$$F_l^{\mathcal{R}}(\omega) = \sup\{\sup_{(i,j) \in \overline{\mathrm{Sw}}} F_{ij,l}^{\mathcal{R}}(\omega), X_l^0\}$$

and $F^{\mathcal{R}}(\omega) = (F_1^{\mathcal{R}}(\omega), \ldots, F_{d+1}^{\mathcal{R}}(\omega))$. The map $F^{\mathcal{R}}$ is computable overapproximation of $F^{\sharp}$. Indeed, $F^{\mathcal{R}}(\omega)$ is the optimal value of a semidefinite program which can be solved in polynomial time[2] and thus relies on SDP solvers.

PROPOSITION 7 (SAFE OVERAPPROXIMATION). *The following assertions are true:*

1. For all $l \in [d+1]$*,* $F_l^{\mathcal{R}}$ *is the optimal value of a SDP program;*

2. $F^{\sharp} \leq F^{\mathcal{R}}$.

PROOF. *1.* The first statement is straightforward.

2. We have to prove that for all $k \in [d+1]$, for all $\omega \in \mathbb{R}^{d+1}$, $F_{ij,k}^{\sharp}(\omega) \leq F_{ij,k}^{\mathcal{R}}(\omega)$. We do the proof for the case $k = d+1$. The other cases follows the same proof constructions.

Applying the Weak Duality Theorem (see e.g. [5, Sect. 5.3]), we obtain:

$$\begin{aligned}F_{ij,d+1}^{\sharp}(\omega) \leq \inf_{\lambda \in \mathbb{R}_+^{d+1}} \sup_{x \in X^{ij}} L^j(f^i(x)) + \sum_{k=1}^{d} \lambda_k(\omega_k - x_k^2)\\ +\lambda_{d+1}(\omega_{d+1} - L^i(x))\end{aligned}$$

Let us write $q_{\eta,\lambda} := \eta - L^j(f^i(x)) - \sum_{k=1}^{d} \lambda_k(\omega_k - x_k^2) - \lambda_{d+1}(\omega_{d+1} - L^i(x))$. Then $\sup_{x \in X^{ij}} L^j(f^i(x)) + \sum_{k=1}^{d} \lambda_k(\omega_k - x_k^2) + \lambda_{d+1}(\omega_{d+1} - L^i(x)) = \inf\{\eta \mid q_{\eta,\lambda}(x) \geq 0,\ \forall\, x \in X^{ij}\}$.

From Lemma 2, $\mathbf{M}(q_{\eta,\lambda}) \in \mathbf{C}_{d+1}(E^{ij}) \implies (q_{\eta,\lambda}(x) \geq 0,\ \forall\, x \in X^{ij})$. From Corollary 1, $(\mathbf{M}(q_{\eta,\lambda}) - {E^{ij}}^{\intercal}(Y+Z)E^{ij} \succeq 0$ for some $Y \geq 0$ and $Z \succeq 0) \implies \mathbf{M}(q_{\eta,\lambda}) \in \mathbf{C}_{d+1}(E^{ij})$. Now from Eq. (9) and since $A \to \mathbf{M}(A)$ is linear, we have $\mathbf{M}(q_{\eta,\lambda}) = (\eta - \sum_{k=1}^{d+1}\lambda_k\omega_k)N - \Phi_{ij,d+1}(\lambda, Y, Z) + {E^{ij}}^{\intercal}(Y+Z)E^{ij}$. Since $F_{ij,l}^{\mathcal{R}}$ is the infimum of η over the constraint $(\eta - \sum_{k=1}^{d+1}\lambda_k\omega_k)N - \Phi_{ij,d+1}(\lambda, Y, Z) \succeq 0$, $\lambda \in \mathbb{R}_+^{d+1}$, $\eta \in \mathbb{R}$, $Y \geq 0$ and $Z \succeq 0$, this achieves the proof. □

[2]The term ”polynomial time” here must be taken very carefully. Some precisions over the complexity analysis of SDP problems can be found in [24].

LEMMA 4. *Let $(i,j) \in \overline{\mathrm{Sw}}$, $l \in [d+1]$ and $\omega \in \mathbb{R}^{d+1}$. Then:*

$$F^{\mathcal{R}}_{ij,l}(\omega) = \inf_{\lambda \in \mathbb{R}^{d+1}_{+}} F^{\lambda}_{ij,l}(\omega)$$

where

$$F^{\lambda}_{ij,l}(\omega) = \sum_{m=1}^{d+1} \lambda_m \omega_m + \inf_{\substack{Y \succeq 0 \\ Z \succeq 0}} \sup_{x \in \mathbb{R}^d} \begin{pmatrix} 1 \\ x \end{pmatrix}^{\intercal} \Phi_{ij,l}(\lambda, Y, Z) \begin{pmatrix} 1 \\ x \end{pmatrix} \tag{21}$$

PROPOSITION 8. *Let $(i,j) \in \overline{\mathrm{Sw}}$, $l \in [d+1]$, $\lambda \in \mathbb{R}^{d+1}_{+}$. The following statements are true:*

