Robustification and Parametrization of Switching Controllers for a Class of Set Invariance Problems [⋆]

Liren Yang and Necmiye Ozay

Electrical Engineering and Computer Science Department,
University of Michigan, Ann Arbor, MI 48109 USA
(e-mail: yliren,necmiye@umich.edu).

Abstract: In this paper, we consider the robustification and parametrization of an invariance switching controller with respect to a scalar variable (parameter) that affects the system dynamics. By robustification, we mean searching for a switching controller that guarantees invariance of a set, under a large enough range of the parameter values. In case such a robust controller does not exist, we do parametrization, i.e., searching for a collection of controllers, each one robust to a smaller range of parameter values. A parametrized controller can be applied in real-time by picking the appropriate switching surfaces based on the measurement of the parameter. To be more specific, assuming (i) the system dynamics is affine, and is monotone in the considered parameter, (ii) the invariance switching controller is defined on a rectangle in the state space, we show that the robustification and parametrization problems can be reduced to solving a sequence of linear programming problems. The proposed approach is illustrated by several numerical examples.

Keywords: set invariance, switched affine systems, robustification, parametrization of controllers

1. INTRODUCTION

Control systems in many safety-critical applications, for example in automotive and aerospace domain, are required to keep the states of the system away from certain unsafe regions. For instance, in an autonomous car, the unsafe region may be determined by the distance to the other vehicles or pedestrians; or in an aircraft, the unsafe region may be determined by flight envelopes or runway boundaries. These hard state constraints can be enforced by designing controllers that render the complement of the unsafe regions invariant. Hence, set invariance is crucial in many safety-critical applications (Blanchini (1999)).

We consider controlled invariant sets of switched systems, systems with discrete actuators or those that can switch among predetermined low-level controllers (Liberzon (2012)). Often times, there are uncertain parameters in system dynamics. Given a controlled invariant set for a switched system and a nominal switching controller that renders this set invariant, for some nominal value of the uncertain parameters, we are interested in the following question: for what ranges of the uncertain parameters, can we perturbed the controller parameters and/or the invariant set itself so that the invariance is preserved? This question is important for understanding the robustness of the controller to parameter variations. Secondly, the question is computationally interesting because searching for perturbations of the nominal controller allows us to avoid the potentially expensive computations required for obtaining these controllers directly from scratch. Moreover, if the controllers are implemented as a look-up table; since our construction of the perturbed controllers preserves the size and structure of the table and just changes the values in it, this facilitates code reuse and can reduce certification efforts (Tsai et al. (2005)). Finally, the question is also relevant for compositional synthesis, in which case the uncertain parameter is indeed an output of another subsystem (Kim et al. (2016); Smith et al. (2016)). Finding the maximal range of uncertain parameters against which invariance can be preserved is similar to finding least restrictive assumptions in contract-based design (Benvenuti et al. (2008); Chatterjee et al. (2008)), though we use a different formalism for representing the assumptions.

We limit our attention to rectangular invariant sets and switched affine dynamics that monotonically depend on a scalar uncertain parameter. Rectangular invariant sets naturally arise when considering monotone or multiaffine systems (Belta and Habets (2006); Abate et al. (2009); Meyer et al. (2016); Sadraddini and Belta (2016)). They are also common when the invariant set is obtained via an abstraction-based synthesis technique (Nilsson and Ozay (2014); Yang et al. (2016); Coogan and Arcak (2015)). For this class of dynamics and invariant sets, we show that a switching controller with rectangular switching surfaces can be robustified or parametrized using a sequence of linear programs. A notable property of the proposed linear program is that its size depends only linearly on the state-space dimension. We illustrate the proposed approach on several examples.

2. PRELIMINARIES

Let $\mathbb{R}^n$ be n dimensional Euclidean space. For a set $S \subseteq \mathbb{R}^n$, let ∂S be its boundary, $\text{int}(S)$ be its interior, $\text{relint}(S)$

[⋆] This work is supported in part by Ford Motor Co.

be its relative interior, and $\mathrm{cl}(S)$ be its closure. For a point $x \in \mathbb{R}^n$, x_i denotes its i^{th} component.

2.1 Rectangle & Rectangular Boundary Cover

In the paper we consider controlled invariance of a rectangle. In particular, the controller is defined on the boundary of the considered rectangle. In what follows, we define rectangle, boundary cover of a rectangle, and perturbation of a boundary cover.

Rectangle A set $R \subseteq \mathbb{R}^n$ is a *rectangle* if $R = \{x \in \mathbb{R}^n \mid l_i \leq x_i \leq u_i, i = 1, \dots, n\}$. The minimal and maximal point of rectangle R are defined to be $l(R) := [l_1, \dots, l_n]^T$, $u(R) := [u_1, \dots, u_n]^T$. A rectangle R is *full dimensional* if $l_i < u_i$ for all i. If $l_i = u_i$ for some i, i is called a *flat dimensions* of rectangle R. In particular, if rectangle R has exactly one flat dimension i, we write $flat(R) = i$. For rectangle R, define its *lower facet* and *upper facet* along the i^{th} dimension to be

$$L_i(R) := \{x \in R \mid x_i = l_i\}, \tag{1}$$
$$U_i(R) := \{x \in R \mid x_i = u_i\}. \tag{2}$$

Let $L(R) = \bigcup_{i=1}^n L_i(R)$ be the lower boundary of set R, and $U(R) = \bigcup_{i=1}^n U_i(R)$ be the upper boundary of R. It can be easily shown that $\partial R = L(R) \cup U(R)$. The definitions of $l(R)$, $u(R)$, $L_i(R)$, $U_i(R)$ are illustrated on a 3-dimensional example in Fig. 1.

Rectangular Boundary Cover Suppose R is a rectangle, $\mathcal{F} = \{F_1, \dots, F_N\} \subseteq 2^{\partial R}$ is called a *rectangular boundary cover* of R ("cover of R" for short) if F_j's are rectangles such that $\bigcup_{j=1}^N F_j = \partial R$. Moreover, a cover $\mathcal{F}$ of R is called *proper* if (i) F_j has exactly one flat dimension for all j, and (ii) $\mathrm{relint}(F_j) \cap \mathrm{relint}(F_{j'}) = \emptyset$ if $j \neq j'$. In what follows we will only consider proper covers. Suppose R is a full dimensional rectangle, for each F_j from proper cover $\mathcal{F}$ of R, since F_j's are rectangles with exactly one flat dimension, we have either $F_j \subseteq L(R)$ or $F_j \subseteq U(R)$. Let $normal(F_j) \in \mathbb{R}^n$ denote the *normal vector* of facet F_j, $normal(F_j)$ is defined element-wisely as follows:

$$\big(normal(F_j)\big)_i = \begin{cases} 1 & \text{if } F_j \subseteq L_i(R) \\ -1 & \text{if } F_j \subseteq U_i(R) \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

The definition in (3) guarantees the normal vector of F_j always points inwards the set R.

Perturbation of a Cover Let $\mathcal{F} = \{F_1, \dots, F_N\}$ be a proper cover of rectangle R. $\mathcal{F}$ is defined by $l(F_j)$ and $u(F_j)$ for each rectangle F_j. Therefore one can "perturb" the cover $\mathcal{F}$ by perturbing the values of $l(F_j)$ and $u(F_j)$. The following proposition provides a sufficient condition ensuring that R is still properly covered after perturbing $l(F_j)$ and $u(F_j)$.

Proposition 1. Let $\mathcal{F} = \{F_1, \dots, F_N\}$ be a proper cover of rectangle R, and let $\widetilde{\mathcal{F}} = \{\widetilde{F}_1, \dots, \widetilde{F}_N\}$ be a collection of rectangles with exactly one flat dimension. $\widetilde{\mathcal{F}}$ also properly covers set R if the followings hold for all indices j and j':

(1) $flat(F_j) = flat(\widetilde{F}_j)$,

(2) if $F_j \cap F_{j'} \neq \emptyset$, the order of $l_i(\widetilde{F}_j)$, $l_i(\widetilde{F}_{j'})$, $u_i(\widetilde{F}_j)$, $u_i(\widetilde{F}_{j'})$ is the same as $l_i(F_j)$, $l_i(F_{j'})$, $u_i(F_j)$, $u_i(F_{j'})$ [1],

(3) $l_{flat(F_j)}(F_j) = l_{flat(\widetilde{F}_j)}(\widetilde{F}_j)$.

