Synthesizing Min-Max Control Barrier Functions For Switched Affine Systems *

Sara Kamali [a], Guillaume O. Berger [b], Sriram Sankaranarayanan [a]

[a]*University of Colorado Boulder, USA.*

[b]*UC Louvain, Belgium.*

Abstract

We study the problem of synthesizing non-smooth control barrier functions (CBFs) for continuous-time switched affine systems. Switched affine systems are defined by a set of affine dynamical modes, wherein the control consists of a state-based switching signal that determines the current operating mode. The control barrier functions seek to maintain the system state inside a control invariant set that excludes a given set of unsafe states. We consider CBFs that take the form of pointwise minima and maxima over a finite set of affine functions. Our approach uses ideas from nonsmooth analysis to formulate conditions for min- and max- affine control barrier functions. We show how a feedback switching law can be extracted from a given CBF. Next, we show how to automate the process of synthesizing CBFs given a system description through a tree-search algorithm inspired by branch-and-cut methods from combinatorial optimization. Finally, we demonstrate our approach on a series of interesting examples of switched affine systems.

Key words: Control Barrier Functions (CBFs); Switched Systems; Nonsmooth Analysis; Combinatorial Optimization; Safety Enforcement.

1 Introduction

We study safety enforcement for switched affine systems through piecewise affine (min-max) control barrier functions. Designing safe controllers is a problem of critical importance for applications such as autonomous vehicles, robotics, and healthcare applications. Safety ensures that system trajectories remain within a predefined safe region, preventing undesired or hazardous outcomes. Control Barrier Functions (CBFs) [3] form a well-established approach to safety enforcement for autonomous systems. Informally, CBFs are functions over the state-space such that the set of states wherein the CBF is non-positive are control invariant through an appropriate feedback law and, further, the CBF takes on positive values over the unsafe states.

In this study, we focus on the synthesis of multiple "min-max" CBFs for switched affine systems, which are formed by taking pointwise minimum/maximum over a set of affine functions. Specifically, we consider switched

affine systems of the form $\dot{x}(t) = A_l x(t) + b_l$. The control input takes the form of a continuous-time switching signal that selects the mode based on full-state feedback. We use ideas from non-smooth analysis to formulate mathematical conditions for CBFs. We show that given piecewise min/max CBFs, we can extract a state-based feedback in the form of a mapping from states to sets of possible modes that guarantee safety of the closed loop system. We also extend our previous work on synthesis of piecewise affine control Lyapunov functions for switched affine systems to provide an algorithm for synthesizing CBFs from the description of the switched system [7,21]. It is well known that synthesizing polyhedral Lyapunov functions is already a hard problem [7]. Extending this to polyhedral CBFs introduces additional challenges, as the functions must satisfy more constraints to ensure safety. Our work combines the min- and max- barrier functions to establish a single min-max CBF.

Our proposed approach is based on the counterexample-guided inductive synthesis (CEGIS) framework [31]. This approach is a powerful and widely-used method for synthesizing Lyapunov and barrier functions [1,2,7,8,12,28]. Based on the CEGIS approach in our study, the algorithm alternates between a verifier and a learner, iteratively refining candidates until a valid

---

* Will be added in subsequent version.

Email addresses: sara.kamali@colorado.edu (Sara Kamali), gberger@uclouvain.be (Guillaume O. Berger), srirams@colorado.edu (Sriram Sankaranarayanan).

CBF is found or the tree search reaches a predefined expansion limit.

Organization: Section 1.1 reviews related work, and Section 1.2 introduces the notations used throughout the paper. Section 2 presents the preliminaries and the problem statement. In Section 3, we present polyhedral control barrier certificates and pointwise minimum control barriers. The branch-and-bound tree search algorithm is detailed in Section 5. Section 6 explains the Min-Max multiple barriers. Finally, we evaluate our approach using several examples, including a DC-DC converter, a 2D numerical example, a 3D multi-agent system, and a 6D car velocity system, in Section 7.

1.1 Related Work

Control Barrier Functions: CBFs were first conceived by Wieland and Allgower [37] as extensions of the notion of barrier functions [27] to include control inputs, just as control Lyapunov functions [32,33] extend the classic notion of a Lyapunov function. Ames et al. in [4] expanded the definition and introduced the notion of safety filters, that have made important contributions towards assured autonomous systems. Since then, the concept has been extended to address high-degree systems [38,39], uncertain systems [35], and systems with actuation constraints [3]. Notably, many of these works focus on smooth, single CBFs.

Nonsmooth Barrier/Lyapunov Functions: Nonsmooth CBFs were first proposed by Glotfelter et al. in [19], where the authors introduced the use of min and max operators to define CBFs for multi-agent systems. Subsequent works extended the concept of nonsmooth barrier functions considerably and investigated the ability to form Boolean combinations of CBFs [26,20,18]. This paper provides a stronger condition for non-smooth CBFs for switched affine systems and provides algorithms for their synthesis.

Sum Of Squares (SOS) Approaches: The use of Sum-Of-Squares approaches for CBF synthesis has been investigaged by many, including the recent work of Clark [14]. These approaches have studied the problem of synthesizing CBFs for nonlinear systems with polynomial dynamics. However, many existing approaches are restricted to control affine systems and result in bilinear optimization problems that are hard to solve. Here, we focus on an entirely different setup that is characterized by switched affine dynamics and nonsmooth CBFs. Our approach is inspired by a branch-and-bound search with a cutting plane argument to enforce termination. While our approach also suffers from the high complexity of computing CBFs, we synthesize multiple low complexity CBFs and combine them to create CBFs with larger control invariant region. This idea of combining simple barriers to create a larger control invariant set was recently explored by Wajid and Sankaranarayanan for polynomial

CBFs using SOS approaches that are very different and complementary to the ideas in this paper [36].

Neural CBFs: Neural networks have also been utilized for constructing CBFs and CLFs. For example, in [1], the authors introduced FOSSIL, a software tool for synthesizing barrier and Lyapunov functions. Their approach employed neural networks during the learner phase of the CEGIS method to find candidate functions. Poonawala et al. in [30] has utilized single-hidden-layer neural network to synthesize a CLF with ReLU NN for a system with single-layer ReLU NN model. Their approach converts the NN to a PWA function and utilizes nonsmooth analysis to synthesis Lyapunov function and feedback controller.

CEGIS Approaches: Our approach belongs to a class of methods introduced in the formal methods community that are termed "counterexample-guided inductive synthesis" [31], wherein the set of remaining candidate solutions are refined at each step by choosing a candidate and checking whether it is a valid solution. In our setup, the candidate solutions are the coefficients of the desired CBF. Ravanbakhsh et al. [29] use a SOS programming approach to synthesize CLFs. Their approach employed CEGIS to address feasibility problems, utilizing an SMT solver to construct polynomial barrier and Lyapunov functions and design controllers. A similar approach was proposed in [40], where the authors developed an algorithm to synthesize and verify polynomial CBFs by solving SoS and linear inequalities.

Our work builds on previous ideas used to synthesize CLFs using CEGIS. Berger et al. [7] introduced a CEGIS-based method to synthesize polyhedral Lyapunov functions. Their work noted the hardness of proving stability using polyhedral Lyapunov functions. In our recent work [21], we extended this idea to construct polyhedral CLFs for switched affine systems using CEGIS and mixed-integer linear programming (MILP). The key differences in this work include extensions to control barrier functions and combining min/max CBFs. Note that the idea of combining multiple functions is unique to the case of control barriers.

1.2 Notation

Let $\mathbb{R}_{\geq 0}$ denote the set of non-negative real numbers. For a vector $c \in \mathbb{R}^m$, c^t denotes the transpose of vector c. We will denote the set of natural numbers $\{1, 2, 3, \cdots\}$ as $\mathbb{N}$. For $m \in \mathbb{N}$, let $[m] = \{1, \ldots, m\}$. And $\mathrm{co}(V)$ shows the convex hull of the set V.

2 Problem Statement

Definition 1 (Switched Affine System) *A switched affine system* Π *with* $m \in \mathbb{N}$ *modes is specified by a set*
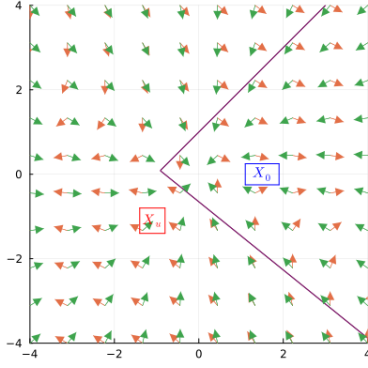
Fig. 1. Quiver plot showing the two dynamical modes from Ex. 2, the initial set X_0 and unsafe set X_u. The solid line shows the boundary of the control invariant region defined by the polyhedral CBF.

of tuples $\{(A_1, b_1), \ldots, (A_m, b_m)\}$ wherein, the dynamics for mode $l \in [m]$ are given by

$$\dot{x}(t) = A_l x(t) + b_l . \tag{1}$$

Here, $x(t) \in \mathbb{R}^n$ is the state, $l \in [m]$ denotes the mode, $A_l \in \mathbb{R}^{n \times n}$ and $b_l \in \mathbb{R}^n$ for all $l \in [m]$. The initial state $x(0)$ belongs to the initial set $X_0 \subset \mathbb{R}^n$.

Example 2 *Consider a switched affine system over $\mathbb{R}^2$ given by two matrices (A_l, b_l) for $l \in [2]$ given as follows:*

$$A_1 = \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix}, \text{ and } A_2 = \begin{bmatrix} -1 & 0.1 \\ 0.2 & -1 \end{bmatrix},$$

with $b_1 = b_2 = [0,0]^t$. The initial set X_0 is given by $[1.1, 1.9] \times [-0.25, 0.25]$. Figure 1 shows the quiver plot.

Given a switched affine system Π, a state-based switching rule is a set-valued map $\sigma : \mathbb{R}^n \rightrightarrows [m]$, wherein for any state $x \in \mathbb{R}^n$, $\sigma(x) \subseteq [m]$ is the set of all the modes that are active for a state x.

A differential inclusion over $x \in \mathbb{R}^n$ is defined as

$$\dot{x}(t) \in F(x(t)) , \tag{2}$$

wherein $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is a set-valued map. A solution to a differential inclusion (in the Carathéodory sense) is a map $x : [0, T) \to \mathbb{R}^n$ for some time $T > 0$ such that (a) x is absolutely continuous and (b) $\dot{x}(t) \in F(x(t))$ holds almost everywhere (using the Lesbegue measure) over the interval $[0, T)$.