1. *$F^{\lambda}_{ij,l}$ is affine;*
2. *$F^{\lambda}_{ij,l}$, $F^{\mathcal{R}}_{ij,l}$ and $F^{\mathcal{R}}_{l}$ are monotone;*
3. *$F^{\mathcal{R}}_{ij,l}$ and $F^{\mathcal{R}}_{l}$ are upper semi-continuous.*

PROOF. The first assertion follows readily from Eq. (21). The function $w \mapsto F^{\lambda}_{ij,l}(w)$ is monotone from the positivity of λ and the two last functions are monotone as the supremum of monotone functions. The function $w \mapsto F^{\mathcal{R}}_{ij,l}(w)$ is upper semi-continuous as the infimum of continuous functions and $w \mapsto F^{\mathcal{R}}_{l}(w)$ is upper semi-continuous as the finite supremum of upper semi-continuous functions. □

To be able to perform a new step in policy iteration, we need a selection property. In our case, the selection property relies on the existence of an optimal dual solution.

DEFINITION 5 (SELECTION PROPERTY). *Let $(i,j) \in \overline{\mathrm{Sw}}$ and $l \in [d+1]$. We say that $\omega \in \mathbb{R}^{d+1}$ satisfies the selection property if there exists $\lambda \in \mathbb{R}^{d+1}_{+}$ such that:*

$$F^{\mathcal{R}}_{ij,l}(\omega) = F^{\lambda}_{ij,l}(\omega) \tag{22}$$

We define:

$$\mathrm{Sol}_{\lambda}\left((i,j), l, \omega\right) := \{\lambda \in \mathbb{R}^{d+1}_{+} \mid F^{\mathcal{R}}_{ij,l}(\omega) = F^{\lambda}_{ij,l}(\omega)\}$$

and

$$\mathcal{S} := \{\omega \in \mathbb{R}^{d+1} \mid \forall (i,j) \in \overline{\mathrm{Sw}}, \forall l \in [d+1], \mathrm{Sol}_{\lambda}\left((i,j), l, \omega\right) \neq \emptyset\} \enspace .$$

COROLLARY 2. *Let $(i,j) \in \overline{\mathrm{Sw}}$, $l \in [d+1]$ and $\omega \in \mathcal{S}$. Now let $\overline{\lambda} \in \mathrm{Sol}_{\lambda}\left((i,j), \omega, p\right)$, then:*

$$\inf_{\substack{Y \succeq 0 \\ Z \succeq 0}} \sup_{x \in \mathbb{R}^d} \begin{pmatrix} 1 \\ x \end{pmatrix}^{\intercal} \Phi_{ij,l}(\lambda, Y, Z) \begin{pmatrix} 1 \\ x \end{pmatrix} = F^{\mathcal{R}}_{ij,l}(\omega) - \sum_{m=1}^{d+1} \overline{\lambda}_m \omega_m \enspace .$$

Let $(i,j) \in \overline{\mathrm{Sw}}$, $l \in [d+1]$ and $\omega \in \mathcal{S}$. From Corollary 2, for all $\lambda \in \mathrm{Sol}_{\lambda}\left((i,j), l, \omega\right)$, for all $v \in \mathbb{R}^{d+1}$, we have:

$$F^{\lambda}_{ij,l}(v) = \sum_{m=1}^{d+1} \lambda_m v_m + F^{\mathcal{R}}_{ij,l}(\omega) - \sum_{m=1}^{d+1} \overline{\lambda}_m \omega_m \tag{23}$$

We remark that $F^{\lambda}_{ij,l}(\omega) = F^{\mathcal{R}}_{ij,l}(\omega)$.

From the first statement of Prop. 6 and the second assertion of Prop. 7, the most precise overapproximation of $\mathcal{R}$ (with this templates basis) is given by $\overline{\omega} = \inf\{\omega \in \mathbb{R}^{d+1} \mid F^{\mathcal{R}}(\omega) \leq \omega\}$. From Tarski's theorem, $\overline{\omega}$ is the smallest fixed point of $F^{\mathcal{R}}$. However, the smallest is difficult to get and since any vector ω such that $F^{\mathcal{R}}(\omega) \leq \omega$ furnishes a valid but less precise overapproximation of $\mathcal{R}$, we perform policy iterations until a fixed point is reached.

4.3 Policies definition

A policy iteration algorithm can be used to solve a fixed point equation for a monotone function written as an infimum of a family of simpler monotone functions, obtained by selecting *policies*, see [9, 13] for more background. The idea is to solve a sequence of fixed point problems involving the simple functions. In the present setting, we look for a representation of the relaxed function:

$$\forall (i,j) \in \overline{\mathrm{Sw}},\ \forall l \in [d+1],\ F^{\mathcal{R}}_{ij,l} = \inf_{\pi \in \Pi} F^{\pi}_{ij,l} \tag{24}$$

where the infimum is taken over a set Π whose elements π are called *policies*, and where each function F^{π} is required to be monotone. The correctness of the algorithm relies on a selection property, meaning in the present setting that for each argument $((i,j), l, \omega)$ there must exist a policy π such that $F^{\mathcal{R}}_{ij,l}(\omega) = F^{\pi}_{ij,l}(\omega)$. The idea of the algorithm is to start from a policy π^0, compute the smallest fixed point ω of F^{π^0}, evaluate $F^{\mathcal{R}}$ at point ω, and, if $\omega \neq F^{\mathcal{R}}(\omega)$, determine the new policy using the selection property at point ω.