Proof. Pick facet $L_i(R)$ as an example. Define index set $\underline{J}_i$ to be such that $j \in \underline{J}_i \Leftrightarrow F_j \subseteq L_i(R)$. We first show that $\bigcup_{j \in \underline{J}_i} \widetilde{F}_j = L_i(R)$. Since a similar result holds for $U_i(R)$ and dimension index i is arbitrary, this will prove $\partial R = \bigcup_{i=1}^n \big(L_i(R) \cup U_i(R)\big) = \bigcup_{j=1}^N \widetilde{F}_j$.

In order to show $\bigcup_{j \in \underline{J}_i} \widetilde{F}_j = L_i(R)$, we will show that (i) $\bigcup_{j \in \underline{J}_i} \widetilde{F}_j$ is a rectangle, (ii) the extreme values of this rectangle are the same as those of $L_i(R)$.

(i) Let $j_1, j_P \in \underline{J}_i$ be such that $l(L_i(R)) \in L_{j_1}$ and $u(L_i(R)) \in L_{j_P}$. By condition (1) (3), we know for all $j \in \underline{J}_i$, $\widetilde{F}_j$ lie on the same affine space as $L_i(R)$. This affine space is given by $\{x \in \mathbb{R}^n \mid x_i = l_i(R)\}$. Apply Lemma 1 in Appendix A on this affine space, we know set $\widetilde{L}_i := \bigcup_{j \in \underline{J}_i} \widetilde{F}_j$ is a rectangle, and

$$\widetilde{L}_i = \{x \in \mathbb{R}^n \mid x_i = l_i(R), \quad l_{i'}(\widetilde{F}_{j_1}) \leq x_{i'} \leq u_{i'}(\widetilde{F}_{j_P}) \text{ for } i' \neq i\}. \tag{4}$$

(ii) Next we show for all $i' \neq i$, $l_{i'}(\widetilde{F}_{j_1}) = l_{i'}(R)$. Since point $l(L_i(R)) \in F_{j_1}$, F_{j_1} must intersects with some $F_{j'} \subseteq L_{i'}(R)$, and

$$l_{i'}(F_{j_1}) = l_{i'}(F_{j'}) = l_{i'}(R). \tag{5}$$

By condition (2), this gives $l_{i'}(\widetilde{F}_{j_1}) = l_{i'}(\widetilde{F}_{j'})$. But $l_{i'}(\widetilde{F}_{j'}) = l_{i'}(R)$ because of condition (3), we hence showed $l_{i'}(\widetilde{F}_{j_1}) = l_{i'}(R)$. A similar argument can be applied to show upper values $u_{i'}(\widetilde{F}_{j_P}) = u_{i'}(R)$, thus the extreme values of $\widetilde{L}_i$ dfined by (4) are the same as $L_i(R)$, and this finishes the proof. ■

If two covers $\mathcal{F}$ and $\widetilde{\mathcal{F}}$ satisfy the conditions in Proposition 1, $\widetilde{\mathcal{F}}$ is called a *perturbation* of cover $\mathcal{F}$. We will use perturbation of a cover to define "perturbation" of a controller in the following parts.

2.2 Switched System

A general continuous-time switched system is governed by the following differential equation:

$$\begin{aligned} &\dot{x} = f^a(x, v, d), \\ &x \in X \subseteq \mathbb{R}^n, a \in A, \\ &v \in V \subseteq \mathbb{R}^m, d \in D \subseteq \mathbb{R}^p, \end{aligned} \tag{6}$$

where x is the state variable, a is the control action from a finite set A, v is the measured external input (i.e., parameter) and d is the disturbance. Let f_i^a denote the i^{th} component of vector field f^a.

In this paper, we consider switched systems that satisfy the following assumptions:

(A1) X, V, D are rectangles;

(A2) $V \subseteq \mathbb{R}$, thus V is an interval by (A1);

(A3) for all $a \in A$, f^a is affine in x;

(A4) for all $a \in A$, f_i^a is continuously differentiable and has sign stable partial derivatives with respect to x, v, d in $X \times V \times D$.

Systems satisfying assumption (A4) are within the class of mixed monotone systems (Coogan and Arcak (2015)).

[1] That is, the inequality between these values is preserved strictly.

2.3 Rectangular Controlled Invariant Set

In this part, we define controlled invariance of a set and the class of controllers considered in this paper, and then give a necessary and sufficient condition to check the controlled invariance of a rectangle under the considered controllers.

Controlled Invariant Set A vector $y \in \mathbb{R}^n$ is called a feasible direction of a set $S \subseteq \mathbb{R}^n$ at $x \in S$ if there exists $\varepsilon > 0$ such that $x + \delta y \in S$ for all $\delta \leq \varepsilon$. The tangent cone of a set S at x is defined to be $T_S(x) := \mathrm{cl}(\{y \mid y \text{ is feasible direction of } S \text{ at } x\})$.

Given a switched system in form of (6) with state space X and action set A, let closed set $S \subseteq X$. A *switching controller* is a map $K : \partial S \to 2^A \setminus \emptyset$. Set S is called *controlled invariant* under switching controller K at parameter v if (Blanchini (1999))

$$\forall x \in \partial S, a \in K(x), d \in D : f^a(x, v, d) \in T_S(x). \tag{7}$$

Remark 1. Note that controller K maps a state $x \in \partial S$ to a nonempty set of control actions. We thus have “ambiguity” in the sense that there might be multiple controls at a point $x \in \partial S$. For controller K to be robust to such ambiguity, we need $f^a(x, v, d) \in T_S(x)$ for all $a \in K(x)$ in Eq. (7).

Valid Switching Function & Controlled Invariance of a Rectangle Consider a proper cover $\mathcal{F}$ of a rectangle R, and a switched system in form of (6) with action set A. We define *switching function* $a : \mathcal{F} \to A$ that assigns a control action to each $F_j \in \mathcal{F}$. F_j equipped with a control is called a *switching facet*. A cover $\mathcal{F}$ together with a switching function a induces a switching controller $K_{\mathcal{F},a} : \partial R \to 2^A$ in the following sense:

$$\forall x \in \partial R : K_{\mathcal{F},a}(x) = \{a(F_j) \mid x \in F_j, F_j \in \mathcal{F}\}. \tag{8}$$

A switching function a is called *valid* if $a(F_j) = a(F_{j'})$ for all $F_j, F_{j'} \in \mathcal{F}$ satisfying the following:

$$\begin{aligned} &\mathit{flat}(F_j) \neq \mathit{flat}(F_{j'}),\ F_j \cap F_{j'} \neq \emptyset, \\ &\forall x \in F_j, y \in F_{j'} : \mathit{normal}(F_j)^T(x - y) \leq 0, \\ &\qquad\qquad \mathit{normal}(F_{j'})^T(y - x) \leq 0. \end{aligned} \tag{9}$$

If switching facet F_j and F_j' satisfy condition (9), we say F_j and $F_{j'}$ form a *convex wedge*. The right part of Fig. 1 shows an example of two facets forming a convex wedge.

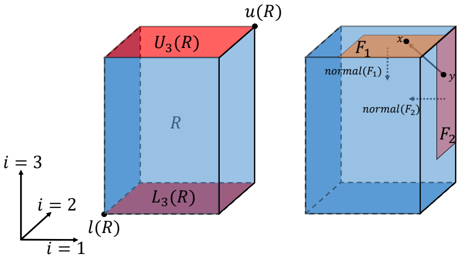

Fig. 1. Left: illustration of minimal point $l(R)$, maximal point $u(R)$, lower facet $L_i(R)$, upper facet $U_i(R)$ along the 3rd dimension. Right: two facets F_1 (green) and F_2 (red) form a convex wedge.