For a given switched system Π and state-based switching rule σ, we associate a *set-valued* map $F[\Pi, \sigma]$ that is defined as follows:

$$F[\Pi, \sigma](x) = \text{co} \{A_l x + b_l \mid l \in \sigma(x)\} .$$

If $F[\Pi, \sigma]$ is a locally bounded and upper semicontinuous set-valued map with nonempty, convex, and compact values, then the existence of solutions to the differential inclusion $\dot{x}(t) \in F[\Pi, \sigma](x(t))$ is guaranteed [17]. Based on the dynamic of $F[\Pi, \sigma]$, if the switching feedback law σ is defined upper semicontinuous and nonempty, then all conditions required for the existence of solutions will be satisfied. In next section, we will define this switching feedback law σ based on the proposed polyhedral control barrier certificate.

3 Polyhedral Control Barrier Certificates

A polyhedral function $B : \mathbb{R}^n \to \mathbb{R}$ is defined by a piecewise maximum of a finite number of affine terms:

$$B(x) = \max_{i=1}^{k} c_i^t x - d_i , \tag{3}$$

wherein $k \in \mathbb{N}$ denotes the number of the pieces, $c_i \in \mathbb{R}^n$ and $d_i \in \mathbb{R}$. In general, polyhedral functions are Lipschitz continuous but not differentiable. We would like to use polyhedral barrier certificates to establish controlled invariant sets for switched affine systems.

Let Π be a switched affine system following Def. 1 with initial condition X_0, and m modes whose dynamics are defined by matrices/vectors $\{(A_1, b_1), \ldots, (A_m, b_m)\}$. Let $X_u \subseteq \mathbb{R}^n$ be a closed set of *unsafe states* that we do not wish to reach. We assume that $X_0 \cap X_u = \emptyset$.

Definition 3 (Polyhedral Control Barrier Candidate) *A polyhedral function $B : \mathbb{R}^n \to \mathbb{R}$ given by (3) is said to be a control barrier candidate for Π iff the following conditions hold:*

(C1) *The barrier function must be negative over the initial set of states: $\forall x \in X_0, \ B(x) < 0$,*
(C2) *The barrier function must be positive over the unsafe states: $\forall x \in X_u, B(x) > 0$,*
(C3) *For every state x such that $B(x) \leq 0$, there must exist a dynamical mode $l \in [m]$ such that every piece of the barrier function that is maximized at x must satisfy a "decrease condition":*

$$\forall x \in \mathbb{R}^n, \exists \, l \in [m], \ \forall i \in [k],$$
$$(B(x) \leq 0 \ \wedge \ c_i^t x - d_i = B(x)) \ \Rightarrow$$
$$c_i^t (A_l x + b_l) < -\lambda(c_i^t x - d_i)$$

where $\lambda \in \mathbb{R}$ is a constant value.

Condition **(C3)** is analogous to the "exponential barrier condition" $\dot{B}(x) < -\lambda B(x)$ which would be valid if B were a differentiable function [23]. Note also that we place no condition on points where $B(x) > 0$. We prove the existence of a switching law σ that satisfies the following properties: **(E1)** $\sigma(x)$ is an upper semicontinuous

set-valued map; **(E2)** σ induces the decrease of $B(x)$ at x whenever $B(x) < 0$; **(E3)** using σ as a feedback law results in the set $\{x \mid B(x) < 0\}$ being a positive invariant set. Therefore, we conclude that any trajectory of the closed-loop system $\dot{x}(t) \in F[\Pi, \sigma](x(t))$ starting at a state $x(0)$ wherein $B(x(0)) < 0$ will be safe: i.e, for all $t \geq 0$, $x(t) \notin X_u$.

3.1 Switching Feedback Law

Definition 4 (Upper Semicontinuity) *[5] A set-valued map $\sigma : X \rightrightarrows [m]$ is upper semicontinuous at $x \in X$ if and only if for any neighborhood $\mathscr{U}$ of $\sigma(x)$, there exists $r > 0$ such that for all $y \in B_X(x, r)$, $\sigma(y) \subset \mathscr{U}$.*

Proposition 5 *[5] If $Dom(\sigma)$ is closed, then σ is upper semicontinuous if and only if $\sigma^{-1}(l)$ is closed for every closed subset in $Dom(\sigma)$ for every $l \in [m]$.*

Based on the Definition 4 and Proposition 5, we formalize both conditions **(E1)** and **(E2)** by defing the switching law with the closure property as follow.

Definition 6 *$\sigma : \mathbb{R}^n \rightrightarrows [m]$ has the closure property if*

(G1) *$\sigma(x)$ is nonempty for all $x \in \mathbb{R}^n$,*
(G2) *$\sigma^{-1}(l) = \{x \mid l \in \sigma(x)\}$ is closed for all $l \in [m]$,*
(G3) *for all $x \in \mathbb{R}^n$, if $l \in \sigma(x)$ then*

$$(\forall i \in [k]), \quad B(x) = c_i^t x - d_i \leq 0 \Rightarrow \tag{4}$$
$$c_i^t(A_l x + b_l) < -\lambda(c_i^t x - d_i).$$

We will define the switching law that has the closure property by idea of defining a *merit function* as follows.

Definition 7 (Associated Switching Rule) *Given a polyhedral function $B : \mathbb{R}^n \to \mathbb{R}$ that satisfies the conditions **(C1)-(C3)** in Def. 3, we define the associated switching rule $\sigma : \mathbb{R}^n \rightrightarrows [m]$. Let us consider an arbitrary point $x \in \mathbb{R}^n$.*

Case-1: *If $B(x) \leq 0$, we define $\sigma(x)$ as*

$$\sigma(x) = \operatorname{argmax}_{l \in [m]} \mathscr{M}(x, l), \quad wherein \tag{5}$$

$$\mathscr{M}(x, l) = \min_{i \in [k]} \mathscr{M}_i(x, l), \quad with \tag{6}$$

$$\mathscr{M}_i(x, l) = \max_{\tau \geq 0}\{\tau \mid \tau \hat{\varphi}_i + \varphi_i + \lambda \tau \varphi_i - \varphi \leq -\tau^2\} \tag{7}$$

where $\hat{\varphi}_i = c_i^t(A_l x + b_l)$, $\varphi_i = c_i^t x - d_i$, and $\varphi = \max_{i \in [k]} c_i^t x - d_i$. The function $\mathscr{M}(x, l)$ will be denoted the merit function *at state x for choice of dynamics l.*

Case-2: *For each x with $B(x) > 0$, we define*

$$\sigma(x) = \{\sigma(x') \mid x' = \operatorname{argmin}_{z \in \mathbb{R}^n}\{\|x - z\| \mid B(z) \leq 0\}\}. \tag{8}$$

In other words, we will project the point x onto the set $\{x \mid B(x) \leq 0\}$ and reuse the definition from case-1.

Proposition 8 *The proposed merit function satisfies the following properties:*

(1) $\mathscr{M}(x, l) \geq 0$ for all x and l;
(2) $\mathscr{M}(x, l) > 0$ if only if $\forall i \in [k]$, $(B(x) = c_i^t x - d_i \leq 0)$ $\Rightarrow c_i^t(A_l x + b_l) < -\lambda(c_i^t x - d_i)$;
(3) $\mathscr{M}(x, l)$ is continuous w.r.t x.

PROOF. (1) If $\tau = 0$, then the inequality in (7) holds, since $\varphi_i \leq \varphi$ by definition. Therefore, $\mathscr{M}_i(x, l) \geq 0$ for all x, l. Hence $\mathscr{M}(x, l) = \min_{i \in [k]} \mathscr{M}_i(x, l) \geq 0$.

(2) ($\Longleftarrow$) Assume the decrease condition holds, i.e., $\forall i \in [k]$, $\varphi = \varphi_i \leq 0 \implies \hat{\varphi}_i < -\lambda \varphi_i$. Since $\varphi = \max_{i \in [k]} \varphi_i$ we have either: case (i) $\varphi = \varphi_i$ and case (ii) $\varphi_i < \varphi$.

Case (i): Considering $\varphi = \varphi_i$, the inequality in (7) can be rewritten as $\tau \hat{\varphi}_i + \lambda \tau \varphi_i \leq -\tau^2$. Considering this and the decrease assumption, $\hat{\varphi}_i < -\lambda \varphi_i$, the inequality in (7) holds for $\tau \in [0, -(\hat{\varphi}_i + \lambda \varphi_i)]$. Based on the decrease assumption, $-(\hat{\varphi}_i + \lambda \varphi_i) > 0$ which results in for all $i \in [k]$, $\mathscr{M}_i(x, l) = \max\{0, -(\hat{\varphi}_i + \lambda \varphi_i)\} = -(\hat{\varphi}_i + \lambda \varphi_i)$ which is a positive value.

Case (ii): Let's define $\vartheta(\tau) = \tau \hat{\varphi}_i + \varphi_i + \lambda \tau \varphi_i - \varphi + \tau^2$. The function $\vartheta(\tau)$ is an upward quadratic function with roots $\bar{\tau}_{1,2} = \frac{-(\hat{\varphi}_i + \lambda \varphi_i) \pm \sqrt{(\hat{\varphi}_i + \lambda \varphi_i)^2 - 4(\varphi_i - \varphi)}}{2}$. Since $\varphi_i < \varphi$, then $\bar{\tau}_{i_{1,2}}$ are real values. Moreover, $\vartheta(0) = \varphi_i - \varphi < 0$. Considering all, it can be easily concluded $\vartheta(\tau) \leq 0$ for all $\tau \in [0, \bar{\tau}_{i_1}]$ where $\bar{\tau}_{i_1} = \frac{-(\hat{\varphi}_i + \lambda \varphi_i) + \sqrt{(\hat{\varphi}_i + \lambda \varphi_i)^2 - 4(\varphi_i - \varphi)}}{2}$ which results in the inequality in (7) holds for $\tau \in [0, \bar{\tau}_{i_1}]$. Therefore, $\mathscr{M}_i(x, l) = \bar{\tau}_{i_1} > 0$ and hence from two cases $\mathscr{M}(x, l)$ is greater than 0.