Let us now identify the policies. Lemma 4 shows that for all $l \in [d+1]$, $F^{\mathcal{R}}_{ij}$ can be written as the infimum of the family of affine functions F^{λ}_{ij}, the infimum being taken over the set of $\lambda \in \mathbb{R}^{d+1}_{+}$. When $\omega \in \mathcal{S}$ is given, choosing a policy π consists in selecting, for each $(i,j) \in \overline{\mathrm{Sw}}$ and for all $l \in [d+1]$, a vector $\lambda \in \mathrm{Sol}_{\lambda}\left((i,j), l, \omega\right)$. We denote by $\pi_{ij,l}(\omega)$ the value of λ chosen by the policy π. Then, the map $F^{\pi_{ij,l}}_{ij,l}$ in Eq. (24) is obtained by replacing $F^{\mathcal{R}}_{ij,l}$ by $F^{\lambda}_{ij,l}$ appearing in Eq. (23). Finally, we define, for all $l \in [d+1]$:

$$F^{\pi}_{l}(\omega) = \sup\{\sup_{(i,j) \in \overline{\mathrm{Sw}}} F^{\pi_{ij,l}}_{ij,l}(\omega), X^0_l\}$$

and $F^{\pi} = (F^{\pi}_1, \ldots, F^{\pi}_{d+1})$.

Now, we can define concretely the policy iteration algorithm at Algorithm 1.

Algorithm 1 Policy Iteration with wPQL functions

1 Choose $\pi^0 \in \Pi$, $k = 0$.

2 Define F^{π^k} by choosing λ according to policy π^k using Eq. (23).

3 Compute the smallest fixed point ω^k in $\mathbb{R}^{d+1}$ of F^{π^k}.

4 If $\omega^k \in \mathcal{S}$ continue otherwise return ω^k.

5 Evaluate $F^{\mathcal{R}}(\omega^k)$, if $F^{\mathcal{R}}(\omega^k) = \omega^k$ return ω^k otherwise take π^{k+1} s.t. $F^{\mathcal{R}}(\omega^k) = F^{\pi^{k+1}}(\omega^k)$. Increment k and go to 2.

4.4 Some details about Policy Iteration algorithm

Initialization. Policy iteration algorithm needs an initial policy. Recall that L was computed from an optimal solution $(\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$ of Problem (PSD) with $\alpha > 0$. The first policy is given by an element in $\mathrm{Sol}_{\lambda}\left((i,j), l, w^0\right)$ where w^0 is defined by:

$$\forall k \in [d],\ \omega^0_k = \beta,\ w^0_{d+1} = \alpha \tag{25}$$

PROPOSITION 9. *The vector ω^0 satisfies $F^{\mathcal{R}}(\omega^0) \leq \omega^0$.*

PROOF. From Prop. 5, we have for all $k \in [d]$, $X_k^0 \leq \beta = \omega_k^0$ and $X_{d+1}^0 = \alpha = \omega_{d+1}^0$. Let $(i,j) \in \overline{Sw}$ and $l \in [d+1]$. We have to prove that $F_{ij,l}^{\mathcal{R}}(\omega^0) \leq \omega^0$. It suffices to prove there exist $\lambda \geq 0$, $Y \geq 0$ and $Z \succeq 0$ such that:

$$(\omega_l^0 - \sum_{k=1}^{d+1} \lambda_k \omega_k) N - \Phi_{ij,l}(\lambda, Y, Z) \succeq 0 \tag{26}$$

Indeed, if Eq. (26) holds then $(\omega_l^0(p), \lambda, Y, Z)$ is feasible for the SDP problem (20) and thus $F_{ij,l}^{\mathcal{R}}(\omega^0) \leq \omega_l^0$.

Let us define $\bar{\lambda}$ by $\bar{\lambda}_{d+1} = 1$ and $\bar{\lambda}_k = 0$ for all $k \in [d]$. Let us simply write $S := (\mathcal{P}, \mathcal{W}, \mathcal{U}, \mathcal{Z}, \alpha, \beta)$.

Let $l = d+1$ and extract U_p^{ij} and U_+^{ij} from $\mathcal{U}$. Then, $(\omega_{d+1}^0 - \sum_{k=1}^{d+1} \bar{\lambda}_k \omega_k) N - \Phi_{ij,l}(\bar{\lambda}, U_p^{ij}, U_+^{ij}) = -F^{i\intercal} \mathbf{M}_L^j F^i + \mathbf{M}_L^i - E^{ij\intercal}(U_p^{ij} + U_+^{ij}) E^{ij}$. Since S satisfies Eq. (16) as an optimal solution of Problem (PSD), Eq (26) holds with $\lambda = \bar{\lambda}$, $Y = U_p^{ij}$ and $Z = U_+^{ij}$.

