Stability Analysis Via Refinement Of Piece-wise Linear Lyapunov Functions

Hasan A. Poonawala

Abstract— **We present an algorithm for finding piece-wise linear Lyapunov functions that verify the asymptotic stability of piece-wise linear differential inclusions. Existing methods either use a fixed set of pieces (a partition) to define the Lyapunov function, or use heuristic methods to split the pieces, thereby refining the partition. Our algorithm involves iteratively refining partitions using an exact criterion which strictly reduces the set of points over which the Lyapunov function is non-decreasing.**

I. Introduction

Lyapunov's second method is widely used to verify the stability properties of dynamical systems. This method involves finding a real-valued function of the state – called a Lyapunov function – whose value decreases along solutions of the dynamical system. There is no general method to search for a Lyapunov function for all dynamical systems. Often, a parametrized structure for the Lyapunov function is chosen, given a parametrized structure of the dynamical system. The search for the parameters of a Lyapunov function then exploits this structure. For example, a common method representing this search is to formulate a (convex) optimization problem whose solution yields a Lyapunov function certifying stability of the dynamical system.

This paper focuses on searching for piece-wise linear Lyapunov functions to analyze the properties of piece-wise linear differential inclusions. The pieces defining these differential inclusions are cones with apices at the origin. Blanchini [1] first proposed using piece-wise linear Lyapunov functions to analyze uncertain linear dynamical systems. Later, Johansson proposed conditions [2] that verify whether a candidate piece-wise linear Lyapunov function is a valid Lyapunov function. The parameters of this Lyapunov function include the partition (pieces) over which the Lyapunov function is defined, and the linear function associated with each piece in the partition. The conditions in [2] facilitate an efficient search for the linear functions associated with the pieces in a fixed partition. This search takes the form of a linear program.

Johansson suggests [2] refining pieces by partitioning them as one way to change the Lyapunov function when it fails to satisfy the stability condition. The main idea he proposes is to solve the dual problem to the linear program used for verifying stability, and to then use the optimal dual variables to determine which cell should be refined. The refinement step involves splitting a polyhedral cell by splitting one of its facets. The properties of this heuristic approach are not investigated. This approach is a form of counter-example guided refinement (CEGAR). Other authors propose CEGAR-based search methods for piece-wise constant dynamics [3] or piece-wise quadratic Lyapunov functions [4].

Contribution: Our main contribution is to propose a method to partition the pieces defining a candidate piece-wise linear Lyapunov function when verifying the stability properties of piece-wise linear differential inclusions, where the pieces are polyhedral cones with apices at the origin. The refinement-based approach leads to a sequential optimization-based algorithm for searching for such functions. Our refinement step enables precise analysis of this search method.

Overview: Sections II and III introduce the piece-wise linear differential inclusions we consider, piece-wise linear Lyapunov functions, and the conditions on their parameters that lead to verification of the system's stability properties. Section IV describes the refinement procedure we propose and the resulting algorithm for finding piece-wise linear Lyapunov functions for verifying piece-wise linear differential inclusions. Section V demonstrates the utility of our algorithm by applying it to example dynamical systems.

II. Preliminaries

Notation: The indices of the elements of a finite set S form the set I_S. We denote the convex hull of a set S by $\mathrm{conv}(S)$, and the interior of S by $\mathrm{Int}(S)$. $\mathrm{pos}(S)$ and $\mathrm{pos}_{>0}(S)$ respectively denote the set of non-negative and strictly positive combinations of the elements of a set S.

The vector $\mathbf{1}_n \in \mathbb{R}^n$ has all elements equal to unity. We omit the subscript n if its value is clear from the context. The set $GL(n)$ represents the set of $n \times n$ matrices with non-zero determinant. The rank of a matrix $E \in \mathbb{R}^{m \times n}$ is given by $r(E)$.

A. Partitions And Refinements

A partition $\mathcal{P}$ is a collection of subsets $\{X_i\}_{i \in I_\mathcal{P}}$; where $I_\mathcal{P}$ is an index set, $X_i \subseteq \mathbb{R}^n$ for each $i \in I_\mathcal{P}$, $n \in \mathbb{N}$, and $Int(X_i) \cap Int(X_j) = \emptyset$ for each pair $i, j \in I_\mathcal{P}$ such that $i \neq j$. We refer to $\cup_{i \in I_\mathcal{P}} X_i$ as the domain of $\mathcal{P}$, which we also denote by $\mathrm{Dom}(\mathcal{P})$. We also refer to the subsets X_i in $\mathcal{P}$ as the cells of the partition. Note that our definition of a partition allows some cells in $\mathcal{P}$ to be the boundary of other cells in $\mathcal{P}$, which is useful for handling sliding modes.

Let $\mathcal{P} = \{Y_i\}_{i \in I}$ and $\mathcal{R} = \{Z_j\}_{j \in J}$ be two partitions of a set $S = \mathrm{Dom}(\mathcal{P}) = \mathrm{Dom}(\mathcal{R})$. A partition $\mathcal{R}$ is a *refinement* of $\mathcal{P}$ if $Z_j \cap Y_i \neq \emptyset$ implies that $Z_j \subseteq Y_i$. We denote the set of

Hasan A. Poonawala is with the Department of Mechanical Engineering, University of Kentucky, Lexington, KY 40506, USA. hasan.poonawala@uky.edu

refinements of a partition $\mathcal{P}$ as $\mathrm{Ref}(\mathcal{P})$. There exists a natural abstraction function $\pi_{\mathcal{R},\mathcal{P}}\colon I_{\mathcal{R}} \mapsto I_{\mathcal{P}}$, given by $\pi_{\mathcal{R},\mathcal{P}}(j) = \{i \in I_{\mathcal{P}}\colon Z_j \subseteq Y_i\}$.