Proposition 2. Given a switched system in the form of (6) with state space X and action set A, a rectangle $R \subseteq X$ with a proper cover $\mathcal{F} = \{F_1, \dots, F_N\}$, and a valid switching function $a : \mathcal{F} \to A$, let $K_{\mathcal{F},a}$ be the switching controller induced by $\mathcal{F}$ and a. Set R is controlled invariant under controller $K_{\mathcal{F},a}$ for a nominal parameter v if and only if for all $F \in \mathcal{F}$:

$$\mathit{normal}(F)^T f^{a(F)}(x, v, d) \geq 0 \quad \forall x \in F, d \in D. \tag{10}$$

Proposition 2 says: if the controller is induced by a valid switching function, it is necessary and sufficient to verify the invariance of set R by checking Eq. (7) on each facet F_j from a cover. There is no need to check condition (7) on ridges, edges, vertices, etc. of set R. Note that condition given by Eq. (10) is necessary for invariance but not sufficient without validity of switching function a. For not valid switching functions, it is required to check Eq. (10) for all lower dimensional faces of the set R, which are exponentially many. Therefore, constraining the search to valid switching functions, significantly improves the computational complexity of invariance check. We also note that valid switching function assumption is not too restrictive in that whenever the set R can be rendered contractive, there exists a valid switching function.

2.4 Perturbation of a Switching Controller

Let $K_{\mathcal{F},a}$ be a switching controller induced by a proper cover $\mathcal{F}$ of a rectangle R and a switching function a defined on $\mathcal{F}$. Let $\widetilde{\mathcal{F}} = \{\widetilde{F}_1, \dots, \widetilde{F}_N\}$ be a perturbation of cover $\mathcal{F} = \{F_1, \dots, F_N\}$. Define $\widetilde{a} : \widetilde{\mathcal{F}} \to A$ to be such that $\widetilde{a}(\widetilde{F}_j) = a(F_j)$, that is, the control action assigned to a facet is preserved after perturbation.

Proposition 3. Let $\mathcal{F}$ be a proper cover of rectangle R, and let a be a switching function defined on $\mathcal{F}$. Suppose $\widetilde{\mathcal{F}}$ is a perturbation of $\mathcal{F}$, the corresponding switching function $\widetilde{a}$ (defined as above) is valid if a is valid.

Proof. Consider condition (9). First by definition of perturbed cover $\widetilde{\mathcal{F}}$, $\mathit{flat}(\widetilde{F}_j) = \mathit{flat}(F_j)$. Since perturbation $\widetilde{\mathcal{F}}$ is still a cover of the rectangle R, for any $\widetilde{F}_j, \widetilde{F}_{j'} \in \widetilde{\mathcal{F}}$ satisfying $\mathit{flat}(\widetilde{F}_j) \neq \mathit{flat}(\widetilde{F}_{j'})$, we have $\mathit{normal}(\widetilde{F}_j)^T(x - y) \leq 0$ and $\mathit{normal}(\widetilde{F}_{j'})^T(y - x) \leq 0$ for all $x \in \widetilde{F}_j, y \in \widetilde{F}_{j'}$. Finally it can be shown that perturbing the cover $\mathcal{F}$ does not create new intersections. That is, if $F_j, F_{j'} \in \mathcal{F}$ are such that $F_j \cap F_{j'} = \emptyset$, we have $\widetilde{F}_j \cap \widetilde{F}_{j'} = \emptyset$. Hence if two facets $\widetilde{F}_j, \widetilde{F}_{j'}$ from the perturbed cover satisfy condition (9), their corresponding facets $F_j, F_{j'}$ in the original cover also satisfy condition (9). By validity of a we have $a(F_j) = a(F_{j'})$. This implies $\widetilde{a}(\widetilde{F}_j) = \widetilde{a}(\widetilde{F}_{j'})$ by definition of $\widetilde{a}$. Therefore $\widetilde{a}$ is valid. ■

The perturbed cover $\widetilde{\mathcal{F}}$ and its corresponding switching function $\widetilde{a}$ induce a new switching controller $K_{\widetilde{\mathcal{F}},\widetilde{a}}$. Controller $K_{\widetilde{\mathcal{F}},\widetilde{a}}$ is called a *perturbation* of controller $K_{\mathcal{F},a}$. We denote the set of all the perturbed controllers of $K_{\mathcal{F},a}$ to be $\mathcal{K}_{\mathcal{F},a}$.

3. PROBLEM STATEMENT

We are in a position to define the problem considered in this paper, with the definitions and notations given in section 2.

Problem 1. (Parametrization of Switching Controller) Consider a switched system governed by differential equation in (6), satisfying assumptions (A1)-(A4). Let $S \subseteq X$ be a set that is controlled invariant under a given controller $K_{\hat{\mathcal{F}},\hat{a}}$ at nominal parameter $\hat{v}$. Our goal is to find

(i) a large enough interval $[\underline{v}, \overline{v}] \subseteq V$ that contains $\hat{v}$,

(ii) a mapping $\pi : [\underline{v}, \overline{v}] \to \mathcal{K}_{\hat{\mathcal{F}},\hat{a}}$ such that S is controlled invariant under controller $\pi(v)$ at v.

Problem 2. (Robustification of Switching Controller) The same as Problem 1 except that we want $\pi(v_1) = \pi(v_2)$ for all $v_1, v_2 \in [\underline{v}, \overline{v}]$.

Note that in both problems we assume that there already exists a controller working for nominal parameter $\hat{v}$. Whenever we can find a controller robust over $v \in [\underline{v}, \overline{v}]$, there is no need to search for a parametrized controller over the same interval. In this case, solving Problem 2 (robustification) is enough. However, it may not always be possible to find a robust controller over a large enough range, in which case, parametrization is required.

4. SOLUTION APPROACH

We solve Problem 1 and Problem 2 by solving a sequence of linear programming problems. The proposed approach can be decomposed into an inner part formulating the linear programming problems, and an outer loop doing line search. To be specific, we start from the given set R that is controlled invariant under switching controller $K_{\hat{\mathcal{F}},\hat{a}}$ at nominal parameter $\hat{v}$. A linear programming feasibility problem parametrized w.r.t. v, denoted as $P_{\hat{\mathcal{F}},\hat{a}}(v)$, is formulated based on $\hat{\mathcal{F}}$, $\hat{a}$ and the knowledge of the dynamics. We proceed by solving a series of such linear program for different values of parameter v. A feasible solution of problem $P_{\hat{\mathcal{F}},\hat{a}}(v)$ defines a cover $\mathcal{F}$ and a switching function a, from which a controller $K_{\mathcal{F},a}$ can be extracted. Under controller $K_{\mathcal{F},a}$, set R is guaranteed to be invariant at parameter v. By proceding this way we can parametrize controller $K_{\mathcal{F},a}$ w.r.t. v. At the end of this procedure, a condition is given to check if robustification is possible on the overall range of v.

In what follows, we first give the formulation of the linear programming feasibility problem, then we show how to solve robustification and parametrization problem by solving a sequence of such linear programming problems. Pseudo codes summarizing the overall procedure are also given.

4.1 Finding Feasible Controller by Linear Programming

Given a rectangle set R, let $\hat{\mathcal{F}}$ be a proper cover of R, $\hat{a}$ be a valid switching function defined on $\hat{\mathcal{F}}$, and $K_{\hat{\mathcal{F}},\hat{a}}$ be the switching controller induced by $\hat{\mathcal{F}}$, $\hat{a}$. Suppose that set R is controlled invariant at nominal parameter $\hat{v}$ under controller $K_{\hat{\mathcal{F}},\hat{a}}$. In what follows, A feasibility problem is formulated to give all the switching controllers that (i) are perturbed from nominal controller $K_{\hat{\mathcal{F}},\hat{a}}$, and (ii) make set R controlled invariant at a certain parameter v.

Define feasibility problem $P_{\hat{\mathcal{F}},\hat{a}}(v)$ with variable u, l and parameter v, and let $Feasible\big(P_{\hat{\mathcal{F}},\hat{a}}(v)\big)$ denote the feasible set of problem $P_{\hat{\mathcal{F}},\hat{a}}(v)$:

$$\begin{aligned}
\text{find}\ & l, u \\
\text{s.t.}\ & f^{\hat{a}(F)}_{flat(F)}(\underline{x}^F, v, \underline{d}^F) \geq 0, \forall F \in \hat{\mathcal{F}}, F \subseteq L(R), && \text{(C1)} \\
& f^{\hat{a}(F)}_{flat(F)}(\overline{x}^F, v, \overline{d}^F) \leq 0, \forall F \in \hat{\mathcal{F}}, F \subseteq U(R), && \text{(C2)} \\
& u^F_i > l^F_i, && \text{(C3)} \\
& \forall F \in \hat{\mathcal{F}}, i \in \{1, \dots, n\} \setminus \{flat(F)\}, \\
& && (P_{\hat{\mathcal{F}},\hat{a}}(v)) \\
& u^F_{flat(F)} = u_{flat(F)}(F), \\
& l^F_{flat(F)} = l_{flat(F)}(F), && \text{(C4)} \\
& \forall F \in \hat{\mathcal{F}}, \\
& \text{order of } l^F_i, l^E_i, u^F_i, u^E_i \text{ same as} \\
& l_i(F), l_i(E), u_i(F), u_i(E), && \text{(C5)} \\
& \forall E, F \in \hat{\mathcal{F}} : E \cap F \neq \emptyset.
\end{aligned}$$

Next, we explain in details the meaning of variables and each constraint in the above feasibility problem $P_{\hat{\mathcal{F}},\hat{a}}(v)$.