($\Longrightarrow$) We prove this by contradiction. Assume that the decrease condition does not hold, i.e., there exists $i \in [k]$ such that $\varphi = \varphi_i \leq 0 \implies \hat{\varphi}_i \geq -\lambda \varphi_i$. So, we can write $\exists i \in [k]$ such that $\vartheta(\tau) = \tau(\hat{\varphi}_i + \lambda \varphi_i) + \tau^2$ which is less than or equal zero just at $\tau = 0$. So, there exists $i \in [k]$ such that $\mathscr{M}_i(x, l) = 0$ and hence $\mathscr{M}(x, l) = 0$.

(3) If $\varphi_i = \varphi$, then $\mathscr{M}_i(x, l) = -(\hat{\varphi}_i + \lambda \varphi_i) = -c_i^t(A_l x + b_l) - \lambda(c_i^t x - d_i)$ which is continuous map w.r.t. x. And if $\varphi_i < \varphi$, then $\mathscr{M}_i(x, l) = \bar{\tau}_{i_1}$ which is a continuous function w.r.t. x. Thus, the merit function $\mathscr{M}(x, l) = \min_{i \in [k]} \mathscr{M}_i(x, l)$ is continuous.

Theorem 9 *The proposed switching law σ in Definition 7 has the closure property.*

PROOF. **(G1)** and **(G3)** are direct consequence of condition **(C3)** from Def. 3, the definition of the

associated switching rule in Def. 7, and its properties in Prop. 8. To prove **(G2)**, consider $B(x)$ is a CBF. Let $l \in [m]$. For any $x \in \mathbb{R}^n$ such that $B(x) \leq 0$, it holds that $l \in \sigma(x)$ if and only if $\mathscr{M}(x, l) - \max_{l' \in [m]} \mathscr{M}(x, l') \geq 0$. Hence, $\{x \mid B(x) \leq 0, l \in \sigma(x)\}$ is the pre-image of $[0, \infty)$ by the continuous function $\mathscr{M}(x, l) - \max_{l' \in [m]} \mathscr{M}(x, l')$. Hence, it is closed since the pre-image of a closed set by a continuous function is closed. On the other hand, from (8), for all $x \in \mathbb{R}^n$ such that $B(x) \geq 0$, $l \in \sigma(x)$ if and only if $\min_{\{z \mid B(z) \leq 0, l \in \sigma(z)\}} \|x - z\| = \min_{\{z \mid B(z) \leq 0\}} \|x - z\|$. Hence, for the same reason as above [distance to a closed set is continuous], $\{x \mid B(x) \geq 0, l \in \sigma(x)\}$ is closed.

Theorem 10 *If σ be upper semicontinuous, then $F[\Pi, \sigma]$ is upper semicontinuous.*

PROOF. Let $x \in \mathbb{R}^n$. Let $(x_n)_{n \in \mathbb{N}}$ be a sequence converging to x. Assume σ is upper semicontinuous. So,

$$\exists N \in \mathbb{N} \; s.t. \; \forall n \geq N, \; \sigma(x_n) \subseteq \sigma(x)$$

which results in

$\forall \varepsilon > 0 \; \exists N \in \mathbb{N} \; s.t. \; \forall n \geq N,$
$\{A_l x_n + b_l \mid l \in \sigma(x_n)\} \subseteq \{A_l x + b_l \mid l \in \sigma(x)\} + \varepsilon \mathscr{B}$

where $\mathscr{B}$ is a unit ball. Applying the convex hull,

$\forall \varepsilon > 0 \; \exists N \in \mathbb{N} \; s.t. \; \forall n \geq N,$
$\mathsf{co}\{A_l x_n + b_l \mid l \in \sigma(x_n)\} \subseteq \mathsf{co}\{A_l x + b_l \mid l \in \sigma(x)\} + \varepsilon \mathscr{B}$

which is equivalent to

$$\forall \varepsilon > 0, \exists N \in \mathbb{N} \; s.t. \; \forall n \geq N,$$
$$F[\Pi, \sigma](x_n) \subseteq F[\Pi, \sigma](x) + \varepsilon \mathscr{B}$$

that means $F[\Pi, \sigma]$ is upper semicontinuous.

The following result is a direct consequence of Definition 6, and Theorems 9 and 10.

Theorem 11 *If the switching law σ has the closure property, all trajectories starting from X_0 are complete.*

4 Soundness of Polyhedral Barrier Certificate

We will now prove the soundness of polyhedral control barrier functions for a switched affine system Π with polyhedral barrier function B and associated switching rule σ. First, we recall some facts from nonsmooth analysis that will be helpful for establishing the results. Then, we characterize the weak Lie derivative of B w.r.t the differential inclusion $F[\Pi, \sigma](x(t))$ and show the soundness. For convenience, we will write $F = F[\Pi, \sigma]$ throughout this section.

4.1 Nonsmooth Analysis

We will now recall some facts from nonsmooth analysis that will be helpful in establishing the key results in this paper. We refer the reader to the survey by Cortes [17] and the textbooks of Clarke [15] and Lakshmikantham et al. [24] for detailed definitions and proofs. Let $\dot{x}(t) \in F(x(t))$ be a differential inclusion, wherein F is locally bounded and upper semicontinuous. We will assume that all differential inclusions in this section are locally bounded and upper semicontinuous, thus guaranteeing the existence of solutions. Let $g(x) = \max_{i=1}^k g_i(x)$ be a piecewise maximum of finitely many smooth (infinitely differentiable) functions $g_1, \ldots, g_k$. We recall Prop. 2.3.12 in [16]:

Theorem 12 *For all $x \in \mathbb{R}^n$, it holds that $\partial g(x) \subseteq \mathsf{co}\left(\{\nabla g_j(x) \mid j \in [k], g_j(x) = g(x)\}\right)$.*

Definition 13 (Weak Set-Valued Lie Derivative)
The weak Lie derivative of a function g w.r.t. the differential inclusion F is given by

$$\mathcal{L}_F^w g(x) = \{\theta \cdot \xi \mid \theta \in F(x), \xi \in \partial g(x)\}, \qquad (9)$$

wherein $\partial g(x)$ is the Clarke generalized gradient of g at x, and $a \cdot b$ denotes the inner product of vectors a, b.

The following lemma is useful for computing ∂g wherein $g = \max_{i=1}^k g_i$ for Lipschitz continuous functions g_i.

Lemma 14 *Let $g(x) = \max_{i=1}^k g_i(x)$, wherein $g_i(x)$ are smooth functions and $F(x) = \mathsf{co}\{f_1(x), \ldots, f_m(x)\}$.*

$$\mathcal{L}_F^w g(x) \subseteq \mathsf{co}\{f_i(x) \cdot \nabla g_j(x) \mid i \in [m], j \in [k], g_j(x) = g(x)\}.$$

PROOF. From Eq. (9), we have that each element a of $\mathcal{L}_F^w$ is an inner product of the form $\theta \cdot \xi$ wherein $\theta \in F(x)$ and $\xi \in \partial g(x)$. By assumption, we may write $\theta = \sum_{i=1}^m \lambda_i f_i$, wherein $\lambda_i \geq 0$ for $i \in [m]$ and $\sum_{i \in [m]} \lambda_i = 1$. From Theorem 12, we note that $\partial g(x) \subseteq \mathsf{co}\left(\{\nabla g_j(x) \mid j \in [k], g_j(x) = g(x)\}\right)$. Therefore, consider the set $G(x) = \{\nabla g_j(x) \mid j \in [k], g_j(x) = g(x)\}$. We have $\mathcal{L}_F^w g(x) \subseteq L(x)$ wherein $L(x) = \{\theta \cdot \xi \mid \theta \in F(x), \xi \in G(x)\}$. Each $a \in L(x)$ can be written as $a = \left(\sum_{i=1}^m \lambda_i f_i\right) \cdot \left(\sum_{j=1}^k \gamma_j g_j\right) = \sum_{i=1}^m \sum_{j=1}^k \lambda_i \gamma_j (f_i \cdot g_j)$, wherein $\gamma_j \geq 0$, and $\sum_{j \in [k]} \gamma_j = 1$. Thus, $\lambda_i \gamma_j \geq 0$ and $\sum_{i \in [m]} \sum{j \in [k]} \lambda_i \gamma_j = 1$, yielding $a \in \mathsf{co}\{f_i(x) \cdot g_j(x) \mid i \in [m], j \in [k]\}$.

Theorem 15 *Let $x : [0, T] \mapsto \mathbb{R}^n$ be an absolutely continuous function that is a solution to the differential inclusion $\dot{x}(t) \in F(x(t))$, and g be a locally Lipschitz continuous function. The inclusion holds almost everywhere in the interval $[0, T]$: $\frac{d}{dt} g(x(t)) \in \mathcal{L}_F^w g(x(t))$.*

PROOF. Proof follows from Lemma 1 of [6], Proposition 2.2.2 of Clarke's book [15], and Remark 2 of [19].

For a scalar $\lambda \in \mathbb{R}$, we define the function $\alpha : \mathbb{R} \to \mathbb{R}$:

$$\alpha(s) = \begin{cases} -\lambda s & s \leq 0 \\ \infty & \text{otherwise} \end{cases} \qquad (10)$$

Since, α is a Lipschitz continuous function, the ODE $\dot{s} = \alpha(s)$ with initial condition $s(0) = s_0 \in \mathbb{R}_{\leq 0}$ has a unique solution $s(t) = e^{-\lambda t} s_0$ when $s_0 \leq 0$. Therefore,

$$s(t) \leq \begin{cases} e^{-\lambda t} s_0 & \text{if } s_0 \leq 0 \\ \infty & \text{otherwise.} \end{cases} \qquad (11)$$

The following useful "comparison principle" follows from Theorem 1.10.2 of Lakshmikantham and Leela [24].

Theorem 16 *Let $x : [0, T) \to \mathbb{R}^n$ be an absolutely continuous function that is a solution of the differential inclusion $\dot{x}(t) \in F(x(t))$ and $g : \mathbb{R}^n \to \mathbb{R}$ be a Lipschitz continuous function such that for almost every $t \in [0, T)$,*

$$\frac{d}{dt} g(x(t)) \; < \; \alpha(g(x(t))), \text{ and } g(x(0)) \leq 0$$

Thus, $g(x(t)) \; \leq \; e^{-\lambda t} g(x(0)) \leq 0$ for all $t \in [0, T)$.

