# A Data-Driven Approach for Certifying Asymptotic Stability and Cost Evaluation for Hybrid Systems

Carlos A. Montenegro G.
University of California, Santa Cruz,
Santa Cruz, CA 95064, USA
camonten@ucsc.edu

Santiago J. Leudo
University of California, Santa Cruz,
Santa Cruz, CA 95064, USA
sjimen28@ucsc.edu

Ricardo G. Sanfelice
University of California, Santa Cruz,
Santa Cruz, CA 95064, USA
ricardo@ucsc.edu

## ABSTRACT

In this paper, we propose a learning-based algorithm for hybrid systems with a twofold purpose: first, to design Lyapunov functions and, second, to upper bound the cost of solutions to the system. Via enforcing conditions at finitely many points of a set of interest and leveraging regularity properties of the maps defining the dynamics of the system and the stage costs associated to solutions, we extend the conditions to the entire set of interest. The method employs neural networks to learn a Lyapunov function and a value-like function to guarantee the extended pointwise conditions at all points in the set of interest and thus, guarantee practical asymptotic stability of a set or provide an upper bound on the cost of solutions, respectively. The approach is illustrated in a hybrid oscillator system.

## CCS CONCEPTS

• **Theory of computation** → **Mathematical optimization**; • **Computer systems organization** → **Robotic autonomy**; • **Computing methodologies** → **Machine learning algorithms**.

## KEYWORDS

Hybrid Systems, Data-Driven, Lyapunov Stability, Cost Evaluation

**ACM Reference Format:**

Carlos A. Montenegro G., Santiago J. Leudo, and Ricardo G. Sanfelice. 2024. A Data-Driven Approach for Certifying Asymptotic Stability and Cost Evaluation for Hybrid Systems. In *27th ACM International Conference on Hybrid Systems: Computation and Control (HSCC '24), May 14–16, 2024, Hong Kong, Hong Kong*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3641513.3650122


*HSCC '24, May 14–16, 2024, Hong Kong, Hong Kong*

ACM ISBN 979-8-4007-0522-9/24/05.
https://doi.org/10.1145/3641513.3650122

## 1 INTRODUCTION

Results on sufficient conditions to guarantee the satisfaction of dynamical properties, such as stability, safety, and optimality rely on pointwise conditions involving certificates, e.g., Lyapunov functions, barrier functions, and value functions. Though such conditions are sufficient to characterize the behavior of a system, synthesizing the certificate to satisfy the required conditions is an open research area, especially when the system dynamics are nonlinear.

On the one hand, different approaches have been considered to synthesize Lyapunov functions for continuous-time systems with specific dynamics, e.g., sum of squares for polynomial systems [10, 21]. In [24], the authors structure the Lyapunov candidate function such that it inherently yields a provable stability certificate. In [16], the authors propose a framework for learning dynamical systems with stable inference dynamics [2, 17]. In [13], a neural network structure is proposed to provably overcome the curse of dimensionality in the synthesis of Lyapunov functions for continuous-time systems with nonlinear dynamics, whereas in [20] a quadratic Lyapunov function is optimized to provide stability guarantees. Furthermore, in [1], a counterexample-guided approach is proposed using finitely many points, as well as an approach to extend the results to a subset of the state space using satisfiability modulo theories (SMT). A similar approach is proposed in [5], where the authors opted for a mixed-integer linear program (MILP) rather than SMT.

On the other hand, the interconnection of physical systems with computational and communication devices, such as analog-to-digital converters, sample-and-hold devices, quantizers, or coder/decoders, etc., and the presence of discrete behavior such as timers that expire, resets, and impacts, give rise to dynamical systems with both continuous and discrete behavior, namely, *hybrid systems*. Such hybrid dynamics impose additional challenges to the construction of certificates to guaranteeing a desired dynamical property. In recent works, synthesizing Lyapunov function using LMI solvers inside a counter-example guided inductive system framework is shown to be feasible for switched systems [22]. In [6], the authors propose a mixed-integer linear program (MILP) to learn a Lyapunov function for piecewise linear systems. In [27], an approach to learn a Lyapunov function given a parametric form with unknown coefficients, based on a system of linear inequality constraints is proposed. Though impactful, these approaches are not general enough to cover the behavior exhibited by hybrid systems.