Variables l, u l and u are the variables of the feasibility problem, they are aggregation of l^{F_j} and u^{F_j} [2], i.e.,

$$l = [(l^{F_1})^T, \dots, (l^{F_N})^T]^T, \tag{11}$$

$$u = [(u^{F_1})^T, \dots, (u^{F_N})^T]^T. \tag{12}$$

l^F, u^F define a rectangle $\widetilde{F} := \{x \in \mathbb{R}^n \mid l^F_i \leq x_i \leq u^F_i\}$. The aggregated variable l, u hence define

$$\widetilde{\mathcal{F}}_{l,u} = \{\widetilde{F} \mid F \in \hat{\mathcal{F}}\}, \tag{13}$$

We can also define the corresponding switching function $\widetilde{a}_{l,u} : \widetilde{\mathcal{F}}_{l,u} \to A$ to be such that

$$\widetilde{a}_{l,u}(\widetilde{F}) = \hat{a}(F). \tag{14}$$

(C1) $f^{\hat{a}(F)}_{flat(F)}$ is the vector field component along the flat dimension of facet F, under control action $\hat{a}(F)$, which is equal to $\widetilde{a}_{l,u}(\widetilde{F})$. v is the parameter of problem $P_{\hat{\mathcal{F}},\hat{a}}$. $\underline{x}^F$, $\underline{d}^F$ are the minimizer of $f^{\hat{a}(F)}_{flatF)}$ over $\widetilde{F} \times D$, where D is the rectangular domain of disturbance d defined in Eq. (6). The constraint says, if F is part of lower boundary cover of S, the minimum value of $f^{a(F)}_{flatF)}$ on $\widetilde{F}$ is positive. In particular, $\underline{x}^F$, $\underline{d}^F$ are defined element-wise by

$$\underline{x}^F_i = \begin{cases} l^F_i \text{ if } \frac{\partial f^{a(F)}_{flat(F)}}{\partial x_i} \geq 0 \\ u^F_i \text{ if } \frac{\partial f^{a(F)}_{flat(F)}}{\partial x_i} < 0 \end{cases}, \underline{d}^F_j = \begin{cases} l_k(D) \text{ if } \frac{\partial f^{a(F)}_{flat(F)}}{\partial d_k} \geq 0 \\ u_k(D) \text{ if } \frac{\partial f^{a(F)}_{flat(F)}}{\partial d_k} < 0 \end{cases}$$

$$\forall F \in \hat{\mathcal{F}}, F \subseteq L(R), i \in \{1, \dots, n\}, k \in \{1, \dots, p\} \tag{15}$$

Note that $\underline{x}^F$ is defined by variables l^F, u^F, while $\underline{d}^F$ is a constant vector. It can be shown easily that $\underline{x}^F$, $\underline{d}^F$ minimizes $f^{a(F)}_{flatF)}$ on $\widetilde{F}$ under assumption (A4).

(C2) Similar to constraint C1, the maximum value of vector field component $f^{\hat{a}(F)}_{flatF)}$ on $\widetilde{F}$ is negative if F is part of a upper facet. $\overline{x}^F$, $\overline{d}^F$ maximizes $f^{\hat{a}(F)}_{flatF)}$ on $\widetilde{F}$, defined element-wisely by

$$\overline{x}^F_i = \begin{cases} u^F_i \text{ if } \frac{\partial f^{a(F)}_{flat(F)}}{\partial x_i} \geq 0 \\ l^F_i \text{ if } \frac{\partial f^{a(F)}_{flat(F)}}{\partial x_i} < 0 \end{cases}, \overline{d}^F_j = \begin{cases} u_k(D) \text{ if } \frac{\partial f^{a(F)}_{flat(F)}}{\partial d_k} \geq 0 \\ l_k(D) \text{ if } \frac{\partial f^{a(F)}_{flat(F)}}{\partial d_k} < 0 \end{cases}$$

$$\forall F \in \hat{\mathcal{F}}, F \subseteq U(R), i \in \{1, \dots, n\}, k \in \{1, \dots, p\} \tag{16}$$

[2] Note that l^F, u^F are different from $l(F)$, $u(F)$. the latter pair are the minimal and maximal points of F defined in section 2, while l^F, u^F are variables used to define the perturbation of $F \in \hat{\mathcal{F}}$.

By assumption (A3), $f^{\hat{a}}$ is affine. Hence constraints C1, C2 are linear inequalities.

(C3) The constraint ensures that $\widetilde{F} := \{x \in \mathbb{R}^n \mid l_i^F \leq x_i \leq u_i^F\}$ is well defined along all dimensions except $flat(F)$.

(C4) The constraint ensures that $flat(\widetilde{F}) = flat(F)$, and $l_{flat(\widetilde{F})}(\widetilde{F}) = l_{flat(F)}(F)$. Together with constraint C3, we know $\widetilde{F}$ has exactly one flat dimension.

(C5) The constraint ensures that facets F and E satisfy condition (2) in Proposition 1. Note that constraint C5 are set of linear inequalities.

The feasibility problem $P_{\hat{\mathcal{F}},\hat{a}}(v)$ gives all the switching controllers perturbed from nominal controller $K_{\hat{\mathcal{F}}.\hat{a}}$ that make set R controlled invariant at a certain parameter v. The result is formal stated by the following theorem.

Theorem 1. For any $l, u \in Feasible\big(P_{\hat{\mathcal{F}},\hat{a}}(v)\big)$, define $\widetilde{\mathcal{F}}_{l,u}$, $\widetilde{a}_{l,u}$ by Eq. (13) (14). $\widetilde{\mathcal{F}}_{l,u}$ is a proper cover of rectangle R. Moreover, let $K_{\widetilde{\mathcal{F}}_{l,u},\widetilde{a}_{l,u}}$ be the controller induced by $\widetilde{\mathcal{F}}_{l,u}$ and $\widetilde{a}_{l,u}$, the set $\big\{K_{\widetilde{\mathcal{F}}_{l,u},\widetilde{a}_{l,u}} \mid l, u \in Feasible\big(P_{\hat{\mathcal{F}},\hat{a}}(v)\big)\big\} = \big\{K \in \mathcal{K}_{\hat{\mathcal{F}},\hat{a}} \mid K \text{ makes set } R \text{ invariant}\big\}$.

Proof. First $\widetilde{\mathcal{F}}_{l,u}$'s defined by l, u satisfying constraint C3, C4, C5 are exactly the perturbations of nominal cover $\hat{\mathcal{F}}$. By interpretation of constraint C3 and C4, every $\widetilde{F} \in \widetilde{\mathcal{F}}_{l,u}$ is well defined, has exactly one flat dimension, and satisfies condition (1) and (3) in Proposition 1. By constraint C5, any two facet from $\widetilde{\mathcal{F}}_{l,u}$ also satisfy condition (2) in Proposition 1. Hence, by Proposition 1, we know $\widetilde{\mathcal{F}}_{l,u}$ also covers the rectangle R. Conversely, given any perturbation $\widetilde{\mathcal{F}}$ of nominal cover $\hat{\mathcal{F}}$, and $\widetilde{F}_j \in \widetilde{\mathcal{F}}$, one can check that $l := [l(\widetilde{F}_1)^T, \ldots, l(\widetilde{F}_N)^T]^T$, $u = [u(\widetilde{F}_1)^T, \ldots, u(\widetilde{F}_N)^T]^T$ satisfy constraint C3, C4, C5.