B. Polyhedral Cones

We consider partitions $\mathcal{P}$ where each cell $X \in \mathcal{P}$ is a polyhedral cone with apex at the origin. Each cell $X \in \mathcal{P}$ is of the form $X = \{x \in \mathbb{R}^n\colon Ex \geq 0\}$, where $E \in \mathbb{R}^{m\times n}$ and its rank $r(E) = n$. The i^{th} row E^i of matrix E defines a hyperplane $\{x \in \mathbb{R}^n\colon E^i x = 0\}$.

The dual vector space to $\mathbb{R}^n$ is $(\mathbb{R}^n)^*$. The dual cone Z^* to a convex set Z is the set $\{y \in (\mathbb{R}^n)^*\colon \langle y, x\rangle \geq 0 \forall x \in Z\}$, where $\langle\cdot,\cdot\rangle\colon Z^*$ is the evaluation of $x \in \mathbb{R}^n$ by the functional $y \in (\mathbb{R}^n)^*$. This cone lies in the dual space to $\mathbb{R}^n$, however since $\mathbb{R}^n$ is self-dual, we can identify the dual cone with a cone in $\mathbb{R}^n$. Some authors refer to this second cone as the internal dual cone.

If a polyhedral cone $X \in \mathbb{R}^n$ is given by

$$X = \{x \in \mathbb{R}^n\colon Ex \geq 0\}, \tag{1}$$

its dual cone $X^* \in (\mathbb{R}^n)^*$ is given by

$$X^* = \mathrm{pos}\left(\{E^j\}\right),$$

and its interior $\mathrm{Int}(X^*) \in (\mathbb{R}^n)^*$ is given by

$$\mathrm{Int}(X^*) = \mathrm{pos}_{>0}\left(\{E^j\}\right). \tag{2}$$

The polar cone $X^\circ \in (\mathbb{R}^n)^*$ of X is simply $-X^*$.

C. Piece-wise Linear Differential Inclusions

A piece-wise linear differential inclusion $\Omega_{\mathcal{P}}$ associated with partition $\mathcal{P} = \{X_j\}_{j\in I_{\mathcal{P}}}$ is a collection,

$$\Omega_{\mathcal{P}} = \{\mathcal{A}_j(x)\}_{j\in I_{\mathcal{P}}} \tag{3}$$

that to each cell $X_j \in \mathcal{P}$ assigns the affine differential inclusion $\mathcal{A}_j(x) = \mathrm{conv}\left(\{A_{jk}x\}_{k\in I_{\mathcal{A}_j}}\right)$. Therefore,

$$\dot{x}(t) \in \mathcal{A}_j(x), \text{ if } x(t) \in X_j. \tag{4}$$

The cell $X_j \in \mathcal{P}$ is given by

$$X_i = \{x \in \mathbb{R}^n\colon F_j x \geq 0\}. \tag{5}$$

We assume that F_j is full rank for each $j \in I_{\mathcal{P}}$. Furthermore, we assume that $0 \in \mathrm{Int}(\mathrm{Dom}(\mathcal{P}))$. This assumption ensures that X_j is a polyhedral cone with apex at the origin and non-empty interior.

D. Piece-wise Linear Lyapunov Functions

Blanchini [1] advocated use of piece-wise linear (PWL) Lyapunov functions for stability analysis. Later, computational methods to find such functions, given a partition of the state space, were introduced [2], [5]. If the level sets of the PWL Lyapunov function are convex polyhedra, then the Lyapunov function is also known as a polytopic function.

We parameterize a continuous PWL Lyapunov function $V_{\mathcal{Q}}(x)$ with a partition $\mathcal{Q} = \{Z_i\}_{i\in I_{\mathcal{Q}}}$ and a collection of vectors $\{p_i\}_{i\in I_{\mathcal{Q}}}$ such that $V_{\mathcal{Q}}(x) = p_i^T x$, if $x \in Z_i \subseteq \mathbb{R}^n$. Each set $Z_i \in \mathcal{Q}$ is given by $Z_i = \{x \in \mathbb{R}^n\colon E_i x \geq 0\}$, and we assume that $r(E_i) = n$. This definition of $V_{\mathcal{Q}}(x)$ corresponds to polytopic Lyapunov functions [1], [2], which is a subclass of piece-wise linear Lyapunov functions.

III. Stability Conditions

Our concern is with the stability properties of the origin $x = 0$ of a piece-wise linear differential inclusion $\Omega_{\mathcal{P}}$ in (3). We consider a candidate PWL Lyapunov function $V_{\mathcal{Q}}(x)$, where $\mathcal{Q}$ is a refinement (see Section II-A) of $\mathcal{P}$, i.e. $\mathcal{Q} \in \mathrm{Ref}(\mathcal{P})$. To $\mathcal{Q}$, we associate the parameters $\{p_i\}_{i\in I_{\mathcal{Q}}}$. The abstraction function $\pi_{\mathcal{Q},\mathcal{P}}$ associates each cell in $\mathcal{Q}$ with a unique cell in $\mathcal{P}$. Most methods for finding piece-wise Lyapunov function choose $\mathcal{Q} = \mathcal{P}$, so that $\pi_{\mathcal{Q},\mathcal{P}}$ is the identity map.