To close this gap, in this work, for the broad class of hybrid systems in [11], we propose methods for neural network-based synthesis of certificates for asymptotic stability and optimality. Specifically, we present results for the synthesis of Lyapunov functions and for the construction of upper bounds on the cost associated to a solution to a hybrid system. The hybrid systems modeling framework in [11] is rich enough to cover switched systems, impulsive systems, algebraic differential equations, and hybrid automata. The main contribution of our paper is summarized as follows:

- We provide results to extend the satisfaction of stability and cost upper bound point-wise conditions from finitely many conveniently selected points to all of the points in a given compact set.
- We present an algorithm to synthesize a Lyapunov function that provable guarantees asymptotic stability of a set of interest for a hybrid system using finitely many points, via training of a neural network as an optimization program.
- We present an algorithm to synthesize an upper bound on the cost associated to a solution to a hybrid system using finitely many points, via training of a neural network as an optimization program.

To the best of our knowledge, this paper presents the first algorithm able to synthesize Lyapunov functions and to find upper bounds on the cost of solutions for hybrid systems modeled as in [11], by only relying on finitely many samples from the state space. The most related work we are aware of is [18], where the authors propose the use of a neural network to learn a control barrier function and guarantee safety of a set for hybrid systems.

The remainder of the paper is organized as follows. In Section 2, we present preliminary material. In Section 3, we present the data-driven design of Lyapunov functions for hybrid systems. Proposition 3.6 and Theorem 3.8 provide the main results of this section, focusing on how to extend from finitely many samples to all of the points in a given compact set of interest so as to guarantee practical asymptotic stability. Sufficient conditions to find an upper bound on the cost of solutions to autonomous hybrid systems are presented in Section 4, along with a data-driven algorithm to construct cost upper bounds for hybrid systems. An example illustrating the approach is presented in Section 5. Due to space limitations, the proofs of most results have been omitted and will be published elsewhere.

**Notation.** Given two vectors, $x$ and $y$, we use the equivalent notation $(x, y) = [x^\top y^\top]^\top$, and $\langle x, y\rangle$ denotes the Euclidean inner product. We denote by $x^\top$ the transpose of $x$ and by $|x|$ a vector (e.g., Euclidean) norm of $x$. The symbol $\mathbb{N}$ denotes the set of natural numbers, including zero. The symbol $\mathbb{R}$ denotes the set of real numbers, and $\mathbb{R}_{\geq 0}$ denotes the set of nonnegative reals. Given a vector $x$ and a nonempty set $\mathcal{A}$, the distance from $x$ to $\mathcal{A}$ is defined as $|x|_{\mathcal{A}} = \inf_{y\in\mathcal{A}} |x - y|$. We represent by $\mathbb{B}$ the closed Euclidean unit ball and by $x + \varepsilon\mathbb{B}$ the closed ball of radius $\varepsilon$ centered at $x$ and by $\mathcal{A} + \varepsilon\mathbb{B} := \{a + b : a \in \mathcal{A}, b \in \varepsilon\mathbb{B}\}$ the Minkowski sum of $\mathcal{A}$ and $\varepsilon\mathbb{B}$ . We denote by $\mathrm{card}(\mathcal{A})$ the cardinality of $\mathcal{A}$, by $\mathrm{int}\,\mathcal{A}$ its interior, by $\overline{\mathcal{A}}$ its closure, by $\mathrm{vol}(\mathcal{A})$ its Lebesgue measure, and $z \sim \mathbb{U}(\mathcal{A})$ denotes that an element $z$ is sampled from the uniform probability distribution over the set $\mathcal{A}$. Given an open set $U$, the function $f : U \to \mathbb{R}^n$ is said to be of differentiability class $C^k$ if the derivatives $f', f'', \dots, f^{(k)}$ exist and are continuous on $U$. A function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a class-$\mathcal{K}$ function, also written as $\alpha \in \mathcal{K}$, if $\alpha$ is zero at zero, continuous, and strictly increasing. Similarly, a function $\beta : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to belong to class-$\mathcal{KL}$ if it is continuous, nondecreasing in its first argument, nonincreasing in its second argument and $\lim_{r\to 0^+ 0} \beta(r, s) = 0$ for each $s \in \mathbb{R}_{\geq 0}$, and $\lim_{s\to\infty} \beta(r, s) = 0$ for each $r \in \mathbb{R}_{\geq 0}$.