4.2 Soundness of Polyhedral Barrier Certificate

Now we prove that any trajectory $x : [0, T) \to \mathbb{R}^n$ that is a solution to the differential inclusion $\dot{x}(t) \in F(x(t))$ and initial condition $x(0)$ satisfying $B(x(0)) < 0$ where $B(x(t))$ is a polyhedral barrier certificate, will satisfies $B(x(t)) < 0$ for all time $t \in [0, T)$.

Lemma 17 *For all $x \in \mathbb{R}^n$ the following inclusion holds:*

$$\mathcal{L}_F^w B(x) \subseteq \mathsf{co}\left\{ c_i^t(A_l x + b_l) \;\middle|\; \begin{array}{c} i \in [k], c_i^t x - d_i = B(x) \\ l \in \sigma(x) \end{array} \right\}.$$

PROOF. First, $F[\Pi, \sigma](x) = \mathsf{co}\{A_l x + b_l \mid l \in \sigma(x)\}$. Next, $B = \max_{i=1}^{k} B_i$, wherein each $B_i(x) = c_i^t x - d_i$ is a smooth function of x. Applying Lemma 14, we obtain

$$\mathcal{L}_F^w B(x) \subseteq \mathsf{co}\left\{ f_j \cdot \nabla B_i \;\middle|\; \begin{array}{c} i \in [k], j \in [m], \\ f_j \in F(x), \; B_i(x) = B(x) \end{array} \right\}.$$

We now connect $\mathcal{L}_F^w B(x)$ with $\alpha(B(x))$, wherein α is the function defined in (10).

Lemma 18 *For all $x \in \mathbb{R}^n$, $\sup(\mathcal{L}_F^w B(x)) \leq \alpha(B(x))$.*

PROOF. Case-1: Assume $B(x) \leq 0$. From Lemma 17, whenever $B(x) \leq 0$

$$\mathcal{L}_F^w B(x) \subseteq \underbrace{\mathsf{co}\left\{ c_i^t(A_l x + b_l) \;\middle|\; \begin{array}{c} i \in [k], c_i^t x - d_i = B(x) \\ l \in \sigma(x) \end{array} \right\}}_{S(x)}.$$

However, from the definitions of B and σ, we note that for all $l \in \sigma(x)$, and for all $i \in [k]$ such that $c_i^t x - d_i = B(x)$, we have $c_i^t(A_l x + b_l) < \lambda(c_i^t x - d_i)$. Therefore, for all $a \in S(x)$, we conclude that $a < -\lambda B(x) = \alpha(B(x))$. Hence, $\sup(\mathcal{L}_F^w B(x)) \leq \sup(S(x)) < \alpha(B(x))$.

Case-2: Assume $B(x) > 0$. Again, from Lemma 17 $\mathcal{L}_F^w B(x) \subseteq S(x)$. Hence, each value of $\mathcal{L}_F^w B(x)$ is less or equal to infinity. So, when $B(x) > 0$, we have $\sup(\mathcal{L}_F^w B(x)) \leq \sup(S(x)) \leq \infty$. Thus, by comparing with (10), $\sup(\mathcal{L}_F^w B(x)) \leq \alpha(B(x))$ for $B(x) > 0$.

We can now establish the soundness result that follows directly from Theorem 15 and Lemma 18 with the comparison principle established in Theorem 16.

Theorem 19 *Let $B : \mathbb{R}^n \to \mathbb{R}$ be a polyhedral control barrier with the associated map σ. Let $x : [0, T) \to \mathbb{R}^n$ be any solution to the differential inclusion $\dot{x}(t) \in F[\Pi, \sigma](x(t))$ such that $B(x(0)) \leq 0$. For all $t \in [0, T)$, $B(x(t)) \leq e^{-\lambda t} B(x(0)) \leq 0$.*

4.3 Min Control Barrier Certificate

In this section, we consider CBFs that are piecewise minima of a finite number of the affine terms as

$$B(x) = \min_{i=1}^{k} c_i^t x - d_i \qquad (12)$$

Lemma 20 *Let $g(x) = \min_{i=1}^{k} g_i(x)$ for smooth g_i.*

$$\partial g(x) \subseteq \mathsf{co}\left(\{\nabla g_j(x) \mid j \in [k], g_j(x) = g(x)\}\right).$$

PROOF. The proof follows from Theorem 12 and $\min_{i=1}^{k} g_i(x) = -\max_{i=1}^{k}(-g_i(x))$.

Moreover, the piecewise minimum functions holds the locally Lipschitz property. Considering this and Lemma 20, all the lemmas and theorems in section 4.2 can be applied to piecewise minimum functions.

Definition 21 (Piecewise Min Control Barriers)
A piecewise minimum function given by (12) is said to be a control barrier certificate for Π iff the conditions (C1), (C2) (for polyhedral CBFs) and (C'3) hold for a constant $\lambda \geq 0$:

(C'3) For every state x such that $B(x) \leq 0$, there must exist a dynamical mode $l \in [m]$ such that every piece of the barrier function that is <u>minimized</u> at x must satisfy a "decrease condition":

$$\forall x \in \mathbb{R}^n, \exists \, l \in [m], \, \forall i \in [k],$$
$$(B(x) \leq 0 \, \wedge \, \textcolor{red}{c_i^t x - d_i = B(x)}) \Rightarrow$$
$$c_i^t(A_l x + b_l) < -\lambda(c_i^t x - d_i).$$

The key difference in C'3 lies in the condition highlighted in red: $c_i^t x - d_i = B(x) = \min_{l=1}^{k} c_l^t x - d_l$. This condition is expanded as the conjunction of inequalities: $\bigwedge_{l=1}^{k} c_i^t x - d_i \leq c_l^t x - d_l$.

To satisfy the closure property, we define the merit function related to the associated switching rule for the piecewise minimum as

$$\mathcal{M}(x, l) = \min_{i \in [k]} \mathcal{M}_i^{min}(x, l), \quad \text{wherein,}$$

$$\mathcal{M}_i^{min}(x, l) = \max_{\tau \geq 0}\{\tau \, | \, \tau \hat{\varphi}_i - \varphi_i + \lambda \tau \varphi_i + \varphi \leq -\tau^2\}. \quad (13)$$

In a similar manner to what was demonstrated in Proposition 8 and Theorem 9, the associated switching rule to this merit function also preserves the closure property.

Remark 22 *The piecewise minimum over convex functions is convex, and consequently, the regularity property which results in the inclusion in Theorem 15 holds for strong set-valued Lie derivative [19] as*

$$\frac{d}{dt} g(x(t)) \in \mathcal{L}_F^s(g(x(t))) \quad a.e., wherein,$$

$$\mathcal{L}_F^s(g(x)) = \{r \in \mathbb{R} | \exists \theta \in F(x), \forall \xi \in \partial g(x) : \theta \cdot \xi = r\}.$$

The strong derivative can provide less conservative results for finding nonsmooth barrier certificate.

5 Branch and Bound Tree Search Algorithm

In this section, we will provide an algorithmic approach to search for max (polyhedral) barrier certificates as functions of the form $B(x) = \max_{i=1}^{k} c_i^t x - d_i$. Our algorithm is adapted from our previous works on synthesis of polyhedral Lyapunov and control Lyapunov functions [7,21]. Assume that the coefficients c_i, d_i are

Data: See Def. 23
Result: CBF B; or FAIL

1 Initialize: root α_0 with $C(\alpha_0) : \psi_0$ (see section 5.1), $W(\alpha_0) = \emptyset$, $p \leftarrow 1$
    while $\exists$ *Unexplored Leaf* **do**
2 $\beta \leftarrow$ Choose unexplored leaf and mark as explored
3   **if** $(C(\beta)$infeas. $\vee$ CHEBYSHEVRADIUS$(C(\beta)) \leq R_{\min})$
         then continue
4 $(c_i^{(p)}, d_i^{(p)})_{i \in [k]} \leftarrow$ CHOOSEMVECENTER$(C(\beta))$
5   result $\leftarrow$ VERIFYCANDIDATE$((c_i^{(p)}, d_i^{(p)})_{i \in [k]})$
6   **if** result = VERIFIED **then**
7   Return CBF $B(x) := \max_{i \in [k]}(c_i^{(p)} x - d_i^{(p)})$
8   **else**
9   Create children $\beta_{j,i}$, for $i \in [k], j \in [m]$
10 $W(\beta_{j,i}) \leftarrow W(\beta) \cup \{(x, j, i)\}$
11 $C(\beta_{j,i}) \leftarrow C(\beta) \wedge CW(\{(x, j, i)\})$ Cf. (15)
12 $p \leftarrow p + 1$
13 **return** FAIL

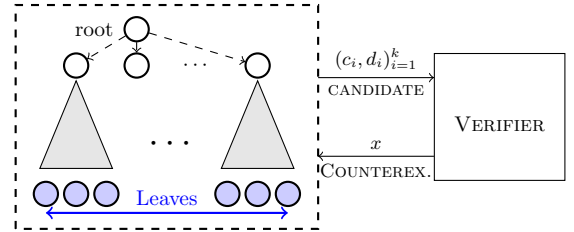**Algorithm 1.** Tree search algorithm pseudocode.

Fig. 2. Branch-and-Bound Tree Search Overview.

bounded by user-input parameter γ. The following constraint expresses this fact:

$$\psi_0^{(a)} := \bigwedge_{i=1}^{k}(-\gamma \, 1_n \leq c_i \leq \gamma \, 1_n \, \wedge \, -\gamma \leq d_i \leq \gamma) \quad (14)$$

1_n refers to the $n \times 1$ vector all of whose entries are 1.

Definition 23 (CBF Synthesis Problem) *The polyhedral CBF synthesis problem is as follows:*

- INPUTS: *Switched affine system Π with m modes (A_i, b_i), $i = 1, \ldots, m$, polyhedral initial set $X_0 \neq \emptyset$, polyhedral unsafe set $X_u \neq \emptyset$, # pieces $k > 0$, constant $\lambda > 0$, parameters $\epsilon > 0, \gamma > 0, R_{\min} > 0$.*
- OUTPUT: *Polyhedral CBF $B(x) = \max_{i=1}^{k} c_i^t x - d_i$ satisfying the conditions in Def. 3 or FAIL, denoting that no CBF was discovered.*

Overview: The branch-and-bound tree search algorithm maintains a tree where each node α of the tree has the following information: (a) a polyhedral constraint $C(\alpha)$ over the unknown coefficients $(c_i, d_i)_{i=1}^{k}$ and (b) a set of witnesses $W(\alpha) : \{(x, j, i) \mid x \in \mathbb{R}^n, j \in [m], i \in [k]\}$. We will discuss the meaning of each witness and the connection between $W(\alpha)$ and $C(\alpha)$ subsequently. The pseudocode of the algorithm is provided as Algorithm 1. Its operation of the algorithm is illustrated in

Figure 2. At each step, the algorithm **proposes a candidate CBF** (line 4) from the tree which is **checked by a verifier** (line 5) that checks conditions **(C1)-(C3)**. If the check succeeds, then the approach yields a valid CBF (line 7). Otherwise, it yields a counterexample that is used to **refine the tree** (lines 9- 11).