Let $l \in [d]$, $\bar{Y} = P_\sigma^\intercal \begin{pmatrix} 0_{n_i-1} & 0_{n_i-1,n_j} \\ 0_{n_j,n_i-1} & W_p^j \end{pmatrix} P_\sigma + U_p^{ij}$ and $\bar{Z} = P_\sigma^\intercal \begin{pmatrix} 0_{n_i-1} & 0_{n_i-1,n_j} \\ 0_{n_j,n_i-1} & W_+^j \end{pmatrix} P_\sigma + U_+^{ij}$ where P_σ is the permutation matrix defined at Prop 5, W_p^j and W_+^j are extracted from $\mathcal{W}$ and U_p^{ij} and U_+^{ij} are extracted from $\mathcal{U}$. We have $(\omega_l^0 - \sum_{k=1}^{d+1} \bar{\lambda}_k \omega_k) N - \Phi_{ij,l}(\bar{\lambda}, \bar{Y}, \bar{Z}) = \mathbf{M}(0,0,\beta-\alpha) - F^{i\intercal} \mathbf{M}_l F^i + \mathbf{M}_L^i - E^{ij\intercal}(\bar{Y} + \bar{Z}) E^{ij}$. Now, remark that $\mathbf{M}_l \preceq \mathbf{M}(\mathrm{Id}, 0, 0)$ and thus $-F^{i\intercal} \mathbf{M}_l F^i + \mathbf{M}(P^i, 2q^i, -\alpha) - E^{ij\intercal}(\bar{Y} + \bar{Z}) E^{ij} + \mathbf{M}(0,0,\beta) \preceq -F^{i\intercal} \mathbf{M}(\mathrm{Id}, 0, 0) F^i + \mathbf{M}(P^i, 2q^i, -\alpha) - E^{ij\intercal}(\bar{Y} + \bar{Z}) E^{ij} + \mathbf{M}(0,0,\beta)$. We conclude that Eq (26) holds with $\lambda = \bar{\lambda}$, $Y = \bar{Y}$ and $Z = \bar{Z}$ from the second assertion of Prop. 5. □

Smallest fixed point computation associated to a policy. For the third step of Algorithm 1, using Lemma 4, F^π is monotone and affine, we compute the smallest fixed point of F^π by solving the following LP see [13, Section 4]:

$$\min \left\{ \sum_{k=1}^{d+1} w_k \text{ s.t. } F^\pi(w) \leq w \right\} \tag{27}$$

Convergence. In [1], it is proved that the policy iterations algorithm in the quadratic setting converges towards a fixed point of $F^{\mathcal{R}}$. Here, we establish a similar result (Th. 1). Combined with Prop. 7, this fixed point provides a safe over-approximation of $\mathcal{R}$.

Let $(w^l)_{l \in \mathbb{N}}$ be the sequence generated by Algorithm 1. If $w^l \notin \mathcal{S}$ and $w^{l-1} \in \mathcal{S}$, then we set $w^k = w^l$ for all $k \geq l$.

THEOREM 1. *The following assertions hold:*
1. For all $l \in \mathbb{N}$, $F^{\mathcal{R}}(w^l) \leq w^l$;
2. The sequence $(w^l)_{l \geq 0}$ is decreasing. Moreover for all $l \in \mathbb{N}$ such that $w^{l-1} \in \mathcal{S}$ either $w^l = w^{l-1}$ and $F^{\mathcal{R}}(w^l) = w^l$ or $w^l < w^{l-1}$;
3. For all $l \in \mathbb{N}$, for all $k \in [d+1]$, $X_k^0 \leq w_k^l \leq w_k^0$;
4. The limit w^∞ of $(w^l)_{l \geq 0}$ satisfies: $F^{\mathcal{R}}(w^\infty) \leq w^\infty$. Moreover if $\forall\, k \in \mathbb{N}$, $w^k \in \mathcal{S}$ then $F^{\mathcal{R}}(w^\infty) = w^\infty$.

PROOF. *1.* From Prop. 9, $F^{\mathcal{R}}(w^0) \leq w^0$. Now, let $l > 0$ and assume $w^{l-1} \in \mathcal{S}$, there exists π^l such that, $F^{\pi^l}(w^l) = w^l$ and since $F^{\mathcal{R}} = \inf_\pi F^\pi$, we get $F^{\mathcal{R}}(w^l) \leq F^{\pi^l}(w^l) = w^l$. If $w^{l-1} \notin \mathcal{S}$, then there exists $k \in \mathbb{N}$, $k \leq l-1$ such that $w^{k-1} \in \mathcal{S}$ and $w^l = w^k$, and thus $F^{\mathcal{R}}(w^k) \leq w^k$.