To establish stability [6], for each $Z_i \in \mathcal{Q}$, we need to verify that $p_i^T A_{jk} x < 0$ for all $x \in Z_i$, and for all $k \in I_{\mathcal{A}_j}$, where $j = \pi_{\mathcal{Q},\mathcal{P}}(i)$. Lemma 1 below shows how to convert verification of sign definiteness of a linear function on all points in a cone into a constrained optimization problem.

Lemma 1 ([2], [7])**.** *Let $Z = \{x \in R^n\colon Ex \geq 0\}$ where $E \in \mathbb{R}^{m\times n}$ has rank $r(E) = n$ and $Z\backslash\{0\}$ is non-empty. Given $v \in \mathbb{R}^n$, the following are equivalent*

i) $Ex \geq 0, x \neq 0 \implies v^T x > 0$
ii) *$\exists \mu \in \mathbb{R}^n$ such that $\mu > 0$ and $v = E^T\mu$*
iii) $v \in \mathrm{Int}(Z^*)$

By considering all the cells in our partition $\mathcal{Q}$, Lemma 1, and stability results from [6], we formulate a constrained optimization problem below that will form the basis of our search for a PWL Lyapunov function. To iterate over all these combinations, we use the index set $I_{dec}(\mathcal{Q})$, so that if $(i, j, k) \in I_{dec}(\mathcal{Q})$ then $i \in I_{\mathcal{Q}}$ and $k \in I_{\mathcal{A}_j}$, where $j = \pi_{\mathcal{Q},\mathcal{P}}(i)$. We also need to ensure that $V_{\mathcal{Q}}$ is continuous at the boundaries $Z_i \cap Z_j$ of cells in $\mathcal{Q}$. Let $I_{cont}(\mathcal{Q}) \subseteq I_{\mathcal{Q}} \times I_{\mathcal{Q}}$ be the set of pairs of indices (i, j) such that $Z_i \cap Z_j \neq \emptyset$. The cells in $\mathcal{Q}$ are convex; each such boundary is contained in a single half-plane parameterized by a vector η_{ij}, so that

$$Z_i \cap Z_j \subset \{x \in \mathbb{R}^n\colon \eta_{ij}^T x = 0\}. \tag{6}$$

The subscript dec and $cont$ are abbreviations of 'decrease' and 'continuity' respectively. The conditions for asymptotic stability of the origin of the piece-wise linear differential inclusion using a PWL Lyapunov function is given in the following result, adapted from [8], [2].

Lemma 2. *Let $\Omega_{\mathcal{P}}$ be a piece-wise linear differential inclusion as in (3). Let $V_{\mathcal{Q}}$ be a candidate PWL Lyapunov function with partition $\mathcal{Q} = \{Z_i\}_{i\in I_{\mathcal{Q}}}$, parameters $\{p_i\}_{i\in I_{\mathcal{Q}}}$, and abstraction function $\pi_{\mathcal{Q},\mathcal{P}}$. Each set Z_i is given by $\{x \in \mathbb{R}^n\colon E_i x \geq 0\}$. Let $I_{\mathcal{Q}}$, $I_{dec}(\mathcal{Q})$ and $I_{cont}(\mathcal{Q})$ be the index sets associated with $\Omega_{\mathcal{P}}$ and $V_{\mathcal{Q}}$. If the set of constraints*

$$p_i = E_i^T \mu_i, \quad \forall i \in I_{\mathcal{Q}}, \tag{7}$$

$$\mu_i \geq \mathbf{1}, \quad \forall i \in I_{\mathcal{Q}}, \tag{8}$$

$$E_i^T \nu_{ijk} = -A_{jk}^T p_i, \quad \forall (i,j,k) \in I_{dec}(\mathcal{Q}), \tag{9}$$

$$\nu_{ijk} \geq \mathbf{1}, \quad \forall (i,j,k) \in I_{dec}(\mathcal{Q}), \tag{10}$$

$$p_i - p_j = \lambda_{ij}\eta_{ij}, \quad \forall (i,j) \in I_{cont}(\mathcal{Q}) \tag{11}$$

is feasible then the origin of $\Omega_{\mathcal{P}}$ is asymptotically stable.

Constraints (7) and (8) ensure that $V_{\mathcal{Q}}(x)$ is positive definite, and constraint (11) ensures that $V_{\mathcal{Q}}(x)$ is continuous.

IV. Refining Lyapunov Functions

Verifying the asymptotic stability of the origin relies on showing that the candidate Lyapunov function decreases along all trajectories passing through every point in a neighborhood of the origin. The core of our refinement algorithm is based on the idea that if a candidate Lyapunov function does not decrease along all trajectories at every point in a cell, we may be able to precisely split the cell into two cells where the Lyapunov function decreases at all points in one of those cells. Successive refinements using this principle may lead to identification of a valid PWL Lyapunov function for the system if one exists. The rest of this section makes the statements above precise and presents the refinement algorithm (in Section IV-C).

A. Identifying Cells To Refine

Let $\Omega_{\mathcal{P}}$ be a piece-wise linear differential inclusion and let $\mathcal{Q}$ be a refinement of $\mathcal{P}$ with abstraction function $\pi_{\mathcal{Q},\mathcal{P}}$. The index set of $\mathcal{Q}$ is $I_{\mathcal{Q}}$. We define index sets $I_{dec}(\mathcal{Q})$ and $I_{cont}(\mathcal{Q})$ as was done in Section III.