## 2 PRELIMINARIES

### 2.1 Modeling Hybrid Systems

This paper considers hybrid systems that will be modeled based on the framework in [11]. In such a framework, the continuous dynamics of the system are modeled by differential equations with constraints, while the discrete dynamics are modeled by difference equations with constraints. A hybrid dynamical system $\mathcal{H}$ is defined as

$$\mathcal{H} \ : \ \begin{cases} \dot{x} \ = F(x) & x \in C \\ x^+ = G(x) & x \in D \end{cases} \tag{1}$$

where $x \in \mathbb{R}^n$ is the state. The *flow map* $F : \mathbb{R}^n \to \mathbb{R}^n$ captures the continuous evolution of the system, when the state is in the *flow set* $C$. The *jump map* $G : \mathbb{R}^n \to \mathbb{R}^n$ describes the discrete evolution of the system when the state is in the *jump set* $D$.

Since solutions to the dynamical system $\mathcal{H}$ as in (1) can exhibit both continuous and discrete behavior, we use ordinary time $t$ to determine the amount of flow elapsed and a counter $j \in \mathbb{N}$ that keeps track of the number of jumps that have occurred. Based on this parametrization of time, the concept of hybrid time domain, over which solutions to $\mathcal{H}$ are defined, is as follows.

Definition 2.1. (Hybrid time domain) *A set $E \subset \mathbb{R}_{\geq 0} \times \mathbb{N}$ is a hybrid time domain if, for each $(T, J) \in E$, the set $E \cap ([0, T] \times \{0, 1, 2, \dots, J\})$ is a compact hybrid time domain, i.e., it can be written in the form $\bigcup_{j=0}^{J}([t_j, t_{j+1}] \times \{j\})$ for some finite nondecreasing sequence of times $\{t_j\}_{j=0}^{J+1}$ with $t_{J+1} = T$. Each element $(t, j) \in E$ denotes the elapsed hybrid time, which indicates that $t$ seconds of flow time and $j$ jumps have occurred.*

A hybrid signal is a function defined on a hybrid time domain. Given a hybrid signal $\phi$ and $j \in \mathbb{N}$, we define $I_\phi^j := \{t : (t, j) \in \mathrm{dom}\,\phi\}$.

Definition 2.2. (Hybrid arc) *A hybrid signal $\phi : \mathrm{dom}\,\phi \to \mathbb{R}^n$ is called a hybrid arc if, for each $j \in \mathbb{N}$, the function $t \mapsto \phi(t, j)$ is* *locally absolutely continuous on the interval* $I^j_\phi$. *A hybrid arc* $\phi$ *is said to be compact if* $\operatorname{dom}\phi$ *is compact.*

Let $\mathcal{X}$ be the set of hybrid arcs $\phi : \operatorname{dom}\phi \to \mathbb{R}^n$. A solution to the hybrid system $\mathcal{H}$ is defined as follows.