2. Let $l \in \mathbb{N}$, if $w^{l-1} \notin \mathcal{S}$, $w^l = w^{l-1}$. Now suppose $w^{l-1} \in \mathcal{S}$. There exists $\pi^l \in \Pi$ such that $F^{\mathcal{R}}(w^{l-1}) = F^{\pi^l}(w^{l-1}) \leq w^{l-1}$ and since $w^l = \inf\{v \in \mathbb{R}^{d+1} \mid F^{\pi^l}(v) \leq v\}$ then $w^l \leq w^{l-1}$. Now if $w^l = w^{l-1}$, $F^{\mathcal{R}}(w^{l-1}) = F^{\mathcal{R}}(w^l) = F^{\pi^l}(w^{l-1}) = F^{\pi^l}(w^l) = w^l = w^{l-1}$.

3. From Prop. 6, Prop. 7 and the first assertion, $X_k^0 \leq F_k^\sharp(w^l) \leq F_k^{\mathcal{R}}(w^l) \leq w_k^l$.

4. First, w^∞ exists since $(w_l)_{l \in \mathbb{N}}$ is decreasing and bounded from below (third assertion). Then, for all $l \in \mathbb{N}$, $w^\infty \leq w^l$ and since $F^{\mathcal{R}}$ is monotone (Prop. 8) $F^{\mathcal{R}}(w^\infty) \leq F^{\mathcal{R}}(w^l) \leq w^l$. Taking the infimum over l, we get $F^{\mathcal{R}}(w^\infty) \leq w^\infty$. Now we prove that $w^\infty \leq F^{\mathcal{R}}(w^\infty)$. Let $l \in \mathbb{N}$. Since $w^l \in \mathcal{S}$, there exists $\pi^{l+1} \in \Pi$ such that $F^{\pi^{l+1}}(w^l) = F^{\mathcal{R}}(w^l)$. Moreover, $w^{l+1} \leq w^l$ and since $F^{\pi^{l+1}}$ is monotone: $w^{l+1} = F^{\pi^{l+1}}(w^{l+1}) \leq F^{\pi^{l+1}}(w^l) = F^{\mathcal{R}}(w^l)$. Now by taking the infimum on l, we get $w^\infty = \inf_l w^{l+1} = \inf_l w^l \leq \inf_l F^{\mathcal{R}}(w^l)$. Finally, since $F^{\mathcal{R}}$ is upper semicontinuous (Prop. 8), then $\inf_k F^{\mathcal{R}}(w^k) = \limsup_k F^{\mathcal{R}}(w^k) \leq F^{\mathcal{R}}(\lim_k w^k) = F^{\mathcal{R}}(w^\infty)$. Hence, $w^\infty \leq F^{\mathcal{R}}(w^\infty)$. □

5. EXAMPLES

The following examples are performed using YALMIP interfaced with the SDP solver MOSEK.

5.1 Example from [21] slighty modified

Consider the followinf PWA: $X^0 = [-1,1] \times [-1,1]$, and, for all $k \in \mathbb{N}$:

$$x_{k+1} = \begin{cases} A^1 x_k & \text{if } x_{k,1} \geq 0 \text{ and } x_{k,2} \geq 0 \\ A^2 x_k & \text{if } x_{k,1} \geq 0 \text{ and } x_{k,2} < 0 \\ A^3 x_k & \text{if } x_{k,1} < 0 \text{ and } x_{k,2} < 0 \\ A^4 x_k & \text{if } x_{k,1} < 0 \text{ and } x_{k,2} \geq 0 \end{cases}$$

with

$$A^1 = \begin{pmatrix} -0.04 & -0.461 \\ -0.139 & 0.341 \end{pmatrix},\ A^2 = \begin{pmatrix} 0.936 & 0.323 \\ 0.788 & -0.049 \end{pmatrix}$$
$$A^3 = \begin{pmatrix} -0.857 & 0.815 \\ 0.491 & 0.62 \end{pmatrix},\ A^4 = \begin{pmatrix} -0.022 & 0.644 \\ 0.758 & 0.271 \end{pmatrix}$$

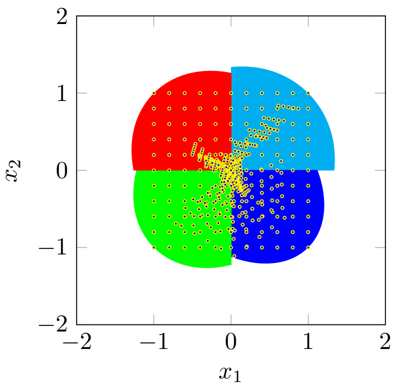
Then, we have $X^1 = \mathbb{R}_+ \times \mathbb{R}_+$, $X^2 = \mathbb{R}_+ \times \mathbb{R}_-^*$, $X^3 = \mathbb{R}_-^* \times \mathbb{R}_-^*$ and $X^4 = \mathbb{R}_-^* \times \mathbb{R}_+$.

From Prop. 2, $\mathrm{In} = \{1,2,3,4\}$ and $\overline{\mathrm{Sw}} = \{(i,j) \mid S(i,j) = 1\}$ with $S = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$.