If the trajectories of $\Omega_{\mathcal{P}}$ do not decrease along any valid set of parameters $\{p_i\}_{i\in I_{\mathcal{Q}}}$ corresponding to $V_{\mathcal{Q}}(x)$, then (9) and (10) will be infeasible for some cells in $\mathcal{Q}$. Equation (2) shows that (9) and (10) together are equivalent to the requirement that for each multi-index $(i,j,k)\in I_{dec}(\mathcal{Q})$,

$$-p_i^T A_{jk} \in \mathrm{Int}(Z_i^*). \tag{12}$$

Let $(i,j,k)\in I_{dec}(\mathcal{Q})$ be a multi-index for which (12) does not hold. Then, $A_{jk}^T p_i + E_i^T \nu \neq 0$ for any vector $\nu > 0$. For each such (i,j,k), we define the quantity $s_{i,k}^{\mathcal{Q}}(p)$ given by $s_{i,k}^{\mathcal{Q}}(p) = E_i^T\nu_{ijk} + A_{jk}^T p$. We then define the scalar $d_i^{\mathcal{Q}}(p,\bar{\nu}_i)$ as

$$d_i^{\mathcal{Q}}(p,\bar{\nu}_i) = \sum_{k\in I_{\mathcal{A}_j}} \min_{\nu_{ijk}\geq\bar{\nu}_i} \|s_{i,k}^{\mathcal{Q}}(p,\nu_{ijk})\|, \tag{13}$$

where $j=\pi_{\mathcal{Q},\mathcal{P}}(i)$, and $\bar{\nu}_i$ is a constant vector associated with each cell in partition $\mathcal{Q}$, given by

$$\bar{\nu}_i > 0, \text{ and } E_i^T\bar{\nu}_i = F_j^T\mathbf{1}. \tag{14}$$

Vector $\bar{\nu}_i$ always exists when $X_j\backslash\{0\}$ is non-empty. By Lemma 1,

$$\begin{aligned} &-p_i^T A_{jk} \in \mathrm{Int}(Z_i^*) \;\; \forall k\in I_{\mathcal{A}_j} \\ &\qquad \iff \exists\bar{\nu}_i > 0 \text{ such that } d_i^{\mathcal{Q}}(p_i,\bar{\nu}_i) = 0. \end{aligned} \tag{15}$$

Therefore, the quantity $d_i^{\mathcal{Q}}(p_i,\bar{\nu}_i)$ becomes an indicator for the infeasibility of constraints (9) and (10) for some $i\in I_{\mathcal{Q}}$. To compute this indicator, we define an objective function as

$$J_{\Omega_{\mathcal{P}},\mathcal{Q}} = \sum_{i\in I_{\mathcal{Q}}}\sum_{k\in I_{\mathcal{A}_j}} \min_{\nu_{ijk}\geq\bar{\nu}_i} \|s_{i,k}^{\mathcal{Q}}(p,\nu_{ijk})\|, \tag{16}$$

where $j=\pi_{\mathcal{Q},\mathcal{P}}(i)$ and $s_{i,k}^{\mathcal{Q}}(p,\nu_{ijk})$ is like a slack variable for constraint (9) for each $(i,j,k)\in I_{dec}(\mathcal{Q})$.

We minimize objective function (16) subject to constraints (7)-(11) to obtain the full (convex) optimization problem below.

$$\min_{p_i,\mu_i,\nu_{ijk},\lambda_{ij}} \quad J_{\Omega_{\mathcal{P}},\mathcal{Q}} \tag{17}$$
$$\text{s.t.}$$
$$p_i = E_i^T\mu_i, \quad \forall i\in I_{\mathcal{Q}}, \tag{18}$$
$$\mu_i \geq \mathbf{1}, \quad \forall i\in I_{\mathcal{Q}}, \tag{19}$$
$$\nu_{ijk}\geq\bar{\nu}_i, \quad \forall (i,j,k)\in I_{dec}(\mathcal{Q}), \tag{20}$$
$$p_i - p_j = \lambda_{ij}\eta_{ij}, \quad \forall (i,j)\in I_{cont}(\mathcal{Q}). \tag{21}$$

If the optimal solution is given by $p_i^\star$, $\mu_i^\star$, $\nu_{ijk}^\star$, and $\lambda_{ij}^\star$ over the usual index sets, then the objective value $J_{\Omega_{\mathcal{P}},\mathcal{Q}}^\star$ at this optimum solution is

$$J_{\Omega_{\mathcal{P}},\mathcal{Q}}^\star = \sum_{i\in I_{\mathcal{Q}}} d_i^{\mathcal{Q}}(p_i^\star,\bar{\nu}_i). \tag{22}$$

According to (15), cells $Z_i\in\mathcal{Q}$ such that $d_i^{\mathcal{Q}}(p_i^\star,\bar{\nu}_i)\neq 0$ may need to be refined. The (convex) optimization problem (17)-(21) has the following properties by construction:

Proposition 3. *The optimization problem* (17)-(21) *always has a feasible solution.*

Proposition 4. *If the optimal value of* (17)-(21) *is zero then the equilibrium of* $\Omega_{\mathcal{P}}$ *is asymptotically stable.*

Proposition 4, fact (15) and equation (22) motivate us to decrease $J_{\Omega_{\mathcal{P}},\mathcal{Q}}^\star$. One way to achieve this objective is by refining $\mathcal{Q}$. The characterization of an appropriate refinement method is the focus of the next section.