Definition 2.3. (Solution to $\mathcal{H}$) *A hybrid arc* $\phi$ *defines a solution to* $\mathcal{H}$ *in (1) if* $\phi \in \mathcal{X}$,

1) $\phi(0,0) \in \overline{C}$ *or* $\phi(0,0) \in D$,
2) *For each* $j \in \mathbb{N}$ *such that* $I^j_\phi$ *has a nonempty interior* $\operatorname{int} I^j_\phi$, *we have, for all* $t \in \operatorname{int} I^j_\phi$,
$$\phi(t,j) \in C$$
*and, for almost all* $t \in I^j_\phi$,
$$\frac{d}{dt}\phi(t,j) = F(\phi(t,j))$$
3) *For all* $(t,j) \in \operatorname{dom}\phi$ *such that* $(t,j+1) \in \operatorname{dom}\phi$,
$$\phi(t,j) \in D$$
$$\phi(t,j+1) = G(\phi(t,j))$$

A solution $\phi$ is a compact solution if $\phi$ is a compact hybrid arc; see Definition 2.2. A solution $\phi$ to $\mathcal{H}$ from $\xi \in \mathbb{R}^n$ is complete if $\operatorname{dom}\phi$ is unbounded. It is maximal if there is no solution $\varphi$ from $\xi$ such that $\phi(t,j) = \varphi(t,j)$ for all $(t,j) \in \operatorname{dom}\phi$ and $\operatorname{dom}\phi$ is a proper subset of $\operatorname{dom}\varphi$. We denote by $\hat{\mathcal{S}}_{\mathcal{H}}(M)$ the set of solutions $\phi$ to $\mathcal{H}$ such that $\phi(0,0) \in M$. The set $\mathcal{S}_{\mathcal{H}}(M) \subset \hat{\mathcal{S}}_{\mathcal{H}}(M)$ denotes all maximal solution from $M$. We define $\sup_t \operatorname{dom}\phi := \sup\{t \in \mathbb{R}_{\geq 0} : \exists j \in \mathbb{N} \text{ s.t. } (t,j) \in \operatorname{dom}\phi\}$ and $\sup_j \operatorname{dom}\phi := \sup\{j \in \mathbb{N} : \exists t \in \mathbb{R}_{\geq 0} \text{ s.t. } (t,j) \in \operatorname{dom}\phi\}$.

Well-posed hybrid systems refer to a class of hybrid systems where the solutions satisfy very useful structural properties [11]. A hybrid system $\mathcal{H}$ as in (1) is well-posed if the basic conditions hold.

Assumption 2.4. (Hybrid Basic Conditions) *Given a hybrid system* $\mathcal{H}$ *as in* (1), *i) the sets* $C$ *and* $D$ *are closed subsets of* $\mathbb{R}^n$, *and ii) the flow map* $F : \mathbb{R}^n \to \mathbb{R}^n$ *and the jump map* $G : \mathbb{R}^n \to \mathbb{R}^n$ *are continuous.*

## 2.2 Stability for Hybrid Systems

The following definition provides the notion of pre-asymptotic stability of a closed set of interest for hybrid systems as in (1).

Definition 2.5. (Pre-asymptotic stability (pAS)) *Given a hybrid system* $\mathcal{H} = (C,F,D,G)$ *as in* (1), *a nonempty set* $\mathcal{A} \subset \mathbb{R}^n$ *is said to be*