Secondly, set R is controlled invariant under the induced controller $K_{\widetilde{\mathcal{F}}_{l,u},\widetilde{a}_{l,u}}$. As already shown, $\widetilde{\mathcal{F}}_{l,u}$ is indeed a proper cover, by Proposition 3, the corresponding switching function $\widetilde{a}_{l,u}$ is always valid. Therefore, by Proposition 2, condition (7) is necessary and sufficient for invariance of set R. But condition (7) is captured exactly by constraint C1 C2, the feasible solutions of $P_{\hat{\mathcal{F}},\hat{a}}$ hence defines all the controllers within $\mathcal{K}_{\hat{\mathcal{F}},\hat{a}}$ that makes set the R controlled invariant. ■

4.2 Solving Robustification/Parametrization by Line Search

By Theorem 1, one can search for invariance controllers for different parameters $v \in [\underline{v}, \overline{v}]$ by solving feasibility problem $P_{\hat{\mathcal{F}},\hat{a}}(v)$. In order to solve Problem 1 (parametrization), however, it requires solving infinitely many such problems on interval $[\underline{v}, \overline{v}]$. To avoid intractability, we desire to (i) find a grid partition of interval $[\underline{v}, \overline{v}]$, denoted by $\mathcal{V} := \{\underline{v}, v_1, \ldots, v_M, \overline{v}\}$, and (ii) search for finitely many controllers, each one robustly guarantees the invariance of set R for all $v \in [v_k, v_{k+1}]$. Under assumption (A4) in section 2.2, the dynamics is monotone in parameter v. We can leverage monotonicity and obtain the following theorem.

Theorem 2. : Assume that

$$(l, u) \in Feasible\big(P_{\hat{\mathcal{F}},\hat{a}}(v_1)\big) \cap Feasible\big(P_{\hat{\mathcal{F}},\hat{a}}(v_2)\big) \qquad (17)$$

then l, u is also feasible to $P_{\hat{\mathcal{F}},\hat{a}}(v)$ for all $v \in [v_1, v_2]$ (assume $v_1 < v_2$ w.l.o.g.).

Proof. First note that the value of parameter v only affects constraint C1 and C2 in problem $P_{\hat{\mathcal{F}},\hat{a}}(v)$. Let $F \subseteq L(R)$ and take condition C1 as an example. Suppose there exists solution l, u satisfying condition (17), and let $\underline{x}^F, \underline{d}^F$ be the variables defined from l, u by Eq. (15), we have

$$f_{flat(F)}^{\hat{a}(F)}(\underline{x}^F, v_1, \underline{d}^F) \geq 0, \;\; f_{flat(F)}^{\hat{a}(F)}(\underline{x}^F, v_2, \underline{d}^F) \geq 0. \qquad (18)$$

Let $v \in [v_1, v_2]$. By assumption (A4), $\frac{\partial f_i^a}{\partial v}$ is sign stable on $X \times V \times D$. If $\frac{\partial f_i^a}{\partial v} \geq 0$ on $X \times V \times D$, $v \geq v_1$ implies $f_{flat(F)}^{\hat{a}(F)}(\underline{x}^F, v, \underline{d}^F) \geq f_{flat(F)}^{\hat{a}(F)}(\underline{x}^F, v_1, \underline{d}^F) \geq 0$; if $\frac{\partial f_i^a}{\partial v} \leq 0$, $v \leq v_2$ implies $f_{flat(F)}^{\hat{a}(F)}(\underline{x}^F, v, \underline{d}^F) \geq f_{flat(F)}^{\hat{a}(F)}(\underline{x}^F, v_2, \underline{d}^F) \geq 0$. That is, l, u satisfy constraint C1 at v in either case. A similar argument can be applied to constraint C2, and this finishes the proof. ■

By Theorem 2, we can expand our parametrization starting from $\mathcal{V} = \{\hat{v}\}$. Then we pick a v_i and solve $P_{\hat{\mathcal{F}},\hat{a}}(v_i)$, if its feasible region has nonempty intersection with the feasible region of $P_{\hat{\mathcal{F}},\hat{a}}(\hat{v})$, then we know that any u, l from this intersection induce a cover $\mathcal{F}$ and a switching function a, under which R is controlled invariant robustly for all $v \in [\hat{v}, v_i]$. We can proceed by expanding $\mathcal{V}$ in two directions. If the new linear programming problem $P_{\hat{\mathcal{F}},\hat{a}}(v_i)$ has empty intersection with all $\bigcup_{v \in \mathcal{V}} Feasible(P_{\hat{\mathcal{F}},\hat{a}}(v))$, then we have to shrink the step size. This procedure will continue until the step size is smaller than a predefined minimum step. Algorithm 2 gives the pseudo code for the above parametrization procedure, returning $\mathcal{V}, \pi$ to solve Problem 1. In particular, if

$$Feasible\Big(P_{\hat{\mathcal{F}},\hat{a}}\big(\min(\mathcal{V})\big)\Big) \cap Feasible\Big(P_{\hat{\mathcal{F}},\hat{a}}\big(\max(\mathcal{V})\big)\Big) \qquad (19)$$

then parametrization is actually not necessary because by Theorem 2 we can instead have a robust controller. A robust controller is better than a parametrized one because we do not need to switch between controllers according to the measurements. Algorithm 1 checks condition (19) and declare either Problem 1 (parametrization) or Problem 2 (robustification) is solved accordingly.

5. EXAMPLES

In this section, we illustrate the proposed approach by two numerical examples, one for parametrization, and one for robustification.

Example 1. The following is a switched affine system with three modes. The system satisfies assumptions (A1)-(A4).

$$\begin{aligned}
&\dot{x} = A_a x + K_a(v), \\
&x \in [0, 50] \times [0, 50], a \in \{1, 2, 3\}, V \in [1, 50], \\
&A_1 = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix}, A_2 = \begin{bmatrix} -1 & -0.5 \\ 0.1 & 1 \end{bmatrix}, A_3 = \begin{bmatrix} -1 & 0.5 \\ 0.5 & -1 \end{bmatrix}, \\
&K_1(v) = \begin{bmatrix} 30 \\ 40 \end{bmatrix}, K_2(v) = \begin{bmatrix} -12.5 + v \\ 33 \end{bmatrix}, K_3(v) = \begin{bmatrix} 10 - v \\ 1 \end{bmatrix}.
\end{aligned} \qquad (20)$$

A rectangular controlled invariant set $[10, 20] \times [10, 30]$ is found by abstraction-based synthesis (Nilsson and Ozay

Algorithm 1 $(\mathcal{V},\pi) = \text{ParametrizeRobustify}(\hat{v},\hat{\mathcal{F}},\hat{a},V)$

```
Require: nominal parameter $\hat{v}$, corresponding cover $\hat{\mathcal{F}}$
    and switching function $\hat{a}$, parameter set $V$
Ensure: finite set $\mathcal{V}$, mapping $\pi$
  $\varepsilon :=$ minimum allowable quantization scale of $\mathcal{V}$
  $\delta :=$ default quantization scale of $\mathcal{V}$
  Initialize $\mathcal{V} = \{\hat{v}\}$, $\pi(\hat{v}) = (\hat{\mathcal{F}},\hat{a})$
  $(\mathcal{V},\pi) = \text{ExpandForward}(\mathcal{V},\pi,\varepsilon,\delta,\hat{\mathcal{F}},\hat{a},V)$
  $(\mathcal{V},\pi) = \text{ExpandBackward}(\mathcal{V},\pi,\varepsilon,\delta,\hat{\mathcal{F}},\hat{a},V)$
  $\underline{v} := \min(\mathcal{V})$
  $\overline{v} := \max(\mathcal{V})$
  if $\mathit{Feasible}\big(P_{\hat{\mathcal{F}},\hat{a}}(\underline{v})\big) \cap \mathit{Feasible}\big(P_{\hat{\mathcal{F}},\hat{a}}(\overline{v})\big) \neq \emptyset$ then
      $\mathcal{V} = \{\underline{v},\overline{v}\}$
      $\pi(\overline{v}) = K_{\mathcal{F}_{u,l},a_{u,l}}$, where
      $(u,l) \in \mathit{Feasible}\big(P_{\hat{\mathcal{F}},\hat{a}}(\underline{v})\big) \cap \mathit{Feasible}\big(P_{\hat{\mathcal{F}},\hat{a}}(\overline{v})\big)$
      and $\mathcal{F}_{u,l}$, $a_{u,l}$ are defined by (13),(14)
      declare robustification
  else
      declare parametrization
  return $\mathcal{V},\pi$
```