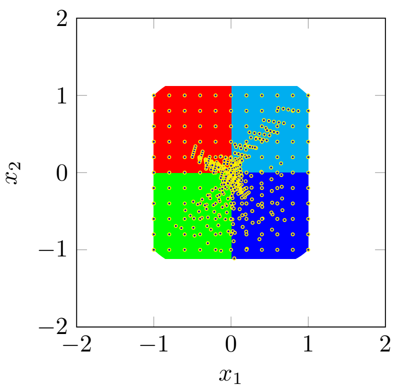
By solving Problem PSD, we get a (optimal) wPQL function L characterized by the following matrices:

$$P^1 = \begin{pmatrix} 1.1178 & -0.1178 \\ -0.1178 & 1.1178 \end{pmatrix},\ P^2 = \begin{pmatrix} 1.5907 & 0.5907 \\ 0.5907 & 1.5907 \end{pmatrix},$$
$$P^3 = \begin{pmatrix} 1.3309 & -0.3309 \\ -0.3309 & 1.3309 \end{pmatrix},\ P^4 = \begin{pmatrix} 1.2558 & 0.2558 \\ 0.2558 & 1.2558 \end{pmatrix}$$

Since $\alpha = \beta = 2$, then $\mathcal{R} \subseteq \{x \in \mathbb{R}^2 \mid L(x) \leq 2\} \subseteq \{x \in \mathbb{R}^2 \mid \|x\|_2^2 \leq 2\}$. The sets $\mathcal{R}$ (discretized version) and $\{x \in \mathbb{R}^2 \mid L(x) \leq 2\}$ are depicted at Figure 1a. Then we enter into policy iteration algorithm. From Eq. (25), we define w^0 by $w_1^0 = 2.0000$, $w_2^0 = 2.0000$, $w_3^0 = 2.0000$. Then we compute the image of w^0 by the relaxed semantics $F^{\mathcal{R}}(w^0)$ using semidefinite programming (see Eq. (20)). We check that w^0 is not a fixed point of $F^{\mathcal{R}}$ and then the initial policy



(a) First overapproximation found by (PSD)



(b) Final overapproximation found by policy iterations

Figure 1: (Discretized) $\mathcal{R}$ in yellow and initial (Fig. 1a) and last overapproximations (Fig. 1b) of $\mathcal{R}$.

$\pi^0((i,j),l,w^0)$ is the vector λ extracted from the optimal solutions (λ, Y, Z) of the semidefinite programs involved in the computation of $F^{\mathcal{R}}(w^0)$. For example, for $(1,3) \in \overline{\mathrm{Sw}}$ and $l = 1$, $\pi^0((1,3),1,w^0) = (0.0000, 0.0000, 0.0430)^\intercal$, where the first two zeros are the Lagrange multipliers associated to $\mathbf{M}_1$ and $\mathbf{M}_2$ and 0.0430 is the Lagrange multiplier associated to $\mathbf{M}(L^1)$. We compute the smallest fixed point associated to π^0 using the LP (27):

$$w_1^1 = 1.1036,\ w_2^1 = 1.2443,\ w_3^1 = 2.0000$$

Moreover, at each step k, policy iterations provides auxiliary values which represent the overapproximations of the polyhedra $\mathcal{R} \cap X^i \cap {A^i}^{-1}(X^j)$ by ellipsoids of the form $\{x \in \mathbb{R}^2 \mid x_1^2 \leq w_{ij,1}^k,\ x_2^2 \leq w_{ij,2}^k,\ L(x_1,x_2) \leq w_{ij,3}^k\}$. For example, for $k=0$:

$$\begin{aligned}
w_{11,1} &= 0.0000,\ w_{11,2} = 0.0000,\ w_{11,3} = 0.0000\\
w_{13,1} &= 0.0573,\ w_{13,2} = 0.0213,\ w_{13,3} = 0.0213\\
w_{14,1} &= 0.3012,\ w_{14,2} = 0.1447,\ w_{14,3} = 0.1447
\end{aligned}$$

Note that we found that for $(i,j) = (1,1)$, $w_{ij,1}^1 = w_{ij,2}^1 = w_{ij,3}^1 = 0$ which means that $\mathcal{R} \cap X^1 \cap {A^1}^{-1}(X^1)$ is reduced to the singleton $(0,0)$. The invariant found is depicted at Figure 1b. Finally, we find after two iterations that for all $k \in \mathbb{N}$, $x_{1,k}^2 \leq 1$, $x_{2,k}^2 \leq 1.2443$ and $L(x_{1,k}, x_{2,k}) \leq 2$.