- *stable for* $\mathcal{H}$ *if, for each* $\epsilon > 0$, *there exists* $\delta > 0$ *such that*
$$|\phi(0,0)|_{\mathcal{A}} \leq \delta \quad \Longrightarrow \quad |\phi(t,j)|_{\mathcal{A}} \leq \epsilon \quad \forall (t,j) \in \operatorname{dom}\phi$$
*for each solution* $\phi$ *to* $\mathcal{H}$;
- *pre-attractive (pA) for* $\mathcal{H}$ *if there exists* $\mu > 0$ *such that every solution* $\phi$ *to* $\mathcal{H}$ *with*
$$|\phi(0,0)|_{\mathcal{A}} \leq \mu$$
*is such that* $(t,j) \mapsto |\phi(t,j)|_{\mathcal{A}}$ *is bounded and if* $\phi$ *is complete*
$$\lim_{\substack{(t,j)\in\operatorname{dom}\phi \\ t+j\to\infty}} |\phi(t,j)|_{\mathcal{A}} = 0;$$
- *pre-asymptotically stable (pAS) for* $\mathcal{H}$ *if it is stable and pre-attractive for* $\mathcal{H}$.
- *practically pre-asymptotically stable for* $\mathcal{H}$ *with respect to an associated parameter* $\varepsilon > 0$ *if, for a given* $\mu > 0$ *and a compact set* $\mathcal{U} \subset \mathbb{R}^n$, *there exists* $\beta \in \mathcal{KL}$ *such that for a small enough value of* $\varepsilon$, *each solution* $\phi$ *to* $\mathcal{H}$, *with* $\phi(0,0) \in (\overline{C} \cup D) \cap \mathcal{U}$ *and remaining in* $(\overline{C} \cup D \cup G(D)) \cap \mathcal{U}$, *satisfies*
$$|\phi(t,j)|_{\mathcal{A}} \leq \beta(|\phi(0,0)|_{\mathcal{A}}, t+j) + \mu \quad \forall (t,j) \in \operatorname{dom}\phi.$$

The conditions guaranteeing pAS of $\mathcal{A}$ for $\mathcal{H}$ without computing solutions to $\mathcal{H}$ rely on Lyapunov functions.

Definition 2.6. (Lyapunov function candidate [26, Definition 3.17]) *Given the sets* $\mathcal{U}, \mathcal{A} \subset \mathbb{R}^n$, *the function* $V : \operatorname{dom} V \to \mathbb{R}$ *defines a Lyapunov function candidate on* $\mathcal{U}$ *with respect to* $\mathcal{A}$ *for* $\mathcal{H}$ *if the following conditions hold:*

1) $(\overline{C} \cup D \cup G(D)) \cap \mathcal{U} \subset \operatorname{dom} V$;
2) $\mathcal{U}$ *contains an open neighborhood of* $\mathcal{A} \cap (C \cup D \cup G(D))$;
3) $V$ *is continuous on* $\mathcal{U}$ *and locally Lipschitz on an open set containing* $\overline{C} \cap \mathcal{U}$;
4) $V$ *is positive definite*[1] *on* $(C \cup D \cup G(D)) \cap \mathcal{U}$ *with respect to* $\mathcal{A}$.

Theorem 2.7. (Sufficient Lyapunov conditions for pre-asymptotic stability [26, Theorem 3.19]) *Consider the sets* $\mathcal{U} \subset \mathbb{R}^n$, *compact* $\mathcal{A} \subset \mathbb{R}^n$, *and a function* $V : \operatorname{dom} V \to \mathbb{R}$ *defining a Lyapunov function candidate on* $\mathcal{U}$ *with respect to* $\mathcal{A}$ *for a system* $\mathcal{H}$ *as in* (1). *If* $\mathcal{H}$ *satisfies Assumption 2.4,* $V \in \mathcal{PD}(\mathcal{A})$, *and*

$$\langle \nabla V(x), F(x) \rangle < 0 \quad \forall x \in (C \cap \mathcal{U}) \setminus \mathcal{A} \tag{2a}$$
$$V(G(x)) - V(x) < 0 \quad \forall x \in (D \cap \mathcal{U}) \setminus \mathcal{A} \tag{2b}$$

*then* $\mathcal{A}$ *is pAS for* $\mathcal{H}$.

If the function $V : \operatorname{dom} V \to \mathbb{R}$ defining a Lyapunov function candidate on $\mathcal{U}$ with respect to $\mathcal{A}$ satisfies the conditions in Theorem 2.7 that guarantee pAS of a nonempty set $\mathcal{A}$ for $\mathcal{H}$, then it is said to be a Lyapunov function on $\mathcal{U}$ with respect to $\mathcal{A}$ for $\mathcal{H}$.