B. Refinement Step

The goal of our refinement step is to split $Z_i\in\mathcal{Q}$ into cells Z_l, Z_m, thereby refining $\mathcal{Q}$ into $\mathcal{R}$ such that either $d_l^{\mathcal{R}}(p_i^\star(\mathcal{Q}),\bar{\nu}_l)$ or $d_m^{\mathcal{R}}(p_i^\star(\mathcal{Q}),\bar{\nu}_m)$ equals zero, where $p_i^\star(\mathcal{Q})$ corresponds to the optimal solution of (17)-(21) for PWL Lyapunov function $V_{\mathcal{Q}}(x)$. This goal ensures successive refinements do not increase the value of the objective function, which we instead want to decrease to zero due to Proposition 4. Furthermore, we want both Z_l and Z_m to be polyhedral cones with apices at the origin. Therefore, the refinement step must involve defining a hyperplane through the origin. Figure 1 depicts an example of such a refinement step. The existence of a hyperplane that achieves our refinement objective is characterized by the result below.

Lemma 5. *Let* $Z=\{x\in R^n\colon Ex\geq 0\}$ *where* $E\in\mathbb{R}^{m\times n}$ *and* $r(E)=n$. *Suppose* $v_1,v_2,\ldots,v_k\notin\mathrm{Int}(Z^*)$. *If there exists a row* E^j *of* E *such that* $E^j(v_i)^T>0$ *for all* $i\in\{1,\ldots,k\}$, *then there exist* $Z_1=\{x\in R^n\colon E_1x\geq 0\}$ *and* $Z_2=\{x\in R^n\colon E_2x\geq 0\}$ *such that*

i) $\{Z_1,Z_2\}\in\mathrm{Ref}(Z)$*, and*
ii) $v_1,v_2,\ldots,v_k\in\mathrm{Int}(Z_1^*)$.

Proof. We demonstrate the result by construction, which serves as a procedure for computing the refinement. Let

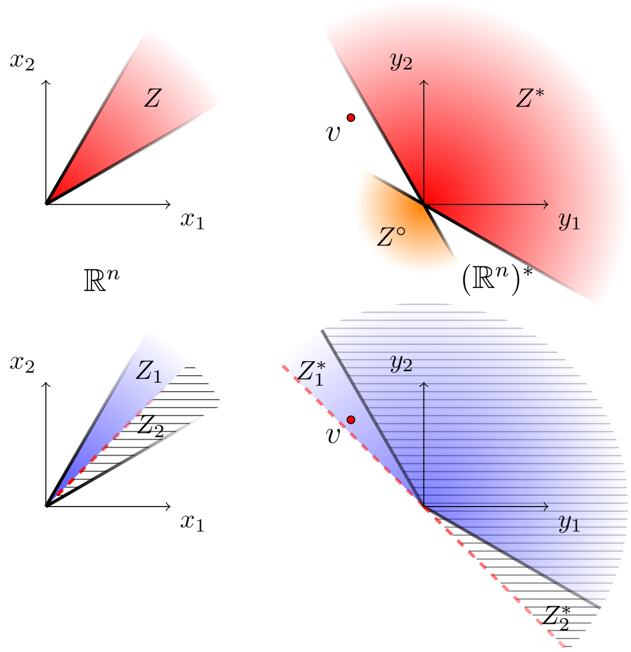

Fig. 1. [Best viewed in color] A sketch of the dual cones Z_1^* (blue) and Z_2^* (hashed) obtained by refining Z into Z_1 and Z_2. These cones are unbounded. The point v is such that $v \notin \mathrm{Int}(Z^*)$, but $v \in \mathrm{Int}(Z_1^*)$.

$v_i = \sum_{j\in\{1,2,\ldots,m\}} \mu_{ij} E^j$, where E^j is the j^{th} row of E. Without loss of generality, let E^1 be the row of E such that $\mu_{i1} > 0$ for all $i \in \{1,2,\ldots,k\}$. Define

$$\mu_1^* = (1-\epsilon) \min_{i\in\{1,2,\ldots,k\}} \mu_{i1},$$

and

$$\mu_j^* = (1+\epsilon) \max_{i\in\{1,2,\ldots,k\}} |\mu_{ij}|,$$

for $j \in \{2,\ldots,m\}$ and some ϵ such that $0 < \epsilon < 1$. Let halfplane E^{part} be given by

$$E^{part} = \mu_1^* E^1 - \sum_{j\in\{2,\ldots,m\}} \mu_j^* E^j.$$

By construction, for any v_i, $\mu_{i1} > \mu_1^*$ and $\mu_{ij} > -\mu_j^*$ for $j \in \{2,\ldots,m\}$. Therefore, for every $i \in \{1,\ldots,k\}$ we can rewrite v_i as

$$v_i \in \mathrm{pos}_{>0}\left(\{E^{part}\} \cup \{E^j\}_{j\in\{1,2,\ldots,m\}}\right). \tag{23}$$

A consequence of (23) is that if we define

$$Z_1 = \{x \in \mathbb{R}^n \colon E_1 x \geq 0\}, \text{ and} \tag{24}$$
$$Z_2 = \{x \in \mathbb{R}^n \colon E_2 x \geq 0\}, \tag{25}$$

where

$$E_1 = \begin{bmatrix} E \\ E^{new} \end{bmatrix}, \text{ and } E_2 = \begin{bmatrix} E \\ -E^{new} \end{bmatrix}, \tag{26}$$

then

i) $\{Z_1, Z_2\} \in \mathrm{Ref}(Z)$, and
ii) $v_1, v_2, \ldots, v_k \in \mathrm{Int}(Z_1^*)$.