Algorithm 2 $(\mathcal{V},\pi) = \text{ExpandForward}(\mathcal{V},\pi,\varepsilon,\delta,\hat{\mathcal{F}},\hat{a},V)$ (ExpandBackward can be defined similarly)

```
Require: finite set $\mathcal{V}$, mapping $\pi$,
    $\varepsilon$, minimum allowable quantization scale of $\mathcal{V}$, and $\delta$,
    default quantization scale of $\mathcal{V}$,
    cover $\hat{\mathcal{F}}$ and switching function $\hat{a}$ for a nominal parameter from set $V$
Ensure: Expanded $\mathcal{V}$ and mapping $\pi$
  Initialize $\delta^+ = \delta$
  while $\delta^+ > \varepsilon$ do
      $\overline{v} = \max(\mathcal{V})$
      $\overline{\overline{v}} = \overline{v} + \delta^+$
      if $\mathit{Feasible}\big(P_{\hat{\mathcal{F}},\hat{a}}(\overline{\overline{v}})\big) \cap \mathit{Feasible}\big(P_{\hat{\mathcal{F}},\hat{a}}(\overline{v})\big) \neq \emptyset$ and
      $\overline{\overline{v}} \leq \max(V)$ then
          $\mathcal{V} = \mathcal{V} \cup \{\overline{\overline{v}}\}$, $\delta^+ = \delta$
          $\pi(\overline{\overline{v}}) = K_{\mathcal{F}_{u,l},a_{u,l}}$, where
          $(u,l) \in \mathit{Feasible}\big(P_{\hat{\mathcal{F}},\hat{a}}(\overline{\overline{v}})\big) \cap \mathit{Feasible}\big(P_{\hat{\mathcal{F}},\hat{a}}(\overline{v})\big)$
          and $\mathcal{F}_{u,l}, a_{u,l}$ are defined as (13),(14)
      else
          $\delta^+ = \delta^+/2$.
  return $\mathcal{V},\pi$
```

(2014)) under nominal value $\hat{v} = 0$, then algorithm 1 is used to find a set of parametrized controllers over interval $[\underline{v},\overline{v}] = [0, 47.375]$. The grid of the interval is given by $\mathcal{V} = \{0, 3, 6, 9, \ldots, 42, 45, 46.5, 47.25, 47.375\}$. The left part of Fig. 2 shows in $X \times V$ space the controlled invariant set and the actions defined on its boundary. Each "slice" in the figure corresponds to the controller on a sub-interval $[v_k, v_{k+1}]$, where v_k, v_{k+1} are two consecutive numbers from grid $\mathcal{V}$. It can be seen that the switching surfaces change with parameter v. For this example, no robust controller can be found for $v \in [\min(\mathcal{V}), \max(\mathcal{V})]$.

Example 2. Consider the following switched affine model for thermal dynamics of an engine (Yang et al. (2016)):

$$\begin{aligned}\dot{x} =& A_a(v,d)x + K_a(v,d),\\ x \in& [380,400]\times[260,310],\ a\in\{1,2,3,4\},\\ v \in& [260,310],\ d\in[0.03,0.045]\times[1.5\times10^4, 1.9\times10^4],\end{aligned} \tag{21}$$

where A_a, K_a are defined by Eq. (22). In this example, state $x = [T_\mathrm{e}, T_\mathrm{r}]^T$, where T_e, T_r denote the engine temperature and radiator temperature, respectively. Parameter v is the ambient temperature T_a. Disturbance $d = [w, q]$ consists of the coolant flow velocity w and the heat q generated by engine combustion. Control $a = \{1, 2, 3, 4\}$ corresponds to four levels of cooling, from the most mild cooling ($a = 1$) to the most aggressive cooling ($a = 4$). Fig. 3 shows the results given by Algorithm 1 on Example 2. The left figure shows the parametrized controller plot in the $X \times V$ space, over the grid $\mathcal{V} = \{280.875, 281.25, 282, 285, 288, 291, 292.5, 292.875\}$. In this example a robust controller is also found and is shown in the right figure.

6. EXTENSION

The proposed approach does not change the size of the controlled invariant rectangle R for different parameter values. However, we can also have the set R's size vary with the parameter, by a modification to the feasibility problem $P_{\hat{\mathcal{F}},\hat{a}}(v)$:

(i) add variables l^R, u^R, and modify constraint C4 into

$$\begin{aligned} l^F_{flat(F)} = u^F_{flat(F)} = l^R_{flat(F)} \text{ if } F \subseteq L(R),\\ l^F_{flat(F)} = u^F_{flat(F)} = u^R_{flat(F)} \text{ if } F \subseteq U(R),\\ l^R_{flat(F)} < u^R_{flat(F)}, \end{aligned} \tag{C4'}$$

(iii) add constraints on l^R, u^R to avoid obtaining an invariant set that is too large,

(iv) maximize the volume of the obtained invariant set by minimizing $-\sum_{i=1}^{n} \log(u^R_i - l^R_i)$.

This extension is applied to the system in Example 1. The obtained controlled invariant set with varying size and the parametrized controller are shown in the right part of Fig. 2. In this example, we pick smaller step size on parameter v when expanding the parametrization, and we enlarge the size of the invariant set when expanding is blocked, instead of shrinking the step size.

In the future, we plan to extend the proposed framework to handle unions of rectangles. Although this extension is straightforward if one checks the invariance conditions on all lower dimensional faces (leading to exponentially many constraints in the feasibility problem), we are seeking more efficient solutions as is done in this paper for rectangles.

REFERENCES

Abate, A., Tiwari, A., and Sastry, S. (2009). Box invariance in biologically-inspired dynamical systems. *Automatica*, 45(7), 1601–1610.

Belta, C. and Habets, L. (2006). Controlling a class of nonlinear systems on rectangles. *IEEE Trans. on Automatic Control*, 51(11), 1749–1759.

Benvenuti, L., Ferrari, A., Mazzi, E., and Vincentelli, A.S. (2008). Contract-based design for computation and verification of a closed-loop hybrid system. In *Proc. of HSCC*, 58–71.

Blanchini, F. (1999). Survey paper: Set invariance in control. *Automatica*, 35(11), 1747–1767.

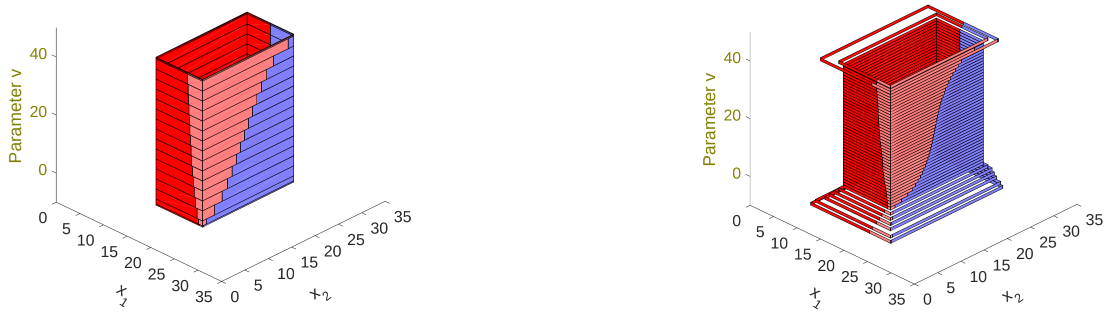

Fig. 2. Example 1. Parametrized controller that guarantees the invariance of rectangle R with fixing size (left), with varying size (right). Red: $a = 1$, pink: $a = 2$, blue: $a = 3$.