Tree node: The polyhedral constraint $C(\alpha)$ associated with node α describes the set of unexplored CBF candidates $(c_i, d_i)_{i \in [k]}$, while the witnesses $W(\alpha)$ is a set that contains elements of the form (x, j, i) wherein $x \in \mathbb{R}^n$ is a state, $j \in [m]$ is a dynamic that has been assigned to the state x by our tree search and $i \in [k]$ asserts that the ith piece must be maximal at point x.

Definition 24 (Constraints for Set of Witnesses)
The set of witnesses for each node $W(\alpha)$ poses the following polyhedral constraint $CW(\alpha)$:

$$\bigwedge_{(x,j,i) \in W(\alpha)} \begin{pmatrix} \bigwedge_{i' \in [k]} c_i^t x \geq c_{i'}^t x \ \wedge \\ c_i^t x - d_i \leq 0 \ \wedge \\ c_i^t (A_j x + b_j) \leq -\lambda(c_i^t x - d_i) - \epsilon \end{pmatrix} \quad (15)$$

The first line expresses that the ith piece is maximal at x, the second line enforces that the CBF is non-positive at x and the third enforces a decrease condition through the dynamics $j \in [m]$ at point x. Note that we have used a fixed user-defined tolerance ϵ to avoid working with a strict inequality. Finally, we define the constraint $C(\alpha)$ associated with node α as:

$$C(\alpha) \ = \ CW(\alpha) \ \wedge \ \psi_0, \quad (16)$$

where ψ_0 will be derived in the subsequent section.

5.1 Initializing the Tree

The initial tree just has a root node α_0 with $W(\alpha_0) = \emptyset$ and $C(\alpha_0) = \psi_0 = \psi_0^a \wedge \psi_0^b \wedge \psi_0^c$, wherein the constraint ψ_0^a (Cf. (14)) enforces the bound on the coefficients, and constraints ψ_0^b and ψ_0^c that enforce conditions **(C1)** and **(C2)** from Def. 3, respectively.

Recall that condition **(C1)** requires $B(x) < 0$ for all $x \in X_0$. We strengthen it to the condition $B(x) \leq -\epsilon$ for all $x \in X_0$, wherein $\epsilon > 0$ is a user-provided tolerance. Equivalently, we may write this as: $\forall x \in X_0, \bigwedge_{i \in [k]} c_i x - d_i \leq -\epsilon$. Assume that X_0 is a polyhedron specified as $A_0 x \leq b_0$. We may dualize the constraints by adding extra dual variables and using Farkas' Lemma [13] to obtain:

$$\psi_0^b : \bigwedge_{i \in [k]} A_0^t \lambda_i = c_i \ \wedge \ b_0^t \lambda_i \leq d_i - \epsilon \ \wedge \ \lambda_i \geq 0.$$

The multipliers $\lambda_1, \ldots, \lambda_k$ can remain in the constraint for the purposes of Alg. 1 or eliminated using a suitable projection algorithm for polyhedra such as Fourier-Motzkin elimination [13]. Alternatively, given the vertices $x_0^{(1)}, \ldots, x_0^{(K)}$ of the polyhedron X_0, then we may write

$$\psi_0^b : \bigwedge_{i=1}^{k} \bigwedge_{r=1}^{K} c_i x_0^{(r)} - d_i \leq -\epsilon.$$

Lemma 25 *Assuming compactness of X_0, any candidate solution $(c_i, d_i)_{i=1}^k$ that satisfies constraint ψ_0^b also satisfies **(C1)** from Def. 3.*

Condition **(C2)** requires that $B(x) > 0$ for all $x \in X_u$. Alternatively, we assert that the set $C_0 := \{x \mid B(x) \leq 0\}$ does not intersect the set X_u. Assuming that X_u is a non-empty and compact polyhedral set, there must exist a separating hyperplane $c^t x - d$ such that $C_0 \subseteq \{x \mid c^t x - d \leq 0\}$ and $X_i \subseteq \{x \mid c^t x - d \geq \epsilon\}$. A simple trick that saves computational time is to assert that the "first piece" $c_1^t x - d_1$ forms this separating hyper plane. This is a sufficient condition that guarantees **(C2)**. This can be achieved through Farkas lemma at the cost of having extra multiplier variables or assuming that the vertex set $x_u^{(1)}, \ldots, x_u^{(L)}$ of the polyhedron X_u is known.

$$\psi_0^c : \bigwedge_{r=1}^{L} c_1 x_0^{(r)} - d_1 \geq \epsilon.$$

Lemma 26 *Assuming compactness of X_u any candidate solution $(c_i, d_i)_{i=1}^k$ that satisfies constraint ψ_0^c also satisfies **(C2)** from Def. 3.*

The initial tree node α_0 has the constraint $C(\alpha_0) := \psi_0^a \wedge \psi_0^b \wedge \psi_0^c$

The algorithm proceeds iteratively, with each iteration number p consisting of three steps: (a) selecting a candidate node (line 4); (b) verifying the candidate (line 5); and (c) refining the tree if verification fails (lines 9 - 11).

5.2 Candidate Selection

At the beginning of the pth iteration, we select a candidate CBF $B^{(p)} = \max_{i \in [k]} c_i^{(p)} x - d_i^{(p)}$. This is performed by selecting a previously unexplored leaf β of the tree. The leaf is then marked as explored. We then choose a point $(c_i^{(p)}, d_i^{(p)})_{i \in [k]}$ that satisfies $C(\beta)$. If $C(\beta)$ is infeasible, the algorithm simply moves on to iteration $p + 1$.

Due to the way we have defined ψ_0 in Section 5.1, we note that any candidate satisfies ψ_0.

Lemma 27 *Any candidate $(c_i^{(p)}, d_i^{(p)})_{i=1}^k$ chosen at the pth iteration of Algorithm 1 satisfies ψ_0 and therefore, satisfies constraint **(C1)** and **(C2)** of Def. 3.*

PROOF. The candidate must satisfy $C(\alpha)$ for the chosen node α. Note that $C(\alpha)$ has ψ_0 as a conjunct by definition (see (16)). The rest follows from applying Lemmas 25 and 26.

5.3 Verifier

The verifier receives a function $B^{(p)}$ and either needs to certify that $B^{(p)}$ is a CBF or find a counter-example $x \in \mathbb{R}^n$ showing that $B^{(p)}$ is not a CBF. To do so, we check conditions **(C1)**-**(C3)** (Def. 3). However, conditions **(C1)** and **(C2)** are already safisfied through Lemma 27. It remains to verify condition **(C3)**. Recall that **(C3)** requires for every state x such that $B(x) \leq 0$, there must exist a dynamical mode $l \in [m]$ such that every piece of the barrier function that is maximized at x must satisfy a "decrease condition" We search for a state x that violates **(C3)**:

$$
\left.
\begin{array}{l}
\text{find } x \in \mathbb{R}^n \text{ s.t.} \\[4pt]
\bigwedge_{l=1}^{m} \bigvee_{i=1}^{k}
\begin{pmatrix}
(c_i^{(p)})^t x - d_i^{(p)} \leq 0 \ \wedge \\
\bigwedge_{i' \in [k]} (c_i^{(p)})^t x - d_i^{(p)} \geq (c_{i'}^{(p)})^t x - d_{i'}^{(p)} \\
(c_i^{(p)})^t (A_l x + b_l) \geq -\lambda((c_i^{(p)})^t x - d_i^{(p)})
\end{pmatrix}
\end{array}
\right\}
\tag{17}
$$

Note that the constraint can be encoded as a mixed-integer optimization problem due to the presence of the disjunction. If the MILP is infeasible, the condition **(C3)** is verified and $B^{(p)}$ is the barrier function we seek. Otherwise, we obtain a witness x_p.

5.3.1 Mixed Integer Linear Program Encoding

We will now encode the constraint for finding a potential counterexample for condition **(C3)** as a mixed-integer optimization problem. This constraint corresponds to Eq. (17) and is recalled below:

$$
\left.
\begin{array}{l}
\text{find } x \in \mathbb{R}^n \text{ s.t.} \\[4pt]
\bigwedge_{l=1}^{m} \bigvee_{i=1}^{k}
\begin{pmatrix}
(c_i^{(p)})^t x - d_i^{(p)} \leq 0 \ \wedge \\
\bigwedge_{i' \in [k]} (c_i^{(p)})^t x - d_i^{(p)} \geq (c_{i'}^{(p)})^t x - d_{i'}^{(p)} \\
(c_i^{(p)})^t (A_l x + b_l) \geq -\lambda((c_i^{(p)})^t x - d_i^{(p)})
\end{pmatrix}
\end{array}
\right\}
\tag{18}
$$

To encode this as a MILP, we first introduce binary variables of the form $w_{l,i} \in \{0,1\}$ for $l = 1, \ldots, m$ and $i = 1, \ldots, k$. If $w_{l,i} = 1$ for some specific value of l, i, we will ensure that the inequalities shown above will hold for that l, i. In other words, we wish to satisfy:

$$
\begin{pmatrix}
w_{l,i} = 1 \ \Rightarrow \ (c_i^{(p)})^t x - d_i^{(p)} \leq 0 \ \wedge \\
\bigwedge_{i' \in [k]} w_{l,i} = 1 \ \Rightarrow \ (c_i^{(p)})^t x - d_i^{(p)} \geq (c_{i'}^{(p)})^t x - d_{i'}^{(p)} \\
w_{l,i} = 1 \ \Rightarrow \ (c_i^{(p)})^t (A_l x + b_l) \geq -\lambda((c_i^{(p)})^t x - d_i^{(p)})
\end{pmatrix}
$$

Additionally, we enforce the conjunction of disjunctions by the constraint:

$$
w_{l,1} + \cdots + w_{l,k} \geq 1, \ \textbf{for } l = 1, \ldots, m
$$

The implication is encoded using a M constraint. Consider a constraint of the form

$$
w_{l,i} = 1 \ \Rightarrow \ c^t x - d \geq 0 \ , .
$$

We encode this by introducing a constraint of the form:

$$
c^t x - d \geq -(M \|c\|_1 + \|d\|_1)(1 - w_{l,i}) \, .
$$

Here M is chosen so that the following property holds: If there exists a feasible solution x to (18) then there exists a solution with $\|x\|_\infty \leq M$. Note that if $w_{l,i}$ equals 1, we obtain the constraint $c^t x - d \geq 0$. Otherwise, we obtain the constraint $c^t x - d \geq -(M \|c\|_1 + \|d\|_1)$, which is simply a consequence of $\|x\|_\infty \leq M$.