□

If $-p_i^T A_{jk} \in \mathrm{Int}(Z^*)$ for all multi-indices $(i,j,k) \in I_{dec}(\mathcal{Q})$ associated with cell $Z_i \in \mathcal{Q}$, then there exists $\bar{\nu} > 0$ such that $d_i^{\mathcal{Q}}(p,\bar{\nu}) = 0$. However, the constant $\bar{\nu}_i$ given in (14) may be such that $d_i^{\mathcal{Q}}(p,\bar{\nu}_i) \neq 0$. Therefore, to refine a cell $Z_i \in \mathcal{Q}$ into Z_j, Z_k such that either $d_j^{\mathcal{R}}(p_i^\star,\bar{\nu}_j)$ or $d_k^{\mathcal{R}}(p_i^\star,\bar{\nu}_k)$ equals zero as mentioned earlier, we must have that

$$\left(-(p_i^\star)^T A_{jk} - \mathbf{1}^T F_j\right) \notin Z_i^\circ \tag{27}$$

for all $k \in I_{A_j}$, where $j = \pi_{\mathcal{Q},\mathcal{P}}(i)$. We refer to the method of refinement in Lemma 5 as a *d-split* refinement, since it affects the quantity $d_i^{\mathcal{Q}}$ in (13) as shown in the result below. The partition it induces is called a d-split partition.

Lemma 6. *Let $\Omega_{\mathcal{P}}$ be a piece-wise affine dynamical system and $V_{\mathcal{Q}}$ be a piecewise linear Lyapunov function where $\mathcal{Q} \in \mathrm{Ref}(\mathcal{P})$. Let $p_i^\star \in \mathbb{R}^n$ be the optimal value of the parameter p_i for cell $Z_i \in \mathcal{Q}$ corresponding to the solution of* (17)-(21) *for Lyapunov function $V_{\mathcal{P}}(x)$ such that $p_i^\star$ satisfies* (27) *but not* (12). *Let Z_j and Z_k be a d-split refinement of $Z_i \in \mathcal{Q}$ as in Lemma 5, creating a partition $\mathcal{R} = \{Z_j, Z_k\} \cup \{Z_l\}_{l\in I_{\mathcal{Q}}, l\neq i}$. Then either*

i) $d_j^{\mathcal{R}}(p_i^\star,\bar{\nu}_j) = 0$ *and* $d_k^{\mathcal{R}}(p_i^\star,\bar{\nu}_k) = d_i^{\mathcal{Q}}(p_i^\star(\mathcal{Q}),\bar{\nu}_i)$, *or*
ii) $d_j^{\mathcal{R}}(p_i^\star,\bar{\nu}_j) = d_i^{\mathcal{Q}}(p_i^\star,\bar{\nu}_i)$ *and* $d_k^{\mathcal{R}}(p_i^\star,\bar{\nu}_k) = 0$.

Proof. The fact that $d_j^{\mathcal{R}}(p_i^\star,\bar{\nu}_j) = 0$ or $d_k^{\mathcal{R}}(p_i^\star,\bar{\nu}_k) = 0$ holds by a direct application of Lemma 5 and (15). The fact that $d_j^{\mathcal{R}}(p_i^\star,\bar{\nu}_j) = d_i^{\mathcal{Q}}(p_i^\star,\bar{\nu}_i)$ or $d_k^{\mathcal{R}}(p_i^\star,\bar{\nu}_k) = d_i^{\mathcal{Q}}(p_i^\star(\mathcal{Q}),\bar{\nu}_i)$ can be shown using the construction of E^{part} in Lemma 5, which we omit due to space constraints. □

The value of a d-split refinement is that it allows us to strictly reduce the set of points for which the current PWL Lyapunov function is non-decreasing, without increasing the objective function (16). This property is characterized by the following result.

Theorem 7. *Let $\Omega_{\mathcal{P}}$, $\mathcal{Q}$, and $\mathcal{R}$ satisfy the conditions of Lemma 6. Then,*

$$J^\star_{\Omega_{\mathcal{P}},\mathcal{R}} \leq J^\star_{\Omega_{\mathcal{P}},\mathcal{Q}}. \tag{28}$$

Proof. Given a Lyapunov function $V_{\mathcal{Q}}(x)$, we can solve (17)-(21) to obtain an optimal solution $\{p_i^\star\}$, $\{\mu_i^\star\}$, $\{\nu_{ij}^\star\}$, and $\{\lambda_{ij}^\star\}$. We denote $p_i^\star$ corresponding to a Lyapunov function $V_{\mathcal{Q}}(x)$ as $p_i^\star(\mathcal{Q})$ when the partition is not clear. We have that

$$J^\star_{\Omega_{\mathcal{P}},\mathcal{Q}} = \sum_{i\in I_{\mathcal{Q}}} d_i^{\mathcal{Q}}(p_i^\star,\bar{\nu}_i). \tag{29}$$

Due to Lemma 6,

$$d_j^{\mathcal{R}}(p_j,\bar{\nu}_j) + d_k^{\mathcal{R}}(p_k,\bar{\nu}_k) = d_i^{\mathcal{Q}}(p_i^\star,\bar{\nu}_i). \tag{30}$$

In turn,

$$\sum_{j\,\in I_{\mathcal{R}}} d_j^{\mathcal{R}}(p^\star_{\pi_{\mathcal{R}\to\mathcal{Q}}(j)}(\mathcal{Q}),\bar{\nu}_j) = J^\star_{\Omega_{\mathcal{P}},\mathcal{Q}}. \tag{31}$$

Since $\mathcal{R}$ is a refinement of $\mathcal{Q}$, the optimal parameters $p_i^\star(\mathcal{Q})$ of (17)-(21) with Lyapunov function $V_{\mathcal{Q}}(x)$ can be mapped to a set of parameters $\{p_j\}_{j\in\mathcal{R}}$ that correspond to a