5.2 A (piecewise) affine example

We now consider the following PWA: $X^0 = [0,3] \times [0,2]$ and for all $k \in \mathbb{N}$:

$$x_{k+1} = \begin{cases} A^1 x_k + b^1 & \text{if } T(x_k) < c \\ A^2 x_k + b^2 & \text{if } T(x_k) \geq c \end{cases}$$

with

$$A^1 = \begin{pmatrix} 0.4197 & -0.2859 \\ 0.5029 & 0.1679 \end{pmatrix},\quad b^1 = \begin{pmatrix} 2.0000 \\ 5.0000 \end{pmatrix},$$
$$A^2 = \begin{pmatrix} -0.0575 & -0.4275 \\ -0.3334 & -0.2682 \end{pmatrix},\quad b^2 = \begin{pmatrix} -4.0000 \\ 4.0000 \end{pmatrix}$$

$$T = \begin{pmatrix} 3.0000 & 8.0000 \end{pmatrix} \text{ and } c = -3.0000$$

By Prop. 2, $\overline{\mathrm{Sw}} = \mathcal{I}^2 = \{(1,1),(1,2),(2,1),(2,2)\}$ and $\mathrm{In} = \{2\}$. Using Problem (PSD), we compute the wPQL function L characterized by:

$$P^1 = \begin{pmatrix} 2.9888 & -1.7890 \\ -1.7890 & 8.0295 \end{pmatrix},\quad q^1 = \begin{pmatrix} -14.7283 \\ -94.1347 \end{pmatrix}$$
and
$$P^2 = \begin{pmatrix} 2.7192 & 2.0930 \\ 2.0930 & 6.1110 \end{pmatrix},\quad q^2 = \begin{pmatrix} 5.5737 \\ -16.4198 \end{pmatrix}$$

and the invariant found is $\{x \in \mathbb{R}^2 \mid L(x) \leq 58.1165\}$ and an upper bound over the square Euclidian norm of the state variable is 286.4932. We run the policy iteration to get finally after 4 iterations the following bound vector:

$$w_1 = 41.8956,\ w_2 = 31.4449,\ w_3 = 58.1165$$

corresponding to the invariant set $\{x \in \mathbb{R}^2 \mid x_i^2 \leq w_i,\ L(x) \leq w_3\}$.

We obtain interesting information during policy iterations. At step $k = 0$, when we select the initial policy, the SDP solver returns for all $l = 1,2,3$, $F_{11,l}^{\mathcal{R}}(w^0) = -\infty$ and from Prop. 7 this implies that $\sup_{x \in \mathcal{R} \cap X^1 \cap {f^1}^{-1}(X^1)} p(A^1 x + b^1)$ is not feasible hence $(1,1) \notin \mathrm{Sw}$. At the iteration step $k=1$, the SDP solver provides for all $l = 1,2,3$, $F_{21,l}^{\mathcal{R}}(w^1) = -\infty$ and from Prop. 7 this implies that $(2,1) \notin \mathrm{Sw}$. Finally, $\mathrm{Sw} \subseteq \{(1,2),(2,2)\}$. Recalling that $1 \notin \mathrm{In}$, we conclude that the system state variable only stays in X^2 and thus the system is actually equivalent to a *constrained affine system*. This information is computed *automatically*.

6. CONCLUSION AND FUTURE WORKS

We have developed a method to compute *automatically* by semi-definite programming precise bounds over the reachable values set of a piecewise affine system. The method combines weak piecewise quadratic Lyapunov functions to generate a first overapproximation and policy iterations used to reduce the initial overapproximation.

Future works could be to design a repartitioning method in order to improve the feasibility of Problem (PSD). In the same direction, we could use SOS programming to rewrite copositive constraints. Also, future works should contain the study of the optimality of the presented policy iterations algorithm (providing the most precise overapproximation considering bounding the square of coordinates variables or not).

7. REFERENCES

[1] A. Adjé. Policy iteration in finite templates domain. In *Numerical Software Verification (NSV 2014)*, 2014.

[2] A. Adjé and P.-L. Garoche. Automatic Synthesis of Piecewise Linear Quadratic Invariants for Programs. In *Verification, Model Checking, and Abstract Interpretation - 16th International Conference, VMCAI 2015, Mumbai, India, January 12-14, 2015. Proceedings*, pages 99–116, 2015.

[3] A. Adjé, S. Gaubert, and E. Goubault. Coupling Policy Iteration with Semi-Definite Relaxation to Compute Accurate Numerical Invariants in Static Analysis. *Logical Methods in Computer Science*, 8(1), 2012.

[4] X. Allamigeon. *Static Analysis of Memory Manipulations by Abstract Interpretation — Algorithmics of Tropical Polyhedra, and Application to Abstract Interpretation*. PhD thesis, École Polytechnique, Palaiseau, France, November 2009.

[5] A. Auslender and M. Teboulle. *Asymptotic Cones and Functions in Optimization and Variational Inequalities*. Springer Science & Business Media, 2006.

[6] H. Bazille, O. Bournez, W. Gomaa, and A. Pouly. On The Complexity of Bounded Time Reachability for Piecewise Affine Systems. In *Reachability Problems: 8th International Workshop, RP 2014, Oxford, UK, September 22-24, 2014. Proceedings*, pages 20–31, Cham, 2014. Springer International Publishing.

[7] I. M. Bomze, W. Schachinger, and G. Uchida. Think Co(mpletely)positive ! Matrix Properties, Examples and a Clustered Bibliography on Copositive Optimization. *Journal of Global Optimization*, 52(3):423–445, 2012.