Since, the candidate $(c_i^{(p)}, d_i^{(p)})_{i=1}^{k}$ is fixed. Assuming that $c_i^{(p)}, d_i^{(p)}$, the entries of A_j and λ are all rational numbers. Let $L > 0$ be the least common multiple (LCM) of all the denominators of $c_i^{(p)}, d_i, A_j, \lambda$ for $i \in [k]$ and $j \in [m]$. A bound M over the magnitude of $\|x_c\|_\infty$ is obtained as follows:

Lemma 28 Let $x_c \in \mathbb{R}^n$ be a solution to (18). Let $a = \max_{i=1}^{k} \max(\|L c_i^{(p)}\|_1, L d_i^{(p)})$ and $U = L^2 \times \max(2, \lambda + \|A_1\|_\infty, \ldots, \lambda + \|A_m\|_\infty) a$. It follows that $\|x_c\|_\infty \leq (nU)^n$.

PROOF. Note that by (18), there exists a map $\mu : [m] \to [k]$ and a number $\epsilon > 0$ such that x_c belongs to a polyhedron defined by inequalities of the form:

$$
\bigwedge_{l=1}^{m}
\begin{pmatrix}
c_{\mu(l)}^{(p)} x_c \leq d_{\mu(l)}^{(p)} \\
\bigwedge_{j=1}^{k} (c_{\mu(l)}^{(p)} - c_j^{(p)})^t x_c \geq d_{\mu(l)}^{(p)} - d_j^{(p)} \\
(c_{\mu(l)}^{(p)})^t (A_l + \lambda I) x_c \geq \epsilon - (c_{\mu(l)}^{(p)})^t b_l + \lambda d_{\mu(l)}^{(p)}
\end{pmatrix}
\tag{19}
$$

A well known result in linear programming (Cf. Bertsimas and Tsitsiklis, Ch. 8 [9]) states that for any polyhedron $Ax \leq b$ where A, b have integer entries, where x is a $n \times 1$ vector of unknowns, if there is a feasible point then

there exists a feasible point that satisfies $\|x\|_\infty \leq (nU)^n$, wherein U is the absolute value of the largest entry in A, b. The result follows by multiplying both sides of each inequality by the LCM L^2 to convert them into integer values. The reason one needs to multiply by L^2 is to ensure that the last inequality, which involves $c^t \times (A_l + \lambda I)$ has integer entries.

5.4 Refining the Search Tree

Suppose a leaf node β were chosen as the node to be explored by the tree search, and the corresponding candidate $B^{(p)}$ resulted in a witness $x_p \in \mathbb{R}^n$ that fails one of the verification conditions **(C1)**, **(C2p)** or **(C3)**, we will refine β by adding children $\beta_{i,j}$ for each $i \in [k]$ and $j \in [m]$. The child $\beta_{i,j}$ has the following attributes:

$$W(\beta_{i,j}) := W(\beta) \cup \{(x_p, j, i)\}, \text{ and}$$
$$C(\beta_{i,j}) := C(\beta) \wedge CW(\{(x_p, j, i)\}). \tag{20}$$

The constraint $C(\beta_{i,j})$ eliminates the candidate chosen at the parent node β:

Lemma 29 *Let* $(c_i^{(p)}, d_i^{(p)})_{i \in [k]}$ *be the candidate with node* β *at the* p^{th} *iteration and* $\beta_{i,j}$ *be a child of* β. *then* $(c_i^{(p)}, d_i^{(p)})_{i \in [k]} \notin C(\beta_{i,j})$.

PROOF. Proof follows directly from the following three observations: x_p is a counterexample to one of the CBF conditions for the candidate $(c_i^{(p)}, d_i^{(p)})_{i \in [k]}$. However, $CW(\beta)$ includes $CW(\{(x_p, j, i)\})$, which constraints all the solutions to $C(\beta_{i,j})$ of the child node $\beta_{i,j}$ to satisfy the CBF conditions at point x_p with dynamic j and piece i being maximal. As a result, $(c_i^{(p)}, d_i^{(p)})_{i \in [k]} \notin C(\beta_{i,j})$.

Theorem 30 *If Alg. 1 returns a CBF, then it must satisfy constraints* **(C1)-(C3)**.

PROOF. Proof follows by a direct inspection of the algorithm and the verification procedure. Note that line 7 is the only place that returns a CBF and it is executed only if the verification of **(C1)-(C3)** succeed.

5.5 Termination

We adapt the argument in our previous work focusing on CLF synthesis to enable termination [7], as follows:

(1) The candidate $(c_i^{(p)}, d_i^{(p)})_{i \in [k]}$ is the center of the maximum volume inscribed ellipsoid (MVE-center) of the polyhedron $C(\beta)$ (line 4). This ensures that for each child $\beta_{i,j}$, the volume of $C(\beta_{i,j}) \leq \gamma C(\beta)$ where $\gamma < 1$ is a volume shrinking factor.
(2) We will stop exploration of a leaf node β whenever $C(\beta)$ is empty or the Chebyshev radius of $C(\beta)$ is below a cutoff threshold $R_{\min}$ (line 3).

Theorem 31 *Algorithm 1 terminates in worst case* $O\left((nk)^{O((n+1)^2 k^2)}\right)$ *iterations, wherein each iteration's cost is dominated by solving a mixed-integer optimization problem corresponding to* (17) *with* $O(mk)$ *binary variables,* $(n+1)k$ *real-valued variables and* $O(mk^2)$ *constraints.*

5.5.1 Proof of Theorem 31

Let α_0 be the root of our search tree. Its volume $V_0 = (2\gamma)^{k(n+1)}$. Next, let α be any node explored by the algorithm. Since we require the Chebyshev radius of $CW(\alpha)$ to be at least R_{min}, we have the following lemma.

Lemma 32 *If a polyhedron* P *has Chebyshev radius at least* $R_{\min}$ *then its volume* $\mathsf{vol}(P) \geq (KR_{\min})^{k(n+1)}$ *for some constant* K.

PROOF. Proof follows from the observation that P must contain a sphere of radius at least $R_{\min}$ over $k(n+1)$ dimensions.

Next, consider two nodes in the tree, parent α and its immediate child β. We have the following relationship between the volumes of the polyhedra $CW(\alpha)$ and $CW(\beta)$.

Lemma 33 *Let* $(c_i, d_i)_{i=1}^k$ *represent the MVE center of* $CW(\alpha)$. *It follows that* $CW(\beta) \subseteq CW(\alpha)$ *and* $(c_i, d_i)_{i \in [k]} \notin CW(\beta)$.

PROOF. Since β is the child of α, we note that $CW(\alpha)$ is feasible (see Algorithm 1, line 3). The MVE center $(c_i, d_i)_{i=1}^k$ of $CW(\alpha)$ is chosen as the candidate. Also, since we created β, this candidate must have failed verification. Therefore, following Theorem 29, we obtain that $(c_i, d_i)_{i=1}^k \notin CW(\beta)$. Also, by definition of $CW(\beta)$ (Eq. (20)), we note that $CW(\beta) \subseteq CW(\alpha)$.

Lemma 34 *The volumes of the polyhedra* $CW(\alpha)$ *and* $CW(\beta)$ *satisfy the relationship*

$$\mathsf{vol}(CW(\beta)) \leq (1 - \gamma)\mathsf{vol}(CW(\alpha)),$$

wherein $\gamma = \frac{1}{(n+1)k}$.

10

PROOF. Proof follows from the so-called "method of inscribed ellipsoids" proposed by Tarasov et al [34] and explained by Boyd and Vandenberghe [10]. Since $CW(\beta) \subseteq CW(\alpha)$ and excludes its MVE-center, we have

$$\mathsf{vol}(CW(\beta)) \le (1-\gamma)\mathsf{vol}(CW(\alpha)),$$

wherein $\gamma = \frac{1}{(n+1)k}$.

Lemma 35 *The depth of any branch of the search tree cannot exceed the bound: $O((n+1)^2 k^2)$.*

PROOF. Note that for each branch the volume of the polyhedron at the root is $V_0 : (2\gamma)^{nk+k}$ and the search terminates whenever a descendant α has $\mathsf{vol}(CW(\alpha)) \le KR_{\min}^{nk+k}$. Furthermore, at each step, the volume of a child node is at most $1-\gamma$ times the volume of its parent. Combining, we obtain that the volume of a node α_D at depth D must be bounded by

$$KR_{\min}^{nk+k} \le \mathsf{vol}(\alpha_D) \le (1-\gamma)^D V_0.$$

Therefore, we have

$$D \le \frac{\log(KR_{\min})^{nk+k} - \log(2\gamma)^{nk+k}}{\log(1-\gamma)}$$
$$\le \frac{1}{\gamma}(nk+k)(\log(KR_{\min}) - \log(2\gamma)) = O((n+1)^2 k^2)$$

Recall that $\gamma = (n+1)k$ and the useful inequality $-\log(1-\frac{1}{r}) \ge \frac{1}{r}$ for $r \ge 2$.

The proof of theorem 31 is completed by observing that the search tree has depth at most $D = O((n+1)^2 k^2)$ and branching factor of at most nk. The cost at each step is dominated by solving a verification problem which involves solving a MILP.

5.6 From Max to Min Barrier Certificates

The process of synthesizing barrier certificates of the form $\min_{i \in [k]}(c_i^t x - d_i)$ is very similar to that shown in Algorithm 1. We briefly sketch the differences, noting that our implementation of Alg. 1 handles the synthesis of min and max barrier functions.