Algorithm 1 Verifying Asymptotic Stability By Refining PWL Lyapunov Functions

Require: $\Omega_{\mathcal{P}}$
Ensure: PWL Lyapunov function $V_{\mathcal{Q}}(x)$ that verifies the origin is asymptotically stable.
  $k \leftarrow 0$ {Loop counter}
  $\mathcal{R}_k \leftarrow \mathcal{P}$
  $J_k \leftarrow \infty$
  while $J_k > 0$ **do**
    Solve (17)-(21) with $\Omega_{\mathcal{P}}$ and $V_{\mathcal{R}_k}(x)$.
    $I_{refine} \leftarrow \{i \in I_{\mathcal{R}_k} : d_i^{\mathcal{Q}}(p_i^\star, \bar{\nu}_i) \neq 0\}$
    $J_k \leftarrow J^\star_{\Omega_{\mathcal{P}},\mathcal{R}_k}$
    for $i \in I_{refine}$ **do**
      if $p_i^\star(\mathcal{R}_k)$ satisfies (27) **then**
        Refine $Z_i \in \mathcal{R}_k$ using a d-split refinement.
      end if
    end for
    $k \leftarrow k+1$
  end while
  return $V_{\mathcal{Q}}(x)$.

feasible solution of (17)-(21) with Lyapunov function $V_{\mathcal{R}}(x)$ using the abstraction function $\pi_{\mathcal{R},\mathcal{Q}}$. Therefore

$$J^\star_{\Omega_{\mathcal{P}},\mathcal{R}} \leq \sum_{j \in I_{\mathcal{R}}} d_j^{\mathcal{R}}(p^\star_{\pi_{\mathcal{R},\mathcal{Q}}(j)}(\mathcal{Q}), \bar{\nu}_j) \tag{32}$$

Combining (31) and (32) completes the proof. □

Theorem 7 motivates us to use the d-split refinement in Lemma 5 to develop our refinement algorithm, given in the next section.

C. Refinement Algorithm

Our refinement algorithm involves a sequence of optimization problems of the form (17)-(21) corresponding to a sequence of partitions $\mathcal{R}_k$ where $\mathcal{R}_{k+1} \in \text{Ref}(\mathcal{R}_k)$ and $\mathcal{R}_0 = \mathcal{P}$. The refinement of $\mathcal{R}_k$ involves searching for a cell $Z_i \in \mathcal{R}_k$ for which constraint (9) is infeasible ($d(p_i^\star(\mathcal{R}_k), \bar{\nu}_i) \neq 0$) but can be split into two cells ($p_i^\star(\mathcal{R}_k)$ satisfies (27)) so that one of them will satisfy a constraint of the form (9) corresponding to that cell. This refinement step strictly reduces the set of points for which the Lyapunov function at iteration k is non-decreasing. Algorithm 1 describes this procedure. We can show the following properties.

Proposition 8. *Algorithm 1 is sound.*

Proof. This result is a consequence of Proposition 4 and Theorem 7. □

V. Examples

We present three piece-wise linear dynamical systems and demonstrate the benefits of the refinement-based search for a PWL Lyapunov function proposed in Section IV. We use MatLAB R2018b to implement all computations, using a computer with a 2.6 GHz processor and 16 GB RAM.

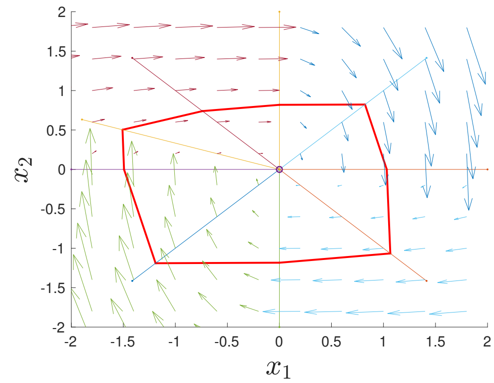

Fig. 2. Contour depicting a level set of a valid piece-wise linear Lyapunov function for Example 1 found by Algorithm 1.

Example 1 (Based on Branciky's example [9])**.** This classic example involves a state-based switched system with two modes, where the dynamics are neutrally stable in each mode. Appropriate switching leads to the origin being asymptotically stable. The dynamics are given by

$$\Sigma_1 : \dot{x} = \begin{bmatrix} 0 & 1 \\ -4 & 0 \end{bmatrix} x, \quad X_1 = \{x \in \mathbb{R}^2 | x_1 x_2 \geq 0\}, \text{ and}$$
$$\Sigma_2 : \dot{x} = \begin{bmatrix} 0 & 1 \\ -0.1 & 0 \end{bmatrix} x, \quad X_2 = \{x \in \mathbb{R}^2 | x_1 x_2 < 0\}.$$

Algorithm 1 finds a valid Lyapunov function with 9 cells in 2.386 seconds. Figure 2 depicts a level set of this Lyapunov function by the solid red line.

Example 2 (State-based 3D Switched System)**.** We convert the two dimensional system in Example 1 into a three dimensional system by adding a third variable with stable linear dynamics decoupled from the first two. The dynamical system $\Omega_{\mathcal{P}}$ is given by

$$\Sigma_1 : \dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ -4 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} x, \quad X_1 = \{x \in \mathbb{R}^3 | x_1 x_2 \geq 0\}, \text{ and}$$
$$\Sigma_2 : \dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ -0.1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} x, \quad X_2 = \{x \in \mathbb{R}^3 | x_1 x_2 < 0\}.$$

Algorithm 1 finds a valid Lyapunov function with 62 cells in 59.667 seconds. Figure 3 depicts a level set of this Lyapunov function through the edges of its facets.