$$
\begin{aligned}
A_1(v,d) &= \begin{bmatrix} -0.133 - 1.133d_1 & 1.133d_1 \\ 4.25d_1 & -4.25d_1 - 4.269 \end{bmatrix}, &
A_2(v,d) &= \begin{bmatrix} -0.133 - 1.133d_1 & 1.133d_1 \\ 4.25d_1 & -4.25d_1 - 15.575 \end{bmatrix}, \\
K_1(v,d) &= \begin{bmatrix} 1.333 \times 10^{-3}d_2 + 0.133v \\ 4.269v \end{bmatrix}, &
K_2(v,d) &= \begin{bmatrix} 1.333 \times 10^{-3}d_2 + 0.133v \\ 15.575v \end{bmatrix}, \\
A_3(v,d) &= \begin{bmatrix} -0.133 - 4.533d_1 & 4.533d_1 \\ 17d_1 & -17d_1 - 4.269 \end{bmatrix}, &
A_4(v,d) &= \begin{bmatrix} -0.133 - 4.533d_1 & 4.533d_1 \\ 17d_1 & -17d_1 - 15.575 \end{bmatrix}, \\
K_3(v,d) &= \begin{bmatrix} 1.333 \times 10^{-3}d_2 + 0.133v \\ 4.269v \end{bmatrix}, &
K_4(v,d) &= \begin{bmatrix} 1.333 \times 10^{-3}d_2 + 0.133v \\ 15.575v \end{bmatrix}.
\end{aligned} \tag{22}
$$

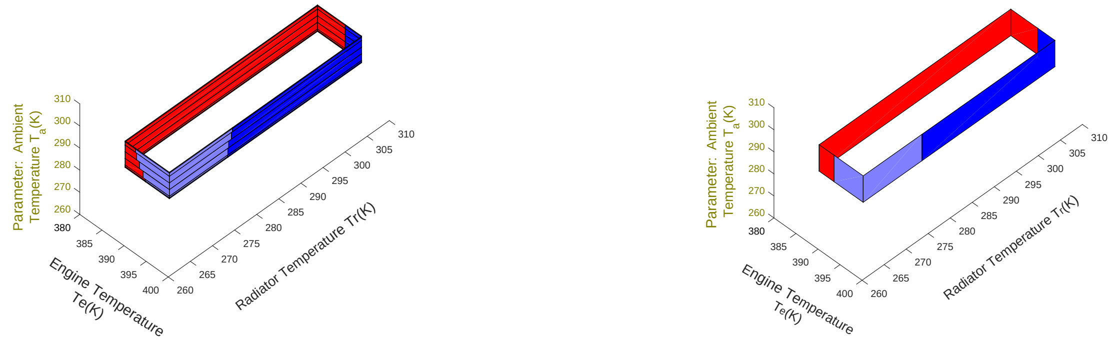

Fig. 3. Example 2: parametrized (left) and robustified (right) controller. Red: $u = 1$, light blue: $u = 3$, dark blue: $u = 4$.

Chatterjee, K., Henzinger, T.A., and Jobstmann, B. (2008). Environment assumptions for synthesis. In *CONCUR*, 147–161.

Coogan, S. and Arcak, M. (2015). Efficient finite abstraction of mixed monotone systems. In *Proc. of HSCC*, 58–67. ACM.

Kim, E.S., Arcak, M., and Seshia, S.A. (2016). Directed specifications and assumption mining for monotone dynamical systems. In *Proc. of HSCC*, 21–30. ACM.

Liberzon, D. (2012). *Switching in systems and control*. Springer Science & Business Media.

Meyer, P.J., Girard, A., and Witrant, E. (2016). Robust controlled invariance for monotone systems: application to ventilation regulation in buildings. *Automatica*, 70, 14–20.

Nilsson, P. and Ozay, N. (2014). Incremental synthesis of switching protocols via abstraction refinement. In *Proc. of IEEE CDC*, 6246–6253.

Sadraddini, S. and Belta, C. (2016). Safety control of monotone systems with bounded uncertainties. *arXiv preprint arXiv:1603.07419*.

Smith, S., Nilsson, P., and Ozay, N. (2016). Interdependence quantification for compositional control synthesis: An application in vehicle safety systems. In *Proc. of IEEE CDC*.

Tsai, W.T., Yu, L., Zhu, F., and Paul, R. (2005). Rapid embedded system testing using verification patterns. *IEEE software*, 22(4), 68–75.

Yang, L., Ozay, N., and Karnik, A. (2016). Synthesis of fault tolerant switching protocols for vehicle engine thermal management. In *Proc. of ACC*, 4213–4220.

Appendix A. LEMMA 1 AND ITS PROOF

The proof of Proposition 1 uses the following lemma.

Lemma 1. Given a rectangle $R \subseteq \mathbb{R}^n$, and a collection of rectangles $\{R_j\}_{j=1}^N$ satisfying

(a1) $R = \bigcup_{j=1}^N R_j$,

(a2) R_j's have disjoint interior, i.e., $\text{int}(R_j) \cap \text{int}(R_{j'}) = \emptyset$,

and a new collection of rectangles $\{\widetilde{R}_j\}_{j=1}^N$ satisfying

(b1) $R_j \cap R_{j'} \neq \emptyset \Rightarrow$ the order of $l_i(R_j), l_i(R_{j'}), u_i(R_j), u_i(R_{j'})$ is the same as the order of $l_i(\widetilde{R}_j), l_i(\widetilde{R}_{j'}), u_i(\widetilde{R}_j), u_i(\widetilde{R}_{j'})$ for all i,

then $\bigcup_{j=1}^M \widetilde{R}_j$ is still a rectangle.

Note that assumption (b1) induces a relation between rectangles with the same indices in collections $\{R_j\}_{j=1}^N$ and $\{\widetilde{R}_j\}_{j=1}^N$. This relation will be crucial in the proof.

Proof. We prove Lemma 1 by induction on dimension n.

1° For $n=1$, R and R_j's are intervals. By assumption (a1), we know $\bigcup_{j=1}^N R_j$ is a single interval. By assumption (a2) $\text{int}(R_j)\cap\text{int}(R_{j'})=\emptyset$, thus intervals $R_j=[l(R_j),u(R_j)]$ can be assumed to be sorted so that $u(R_j)=l(R_{j+1})$. By assumption (b1), $u(\widetilde{R}_j)=l(\widetilde{R}_{j+1})$, hence $\bigcup_{j=1}^N \widetilde{R}_j$ is also a single interval.

2° Assume Lemma 1 holds for n dimensional case. We will show that it holds when the dimension is $n+1$. To do so, we (i) define a rectangle $\widetilde{R}$ using the collection $\{\widetilde{R}_j\}_{j=1}^N$, (ii) show it is non-empty, and (iii) show $\bigcup_{j=1}^N \widetilde{R}_j=\widetilde{R}$ by induction hypothesis.

(i) First we define rectangle $\widetilde{R}$. By assumption (a1), R's minimum point $l(R)$ and maximum point $u(R)$ both belong to some R_j's. Suppose that $l(R)\in R_1$, and $u(R)\in R_N$, define

$$\widetilde{R}:=\{x\in\mathbb{R}^n \mid l_i(\widetilde{R}_1)\le x_i\le u_i(\widetilde{R}_N)\}. \tag{A.1}$$

(ii) Next we show that $l_i(\widetilde{R}_1)\le u_i(\widetilde{R}_N)$, that is, $\widetilde{R}$ is non-empty. Consider the sub-collection of rectangles $\{R_j\}_{j\in\overline{J}_i}$ such that $u_i(R_j)=u_i(R)$ for all $j\in\overline{J}_i$. Note that $N\in\overline{J}_i$. Any two rectangle R_{j_1}, R_{j_M} in this sub-collection can be connected by a line going through a sequence of distinct rectangles with indices $j_1,j_2,\dots,j_M\in\overline{J}_i$. Then, by assumption (b1), $u_i(\widetilde{R}_{j_1})=u_i(\widetilde{R}_{j_2})=\cdots=u_i(\widetilde{R}_{j_M})=u_i(\widetilde{R}_{j_N})$. In other words, for all $j\in\overline{J}_i$, the upper facets of $\widetilde{R}_j$ along the i^{th} dimension lie on the same affine space $\{x\in\mathbb{R}^n\mid x_i=u_i(\widetilde{R}_N)\}$. Similarly, by defining $\underline{J}_i$ to be such that $j\in\underline{J}_i\Leftrightarrow l_i(R_j)=l_i(R)$, we can show for all $j\in\underline{J}_i$, the lower facet of R_j's all lie on the affine space $\{x\in\mathbb{R}^n\mid x_i=l_i(\widetilde{R}_1)\}$.