The key differences are three-fold: (a) the verification procedure needs to be modified appropriately to check **(C3)** for the min-barrier condition stated in Def. 21 from Section 4.3; (b) The definition of $CW(W(\alpha))$ for a node α (Eq. (15)) must be modified as follows:

$$\bigwedge_{(x,j,i) \in W(\alpha)} \begin{pmatrix} \bigwedge_{i' \in [k]} c_i^t x \le c_{i'}^t x \land \\ c_i^t x - d_i \le 0 \land \\ c_i^t(A_j x + b_j) \le -\lambda(c_i^t x - d_i) - \epsilon \end{pmatrix} \quad (21)$$

The key change is shown in red. Finally, (c) the initialization constraints in Section 5.1 need to be modified noting that $B(x) = \min(\cdots)$. Specifically, the techniques used to handle **(C1)** for the max-barrier case will now be used to handle **(C2)** for the min-barrier case and vice-versa.

6 Min-Max Multiple Barriers

The exponential cost for Algorithm 1 is not surprising given our previous work that has established the NP-hardness of elementary problems such as checking polyhedral Lyapunov functions [7]. However, compared to Lyapunov functions, the synthesis of CBFs has two distinct advantages: (a) CBFs establish control invariance for a subset of the state-space: as a result, we can attempt to synthesize CBFs with small values of k such as $k = 2$, with the caveat that they may yield relatively small control invariant sets; and (b) to offset the disadvantage, we can combine the control invariant region $CI(B_j) := \{x \mid B_j(x) \le 0\}$ for multiple control barrier functions. Let B_j for $j = 1, \ldots, N$ be a set of CBFs synthesized by our algorithm. These could include barriers formed by max and min of affine functions. Note that $\mathcal{C} = \bigcup_{j=1}^N CI(B_j)$ is a control invariant region, since for any $x \in CI(B_j)$, we have a control strategy that keeps our system inside $CI(B_j)$. The region $\mathcal{C}$ itself can be viewed as the zero sublevel set of the function $B(x) = \min_{j=1}^N (B_j(x))$. This yields a so-called "mini-max" control barrier function obtained by combining min and max barrier functions.

Computing Multiple CBFs: In order to compute multiple CBFs, we adapt the following strategy proposed by Wajid and Sankaranarayanan for the case of polynomial CBFs [36]. As inputs to the CBF problem, we have an unsafe set X_u but do not include an initial set X_0 since our goal is to compute as large a control invariant set as possible. To do so, we select a set of random test ponts $X_t = \{x_1, \ldots, x_N\}$ wherein $x_i \notin X_u$. Next, we apply Algorithm 1 or its counterpart for $\min$ $-$barrier functions using $X_0 = \{x_i\}$ in order to attempt synthesis of a CBF that *separates* x_i from the unsafe set X_u. The final result is a union of all the control invariant sets thus obtained.

7 Empirical Results

In this section, we use our algorithm to synthesize a min-max control barrier function for some switched systems. At first, we study a numerical example. Then we consider a DC-DC converter as a switched affine system to synthesize control barrier functions and switching controllers. Finally, we will utilize the algorithm for higher order systems as a multi-agent system and car's velocity which are modeled by switched dynamics. Table 1 shows the results from these benchmarks at a glance. All the
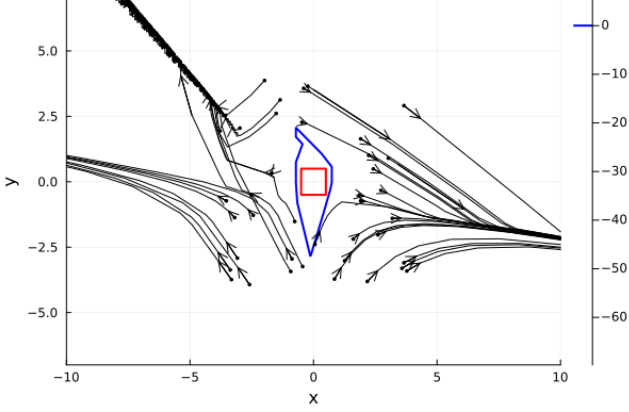
Fig. 3. Zero level set of the min-max barrier $B^*(x) = 0$ (blue lines), unsafe set (red lines), the trajectories for each initial conditions (black lines) for Ex. 36.

synthesized CBFs are provided in the extended version of this paper [22].

Example 36 *Consider a system with 2 state variables and 3 dynamical modes to choose from:*

$$A_1 = \begin{bmatrix} 1 & -1 \\ -0.5 & -2 \end{bmatrix}, \ A_2 = \begin{bmatrix} -2 & 1 \\ 0.5 & -1 \end{bmatrix}, \ A_3 = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$$

$$b_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \ b_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \ b_3 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

The unsafe set is defined as $X_u = [-0.5, 0.5] \times [-0.5, 0.5]$. Using the proposed algorithm, we synthesize a min-max multiple barrier function with $k = 2$. We consider various test points $x_t \notin X_u$ a and for each of them we attempt to find c_i and d_i for a CBF that "separates" x_t as the initial point from the unsafe set X_u. For efficiency, we truncate the search whenever the depth of the tree exceeds $D_{\max} = 6$. We synthesized 11 max barrier functions and 11 min barrier functions, in all. Figure 4 shows the synthesized min-max barrier functions. Figure 3 plots the zero level set of this function and shows the simulations of the trajectories for the corresponding switching rule for randomly chosen initial conditions.

Example 37 (DC-DC Converter) *The dynamics for this DC-DC converter are taken from Ravanbakhsh et al [28]. The switched model of this system has two modes ($l = 2$) with dynamics $\dot{x}(t) = A_l x(t) + b_l$ in which $x(t)$ denotes the state variables $[i(t), v(t)]$. $i(t)$ and $v(t)$ are current and voltage of the converter respectively.*

$$l = 1: \quad A_1 = \begin{bmatrix} 0.0167 & 0 \\ 0 & -0.0142 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 0.3333 \\ 0 \end{bmatrix}$$

$$l = 2: \quad A_2 = \begin{bmatrix} -0.0183 & -0.0663 \\ -0.0711 & -0.0142 \end{bmatrix}, \quad b_2 = \begin{bmatrix} 0.3333 \\ 0 \end{bmatrix}$$

We assume the unsafe set $X_u = [0, 1] \times [0, 1]$. By considering the number of the pieces of the polyhedral function $k = 2$, and applying the proposed algorithm with a depth cutoff of 6 for 16 test points which yielded 14 CBFs each for the max and min cases. The final barrier depicted in Figure 5 combines all the barriers synthesized.

Example 38 (Multi-agent System) *Consider a multi-agent system with the following model:*

$$\dot{x}(t) = \Sigma_{i=1}^n u_i(t) g_i(x(t)).$$

We use a 3-dim Brockett integrator with two inputs [11]:

$$\begin{cases} \dot{x}_1(t) = u_1(t), \ \dot{x}_2(t) = u_2(t) \\ \dot{x}_3(t) = x_1 u_2 - x_2 u_1. \end{cases}$$

In which, $x_i(t) \in R$ is a state that shows the location of ith agent, and $u_i(t) \in R$ is a control signal applied to the corresponding agent. It has been proved that this non holonomic system cannot be asymptotically stabilized by a continuous feedback law [25]. So, we are considering a piece-wise constant control signal $u_1(t) \in \{-\alpha, 0, \alpha\}$ and $u_2(t) \in \{-\beta, 0, \beta\}$ with constants $\alpha, \beta \in \mathbb{R}_{>0}$. By this control signals, the system is modeled as a switched affine system $\dot{x}(t) = A_{i,j} x(t) + b_{i,j}$, wherein $i \in [3], j \in [3]$ reflect the fixed choices of u_1, u_2. In this example we consider $\alpha = 1$ and $\beta = 1$ and the unsafe set $[0, 1] \times [0, 1] \times [0, 1]$. Algorithm 1 was run for 6 test points with $k = 3$ and maximum depth cutoff of 6. Table 1 summarizes the performance of our approach on this benchmark.

Example 39 (Vehicle Model) *For this example, consider the follwing dynamic.*

$$\begin{cases} \dot{x} = V_x, \\ \dot{y} = V_y, \\ \dot{z} = V_z, \end{cases} \begin{bmatrix} \dot{V}_x \\ \dot{V}_y \\ \dot{V}_z \end{bmatrix} = A_{3 \times 3} \begin{bmatrix} V_x - V_{xref} \\ V_y - V_{yref} \\ V_z - V_{zref} \end{bmatrix}$$

wherein

$$A = \begin{bmatrix} -2.5 & 0.8 & 1.2 \\ -1.5 & -2.2 & 0.7 \\ 0.5 & -1.8 & -2.7 \end{bmatrix}.$$

To control the movement, the references' velocities can apply in various directions which results in a switched affine systems. In this example, we have considered 6 directions by defining $[V_{xref}; V_{yref}; V_{zref}]$ as

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}$$

Table 1
Summary of results for the empirical evaluation. max refers to max-CBFs while min refers to min-CBFs. $D_{\max}$ refers to cutoff depth for the tree search.

ID	n	m	#Test Pts.		#CBFs		$D_{\max}$		Time (sec)	
			max	min	max	min	max	min	max	min
Ex.36	2	3	11	11	11	11	6	6	53.5	20.4
Ex.37	2	2	16	14	16	14	6	6	16.3	6.8
Ex.38	3	9	6	6	6	6	6	6	26.6	31.8
Ex.39	6	6	7	-	7	-	14	-	885.61	-

$$\min\begin{pmatrix} \min(-7.32x_1 - 5.43, -9.37x_1 + 0.89x_2 - 9.37),\ \min(-4.02x_1 - 2.96x_2 - 7.52, -5.22x_1 - 1.76x_2 - 7.52), \\ \min(9.04x_1 - 6.7, -9.04x_1 + 1.2x_2 - 9.04), \min(8.83x_1 + 2.07x_2 - 7.88, -9.04x_1 + 1.19x_2 - 8.83), \\ \min(-9.8x_1 + 3.12x_2 - 7.7, 5.86x_1 + 2.27x_2 - 9.8), \min(-8.98x_1 + 1.08x_2 - 7.28, -8.98x_1 - 0.63x_2 - 8.98) \\ \cdots, \\ \max(-9.04x_1 - 6.69, -9.04x_1 + 9.04x_2 + 9.04), \max(-8.18x_1 - 0.09x_2 - 7.36, -8.18x_1 + 3.8x_2 + 8.18), \\ \max(9.04x_1 - 6.69, 9.04x_1 + 9.04x_2 + 9.04), \max(9.22x_1 - 6.57, 4.33x_1 - 9.22x_2 + 7.25), \\ \max(-9.87x_1 + 3.00x_2 - 7.6, -4.35x_1 - 2.55x_2 + 9.07), \max(-8.43x_1 + 3.56x_2 - 8.65, -4.91x_1 - 1.24x_2 - 0.32), \\ \cdots \end{pmatrix}$$