[8] S. Bundfuss and M. Dür. An Adaptive Linear Approximation Algorithm for Copositive Programs. *SIAM J. on Optimization*, 20(1):30–53, March 2009.

[9] A. Costan, S. Gaubert, E. Goubault, M. Martel, and S. Putot. A Policy Iteration Algorithm for Computing Fixed Points in Static Analysis of Programs. In *Computer aided verification*, pages 462–475. Springer, 2005.

[10] P. H. Diananda. On Non-negative Forms in Real Variables Some or All of Which are Non-negative. *Mathematical Proceedings of the Cambridge Philosophical Society*, 58:17–25, 1 1962.

[11] Gang Feng. Stability Analysis of Piecewise Discrete-time Linear Systems. *IEEE Transactions on Automatic Control*, 47(7):1109, 2002.

[12] Giancarlo Ferrari-Trecate, Francesco Alessandro Cuzzola, Domenico Mignone, and Manfred Morari. Analysis of Discrete-time Piecewise Affine and Hybrid Systems. *Automatica*, 38(12):2139–2146, 2002.

[13] S. Gaubert, E. Goubault, A. Taly, and S. Zennou. Static Analysis by Policy Iteration on Relational Domains. In *Programming Languages and Systems*, pages 237–252. Springer, 2007.

[14] T. Gawlitza, H. Seidl, A. Adjé, S. Gaubert, and E. Goubault. Abstract Interpretation Meets Convex Optimization. *J. Symb. Comput.*, 47(12):1416–1446, 2012.

[15] A. J. Hoffman and R. M. Karp. On Nonterminating Stochastic Games. *Management Science*, 12(5):359–370, 1966.

[16] R. A. Howard. *Dynamic Programming and Markov Processes*. MIT Press, Cambridge, MA, 1960.

[17] M. Johansson. On Modeling, Analysis and Design of Piecewise Linear Control Systems. In *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on*, volume 3, pages III–646–III–649 vol.3, May 2003.

[18] D.H. Martin and D.H. Jacobson. Copositive Matrices and Definiteness of Quadratic Forms Subject to Homogeneous Linear Inequality Constraints. *Linear Algebra and its Applications*, 35(0):227 – 258, 1981.

[19] D. Massé. Proving Termination by Policy Iteration. *Electronic Notes in Theoretical Computer Science*, 287(0):77 – 88, 2012. Proceedings of the Fourth International Workshop on Numerical and Symbolic Abstract Domains, NSAD 2012.

[20] J. E. Maxfield and H. Minc. On the Matrix Equation $X'X = A$. *Proceedings of the Edinburgh Mathematical Society (Series 2)*, 13:125–129, 12 1962.

[21] D. Mignone, G. Ferrari-Trecate, and M. Morari. Stability and Stabilization of Piecewise Affine and Hybrid systems: an LMI approach. In *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, volume 1, pages 504–509 vol.1, 2000.

[22] T. S. Motzkin. Two Consequences of the Transposition Theorem on Linear Inequalities. *Econometrica*, 19(2):184–185, 1951.

[23] SV Rakovic, P Grieder, M Kvasnica, DQ Mayne, and M Morari. Computation of Invariant Sets for Piecewise Affine Discrete Time Systems Subject to Bounded Disturbances. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 2, pages 1418–1423. IEEE, 2004.

[24] M. V. Ramana and P. M. Pardalos. Semidefinite Programming. *Interior point methods of mathematical programming*, 5:369–398, 1997.

[25] P. Roux, R. Jobredeaux, P.-L. Garoche, and E. Feron. A Generic Ellipsoid Abstract Domain for Linear Time Invariant Systems. In *Hybrid Systems: Computation and Control (part of CPS Week 2012), HSCC'12, Beijing, China, April 17-19, 2012*, pages 105–114, 2012.

[26] P. Schrammel and P. Subotic. Logico-Numerical Max-Strategy Iteration. In Roberto Giacobazzi, Josh Berdine, and Isabella Mastroeni, editors, *Verification, Model Checking, and Abstract Interpretation*, volume 7737 of *Lecture Notes in Computer Science*, pages 414–433. Springer Berlin Heidelberg, 2013.

[27] S. Schupp, E. Ábrahám, X. Chen, I. Ben Makhlouf, G. Frehse, S. Sankaranarayanan, and S. Kowalewski. Current Challenges in the Verification of Hybrid Systems. In *Cyber Physical Systems. Design, Modeling, and Evaluation - 5th International Workshop, CyPhy 2015, Amsterdam, The Netherlands, October 8, 2015, Proceedings*, pages 8–24, 2015.

[28] P. Sotin, B. Jeannet, F. Védrine, and E. Goubault. Policy iteration within logico-numerical abstract domains. In *Automated Technology for Verification and Analysis*, pages 290–305. Springer, 2011.

[29] A. Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific J. Math.*, 5(2):285–309, 1955.

[30] S. A. Vavasis. Quadratic programming is in NP. *Information Processing Letters*, 36(2):73 – 77, 1990.