Now, arbitrarily pick $\widetilde{R}_{j^*}$, we claim that $l_i(\widetilde{R}_1)\le l_i(\widetilde{R}_{j^*})\le u_i(\widetilde{R}_{j^*})\le u_i(\widetilde{R}_N)$. To prove this, let R_{j^*} be the corresponding rectangle in the original collection, and pick $x^*\in R_{j^*}$. Consider a line passing through x^* along the direction of the i^{th} dimension. This line crosses a sub-collection $\{R_{j_k}\}_{j_k\in J^*}$ of rectangles. Obviously $j^*\in J_i^*$. Note that R_{j_k}'s satisfy

$$\begin{aligned} l_i(R_{j_1})\le u_i(R_{j_1})=l_i(R_{j_2})\le\dots\\ \le u_i(R_{j_{P-1}})=l_i(R_{j_P})\le u_i(R_{j_P})\end{aligned} \tag{A.2}$$

where $P:=|J_i^*|$. By assumption (b1), we know that $\widetilde{R}_{j_k}$'s preserve this order, that is

$$\begin{aligned} l_i(\widetilde{R}_{j_1})\le u_i(\widetilde{R}_{j_1})=l_i(\widetilde{R}_{j_2})\le\dots\\ \le u_i(\widetilde{R}_{j_{P-1}})=l_i(\widetilde{R}_{j_P})\le u_i(\widetilde{R}_{j_P}).\end{aligned} \tag{A.3}$$

Again, noting that $j^*\in J_i^*$, Eq. (A.3) thus gives

$$l_i(\widetilde{R}_{j_1})\le l_i(\widetilde{R}_{j^*})\le u_i(\widetilde{R}_{j^*})\le u_i(\widetilde{R}_{j_P}). \tag{A.4}$$

Finally note that $L_i(R_{j_1})\subseteq L_i(R)$, and $U_i(R_{j_P})\subseteq U_i(R)$, thus $j_1\in\underline{J}_i$ and $j_P\in\overline{J}_i$. Therefore, we have $l_i(\widetilde{R}_{j_1})=l_i(\widetilde{R}_1)$, and $u_i(\widetilde{R}_{j_P})=u_i(\widetilde{R}_N)$, which when combined with Eq. (A.4) gives $l_i(\widetilde{R}_1)\le u_i(\widetilde{R}_N)$. Hence $\widetilde{R}$ defined in Eq. (A.1) is non-empty.

(iii) Note that dimension index i and rectangle index j^* in argument (ii) are arbitrary, thus Eq. (A.4) also shows that $\bigcup_{j=1}^N \widetilde{R}_j\subseteq\widetilde{R}$. Next we show $\widetilde{R}\subseteq\bigcup_{j=1}^N\widetilde{R}_j$. For this purpose, we show respectively $\partial\widetilde{R}\subseteq\bigcup_{j=1}^N\widetilde{R}_j$ and $\text{int}(\widetilde{R})\subseteq\bigcup_{j=1}^N\widetilde{R}_j$.

(iii-1) $\partial\widetilde{R}\subseteq\bigcup_{j=1}^N\widetilde{R}_j$.

Let $L_i(\widetilde{R})$ be the lower facet of rectangle $\widetilde{R}$ along the i^{th} dimension. We known by earlier argument that $\bigcup_{j=1}^N L_i(\widetilde{R}_j)$ lies on affine space $\{x\in\mathbb{R}^n\mid x_i=l_i(\widetilde{R})\}$. Moreover, we know $\bigcup_{j=1}^N L_i(\widetilde{R}_j)$ is a rectangle on that affine space by induction hypothesis. Denote this rectangle to be $\widetilde{L}_i:=\bigcup_{j=1}^N L_i(\widetilde{R}_j)$, we know $\widetilde{L}_i\subseteq L_i(\widetilde{R})$ because $\bigcup_{j=1}^N\widetilde{R}_j\subseteq\widetilde{R}$. To show $\widetilde{L}_i=L_i(\widetilde{R})$, we need $l_{i'}(\widetilde{L}_i)=l_{i'}(\widetilde{R})$ for all $i'\ne i$.

Consider facets of original rectangle R, $L_i(R)$ and $L_{i'}(R)$, there must be a rectangle R_j that contributes to both $L_i(R)$ and $L_{i'}(R)$, in the sense that $L_i(R_j)\subseteq L_i(R)$ and $L_{i'}(R_j)\subseteq L_i(R)$. Let $\widetilde{R}_j$ be the corresponding rectangle in the new collection, $\widetilde{R}_j$ contributes to both $\widetilde{L}_i$ and $\widetilde{L}_{i'}$. This hence implies $l_{i'}(\widetilde{L}_i)\le l_{i'}(\widetilde{R}_j)=l_{i'}(\widetilde{L}_{i'})=l_{i'}(\widetilde{R})$. Since the two facets $L_i(R)$ and $L_{i'}(R)$ are arbitrarily picked, we show $l_{i'}(\widetilde{L}_i)=l_{i'}(\widetilde{R})$ for all $i'\ne i$ (similar for upper values). Therefore $\widetilde{L}_i=L_i(\widetilde{R})$ (similar for upper facets) and $\partial\widetilde{R}=\bigcup_{i=1}^n\big(L_i(\widetilde{R})\cup U_i(\widetilde{R})\big)=\bigcup_{i=1}^n(\widetilde{L}_i\cup\widetilde{U}_i)\subseteq\bigcup_{j=1}^N\widetilde{R}_j$.

(iii-2) $\text{int}(\widetilde{R})\subseteq\bigcup_{j=1}^N\widetilde{R}_j$.

We prove this by contradiction. Assume otherwise, there exists a point $x^\circ\in\text{int}(\widetilde{R})$ but $x^\circ\notin\bigcup_{j=1}^N\widetilde{R}_j$. We know the ray $\{x\in\mathbb{R}^n\mid x=x^\circ+\lambda e_i,\lambda>0\}$[3] must intersect with some rectangles $\widetilde{R}_j$'s because by (iii-1) $\partial\widetilde{R}\subseteq\bigcup_{j=1}^N\widetilde{R}_j$. Let R_{j° be the first rectangle that intersects with this ray, and consider its lower facet $L_i(\widetilde{R}_{j^\circ})$. Facet $L_i(\widetilde{R}_{j^\circ})$ is "exposed" in the sense that $L_i(\widetilde{R}_{j^\circ})\not\subseteq\bigcup_{j=1,j\ne j^\circ}^N\widetilde{R}_j$. Let $L_i(R_{j^\circ})$ be the same lower facet of the corresponding rectangle in the original collection. We know, however, facet $L_i(R_{j^\circ})$ is fully covered because we can show it is not part of $L_i(R)$: since $x^\circ\in\text{int}(\widetilde{R})$, we have $l_i(L_i(\widetilde{R}_{j^\circ}))>x_i^\circ>l_i(\widetilde{R})$, but by assumption (b1), this implies $l_i(R_{j^\circ})>l_i(R)$ and hence $L_i(R_{j^\circ})$ cannot be part of $L_i(R)$.

Coming back to argument (iii-2), define index set J_i° to be such that $j\in J_i^\circ\Leftrightarrow R_j\cap L_i(R_{j^\circ})\ne\emptyset$ and $j\ne j^\circ$. Also define $S_j=R_j\cap L_i(R_{j^\circ})$ for $j\in J_i^\circ$. By assumption (b1) we know that $\widetilde{S}_j:=\widetilde{R}_j\cap L_i(\widetilde{R}_{j^\circ})$ are also nonempty sets. Moreover, using arguments from part (ii), we know $\widetilde{S}_j$'s lie on the same affine space as facet $L_i(\widetilde{R}_{j^\circ})$. Hence by induction hypothesis, $\bigcup_{j\in J_i^\circ}\widetilde{S}_j$ is a rectangle on that affine space. But by assumption (b1), $l(\widetilde{R}_{j^\circ}),u(\widetilde{R}_{j^\circ})\in\bigcup_{j\in J_i^\circ}\widetilde{S}_j$. Therefore $L_i(\widetilde{R}_{j^\circ})\subseteq\bigcup_{j\in J_i^\circ}\widetilde{S}_j\subseteq\bigcup_{j=1,j\ne j^\circ}^N\widetilde{R}_j$. But this contradicts with the fact that $L_i(\widetilde{R}_{j^\circ})$ is "exposed", i.e., $L_i(\widetilde{R}_{j^\circ})\not\subseteq\bigcup_{j=1,j\ne j^\circ}^N\widetilde{R}_j$. To this point the entire proof is completed. ■

[3] e_i is the i^{th} natural base.