Fig. 4. The overall "mini-max" CBF $B^*(x)$ synthesized for Ex. 36 through the repeated application of Algorithm 1 over 11 test points. The overall CBF has 22 terms, each being the min or max of two affine functions over x_1, x_2, of which 12 terms are shown. Coefficients have been rounded to two significant figures for readability.
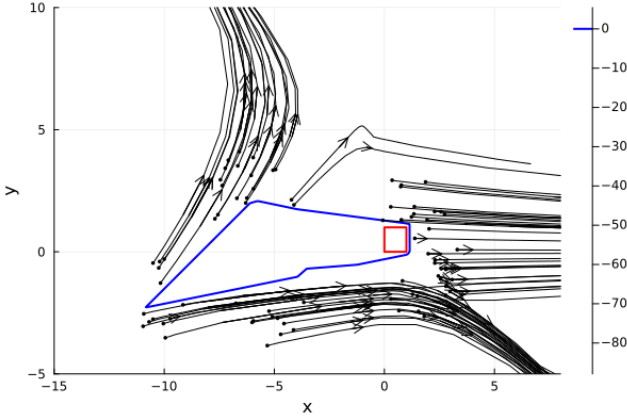

Fig. 5. Zero level set of the min-max barrier $B^*(x) = 0$ (blue lines), unsafe set (red lines), trajectories for various initial conditions (black lines) for Ex. 37.

which results in a switched affine systems with 6 subsystems. The given unsafe set is $[-0.5, 0.5]^3 \times [-10, 10]^3$. For this example, we ran 7 test points for the max barrier functions and note that an additional set of 7 test points timed out after 1 hour. We did not attempt to synthesize min barrier functions.

8 Conclusion

To conclude, we have demonstrated an approach that defines a class of non-smooth control barrier functions for switched affine systems through the application of piecewise max and min over affine functions. We have provided algorithms for synthesizing these control barrier function and demonstrated a scheme for extracting a feedback control from a CBF. Our future work seeks to expand our work to polynomial dynamics using ideas such as Sum-of-Squares (SOS) programming. We are also interested in the use of piecewise linearization of nonlinear dynamics, especially for applications to robust plan execution in robotic systems.

References

[1] Alessandro Abate, Daniele Ahmed, Alec Edwards, Mirco Giacobbe, and Andrea Peruffo. Fossil: a software tool for the formal synthesis of lyapunov functions and barrier certificates using neural networks. In *Proceedings of the 24th international conference on hybrid systems: computation and control*, pages 1–11, 2021.

[2] Daniele Ahmed, Andrea Peruffo, and Alessandro Abate. Automated and sound synthesis of lyapunov functions with smt solvers. In *Tools and Algorithms for the Construction and Analysis of Systems: 26th International Conference, TACAS 2020, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2020, Dublin, Ireland, April 25–30, 2020, Proceedings, Part I 26*, pages 97–114. Springer, 2020.

[3] Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European control conference (ECC)*, pages 3420–3431. IEEE, 2019.

[4] Aaron D Ames, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *53rd IEEE conference on decision and control*, pages 6271–6278. IEEE, 2014.

[5] Jean-Pierre Aubin and Hèléne Frankowska. *Continuity of Set-Valued Maps*, pages 1–39. Birkhäuser Boston, Boston, 2009.

[6] Andrea Bacciotti and Francesca Ceragioli. Stability and stabilization of discontinuous systems and nonsmooth lyapunov functions. *ESAIM: Control, Optimisation and Calculus of Variations*, 4:361–376, 1999.

[7] Guillaume O Berger and Sriram Sankaranarayanan. Learning fixed-complexity polyhedral lyapunov functions from counterexamples. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 3250–3255. IEEE, 2022.

[8] Guillaume O Berger and Sriram Sankaranarayanan. Counterexample-guided computation of polyhedral lyapunov functions for piecewise linear systems. *Automatica*, 155:111165, 2023.

[9] Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization.* Athena Scientific, Belmont, MA, 1997.

[10] Stephen Boyd and Lieven Vandenberghe. Localization and cutting-plane methods. *From Stanford EE 364b lecture notes*, 2007.

[11] R. W. Brockett. *Control Theory and Singular Riemannian Geometry.* Springer New York, New York, NY, 1982.

[12] Shaoru Chen, Mahyar Fazlyab, Manfred Morari, George J Pappas, and Victor M Preciado. Learning lyapunov functions for hybrid systems. In *Proceedings of the 24th International Conference on Hybrid Systems: Computation and Control*, pages 1–11, 2021.

[13] Vašek Chvátal. *Linear Programming.* Freeman, New York, 1983.

[14] Andrew Clark. A semi-algebraic framework for verification and synthesis of control barrier functions. *IEEE Transactions on Automatic Control*, 2024.

[15] Frank H. Clarke. *Optimization and Nonsmooth Analysis.* Canadian Mathematical Society series of monographs and advanced texts. Wiley-Interscience, New York, 1983.

[16] Frank H Clarke. *Optimization and nonsmooth analysis.* SIAM, 1990.

[17] Jorge Cortes. Discontinuous dynamical systems. *IEEE Control systems magazine*, 28(3):36–73, 2008.

[18] Paul Glotfelter, Ian Buckley, and Magnus Egerstedt. Hybrid nonsmooth barrier functions with applications to provably safe and composable collision avoidance for robotic systems. *IEEE Robotics and Automation Letters*, 4(2):1303–1310, 2019.

[19] Paul Glotfelter, Jorge Cortés, and Magnus Egerstedt. Nonsmooth barrier functions with applications to multi-robot systems. *IEEE control systems letters*, 1(2):310–315, 2017.

[20] Paul Glotfelter, Jorge Cortés, and Magnus Egerstedt. Boolean composability of constraints and control synthesis for multi-robot systems via nonsmooth control barrier functions. In *2018 IEEE Conference on Control Technology and Applications (CCTA)*, pages 897–902. IEEE, 2018.

[21] S. Kamali, G. Berger, and S. Sankaranarayanan. Polyhedral control lyapunov functions for switched affine systems. In *Proc. Hybrid Systems: Computation and Control*, HSCC'25. ACM, 2025.

[22] Sara Kamali, Guillaume O. Berger, and Sriram Sankaranarayanan. Synthesizing min-max control barrier functions for switched affine systems, 2025. Note: arXiv ePrint 2506.10296. https://arxiv.org/abs/2506.10296.

[23] Hui Kong, Fei He, Xiaoyu Song, William NN Hung, and Ming Gu. Exponential-condition-based barrier certificate generation for safety verification of hybrid systems. In *International Conference on Computer Aided Verification*, pages 242–257. Springer, 2013.

[24] V. Lakshmikantham and S. Leela. *Differential and Integral Inequalities: Theory and Applications, Volume I: Ordinary Differential Equations*, volume 55 of *Mathematics in Science and Engineering.* Academic Press, New York, 1969.

[25] Daniel Liberzon. *Switching in systems and control*, volume 190. Springer, 2003.

[26] Pio Ong, Beatrice Capelli, Lorenzo Sabattini, and Jorge Cortés. Nonsmooth control barrier function design of continuous constraints for network connectivity maintenance. *Automatica*, 156:111209, 2023.

[27] Stephen Prajna and Ali Jadbabaie. Safety verification of hybrid systems using barrier certificates. In *Hybrid Systems: Computation and Control (HSCC)*, volume 2993 of *Lecture Notes in Computer Science*, pages 477–492. Springer, 2004.

[28] Hadi Ravanbakhsh and Sriram Sankaranarayanan. A class of control certificates to ensure reach-while-stay for switched systems. *arXiv preprint arXiv:1711.10639*, 2017.

[29] Hadi Ravanbakhsh and Sriram Sankaranarayanan. Learning control lyapunov functions from counterexamples and demonstrations. *Autonomous Robots*, 43:275–307, 2018.

[30] Pouya Samanipour and Hasan A. Poonawala. Stability analysis and controller synthesis using single-hidden-layer relu neural networks. *IEEE Transactions on Automatic Control*, 69(1):202–213, 2024.

[31] Armando Solar-Lezama. *Program Synthesis by Sketching.* PhD thesis, University of California, Berkeley, Berkeley, CA, USA, 2008.

[32] Eduardo D Sontag. A lyapunov-like characterization of asymptotic controllability. *SIAM journal on control and optimization*, 21(3):462–471, 1983.

[33] Eduardo D. Sontag. A únniversalćonstruction of artstein's theorem on nonlinear stabilization. *Systems & Control Letters*, 13(2):117–123, 1989.

[34] S. P. Tarasov, L. G. Khachiyan, and I. I. Èrlikh. The method of inscribed ellipsoids. In *Dokl. Akad. Nauk SSSR*, volume 37, pages 226–230, 1988.

[35] Kim P Wabersich, Andrew J Taylor, Jason J Choi, Koushil Sreenath, Claire J Tomlin, Aaron D Ames, and Melanie N Zeilinger. Data-driven safety filters: Hamilton-jacobi reachability, control barrier functions, and predictive methods for uncertain systems. *IEEE Control Systems Magazine*, 43(5):137–177, 2023.

[36] Rameez Wajid and Sriram Sankaranarayanan. Successive control barrier functions for nonlinear systems. In *Hybrid Systems: Computation and Control (HSCC)*, pages 21:1–21:11. ACM, 2025.

[37] Peter Wieland and Frank Allgöwer. Constructive safety using control barrier functions. *IFAC Proceedings Volumes*, 40(12):462–467, 2007.

[38] Wei Xiao and Calin Belta. High-order control barrier functions. *IEEE Transactions on Automatic Control*, 67(7):3655–3662, 2022.

[39] Yuhan Xiong, Di-Hua Zhai, Mahdi Tavakoli, and Yuanqing Xia. Discrete-time control barrier function: High-order case and adaptive case. *IEEE Transactions on Cybernetics*, 53(5):3231–3239, 2023.

[40] Hongchao Zhang, Zhouchi Li, Hongkai Dai, and Andrew Clark. Efficient sum of squares-based verification and construction of control barrier functions by sampling on algebraic varieties. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 5384–5391. IEEE, 2023.