Example 3 (Prajna *et al.* [10])**.** Arbitrary Switching, The system arbitrarily switches between two modes

$$\dot{x} = A_1 x = \begin{bmatrix} -5 & -4 \\ -1 & -2 \end{bmatrix} x, \text{ and } \dot{x} = A_2 x = \begin{bmatrix} -2 & -4 \\ 20 & -2 \end{bmatrix} x.$$

The differential inclusion

$$\dot{x} \in \text{conv}\,(A_1 x, A_2 x) \tag{33}$$

captures this arbitrarily switching between the two modes.

Prajna *et al.* [10] use sums-of-squares (SOS) optimization to find a polynomial Lyapunov function of degree 6 for this

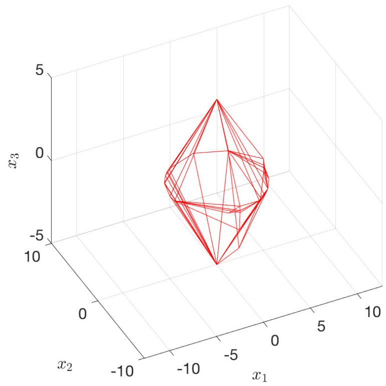

Fig. 3. Edges of a level set of a valid piece-wise linear Lyapunov function for Example 2 found by Algorithm 1.

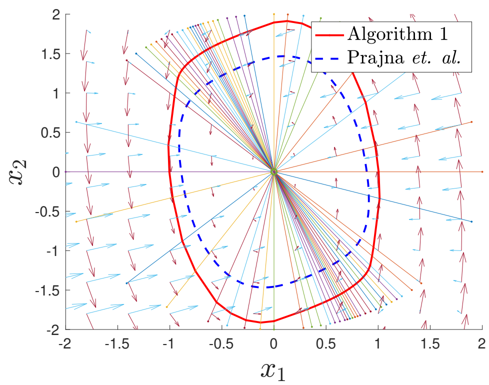

Fig. 4. The red contour depicts a level set of a valid piece-wise linear Lyapunov function for Example 3 found by Algorithm 1. The dashed blue line is a level set of 6^{th} degree polynomial reported in [10].

system. The time taken is not reported. SOS-based methods typically scale poorly with degree. The dashed blue line in Figure 4 depicts a level set of this polynomial.

Algorithm 1 finds a valid Lyapunov function with 66 cells in 5.499 seconds. Figure 4 depicts a level set of this polynomial by the solid red line. Note that it approximates the contour (dashed blue line) obtained in [10].

VI. Discussion And Future Work

We have presented an algorithm for finding PWL Lyapunov functions for verifying the asymptotic stability of piece-wise linear differential inclusions. The algorithm uses a sequential optimization approach, where iteration involves refining cells in a precise and automated manner. If the PWL Lyapunov function can be refined, then the set of points in the state space at which the Lyapunov function is non-decreasing along trajectories strictly decreases. We presented some examples to demonstrate the utility of this algorithm.

Limitations: While the algorithm succeeds in finding PWL Lyapunov functions in the presented examples, the time taken may scale poorly with dimension. Another issue is that if the size of the partition increases too quickly relative to any decrease in the objective function, the computational time can increase significantly.

Our algorithm guarantees that the objective function value is non-increasing at any iteration, but does not guarantee strict decrease. This lack prevents deciding if the algorithm is complete or not.

Future Work: Future work will involve developing methods to improve the refinement procedure by identifying refinement steps that may not decrease the objective function and avoiding refinement of those cells. This approach may also enable showing completeness of the existing algorithm or modifying it to obtain a complete algorithm.

VII. Acknowledgement

The authors would like to thank Dr. Pavithra Prabhakar for helpful discussions on automatic Lyapunov-based verification using refinements, and feedback on this paper.

References

[1] F. Blanchini, "Nonquadratic lyapunov functions for robust control," *Automatica*, vol. 31, no. 3, pp. 451 – 461, 1995.
[2] M. Johansson, "Piecewise linear control systems," Ph.D. dissertation, Lund University, 1999.
[3] P. Prabhakar and M. G. Soto, "Counterexample guided abstraction refinement for stability analysis," in *International Conference on Computer Aided Verification*, 2016, pp. 495–512.
[4] J. Oehlerking, H. Burchardt, and O. Theel, "Fully automated stability verification for piecewise affine systems," in *Hybrid Systems: Computation and Control*, A. Bemporad, A. Bicchi, and G. Buttazzo, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 741–745.
[5] M. Johansson and A. Rantzer, "Computation of piecewise quadratic lyapunov functions for hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 555–559, Apr 1998.
[6] J. Cortes, "Discontinuous dynamical systems," *IEEE Control Systems Magazine*, vol. 28, no. 3, pp. 36–73, June 2008.
[7] O. L. Mangasarian, *Nonlinear Programming*, ser. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 1994.
[8] H. A. Poonawala, N. Lauffer, and U. Topcu, "Training classifiers for feedback control," in *2019 American Control Conference (ACC)*, July 2019, pp. 4961–4967. [Online]. Available: https://arxiv.org/abs/1903.03688
[9] M. S. Branicky, "Multiple lyapunov functions and other analysis tools for switched and hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 475–482, April 1998.
[10] S. Prajna and A. Papachristodoulou, "Analysis of switched and hybrid systems - beyond piecewise quadratic methods," in *Proceedings of the 2003 American Control Conference, 2003.*, vol. 4, June 2003, pp. 2779–2784 vol.4.