[1] We say that a function $g : \operatorname{dom} g \to \mathbb{R}_{\geq 0}$ is positive definite with respect to a set $K$, also written as $g \in \mathcal{PD}(K)$, if $g(x) = 0$ for any $x \in \operatorname{dom} g \cap K$ and $g(x) > 0$ for any $x \in \operatorname{dom} g \setminus K$.

# 3 ON THE DESIGN OF LYAPUNOV FUNCTIONS FOR HYBRID SYSTEMS

In this section, our main objective is to design a Lyapunov function that guarantees asymptotic stability of a set of interest for a system with dynamics $\mathcal{H}$ as in (1) via learning-based methods. Specifically, we solve an optimization program at finitely many points satisfying sufficient stability pointwise conditions. Via a strategic selection of such points, we provide sufficient conditions to guarantee that the set of interest is stable for $\mathcal{H}$ at any point of the state space.

## 3.1 Sets of Flow and Jump Data

Our data-driven approach relies on enforcing conditions on finitely many samples of a set of interest and, under appropriate assumptions, characterize the behavior of all of the points in the set. To provably extend the conditions from samples to the entire set, we use $\varepsilon$−nets, as defined next.

Definition 3.1. ($\varepsilon$-Nets) *Given $\varepsilon > 0$ and a set $X \subset \mathbb{R}^n$, the set $X_\varepsilon \subset \mathbb{R}^n$ is said to be an $\varepsilon$−net over $X$ if, for all $x \in X$, there exists $x' \in X_\varepsilon$ such that $|x - x'| \leq \varepsilon$.*

Equivalently, $X_\varepsilon$ is an $\varepsilon$-net over $X$ if and only if $X$ can be covered by balls with centers in $X_\varepsilon$ and radii $\varepsilon$. In particular,

$$X \subseteq \bigcup_{x' \in X_\varepsilon} x' + \varepsilon\mathbb{B}.$$

Proposition 3.2. (Lower bound on the cardinality of $\mathcal{F}_C$ and $\mathcal{F}_D$ [28, Proposition 4.2.12]) *Given a compact set $\mathcal{U} \subset \mathbb{R}^n$, let $\mathcal{F}_\star$ be an $\varepsilon$-net over $\star \cap \mathcal{U}$, with $\star \in \{C, D\}$. The smallest number of closed balls with centers in $\star \cap \mathcal{U}$ and radii $\varepsilon$ whose union covers $\star \cap \mathcal{U}$ is lower bounded by*

$$\operatorname{card}(\mathcal{F}_\star) \geq \frac{1}{\varepsilon^n} \frac{vol(\star \cap \mathcal{U})}{vol(\mathbb{B})}. \tag{3}$$

The proof can be derived using [28, Proposition 4.2.12]. To construct an $\varepsilon$−net over $\star \cap \mathcal{U}$, a simple randomized algorithm that repeatedly uniformly samples $\star \cap \mathcal{U}$ works with high probability [28, Lemma 4.2.6]. Therefore, as long as we can efficiently sample from $\star \cap \mathcal{U}$ this is a feasible approach. Alternatively, following the lines of [3], a gridding approach can be considered.

If an $\varepsilon$−net can cover a set of interest, the conditions enforced at the centers of every ball (samples) can be extended, under appropriate assumptions, to every point in the set. We elaborate on this in Section 3.3.

Consider a system with dynamics $\mathcal{H}$ as in (1), described by $(C, F, D, G)$, a compact set $\mathcal{A} \subset \mathbb{R}^n$ that we seek to render asymptotically stable for $\mathcal{H}$, and a set $\mathcal{U} \subset \mathbb{R}^n$ that contains an open neighborhood of $\mathcal{A} \cap (C \cup D)$. For given $\varepsilon > 0$, the set of flow data $\mathcal{F}_C$, and the set of jump data $\mathcal{F}_D$, are $\varepsilon$−nets over $C \cap \mathcal{U}$ and $D \cap \mathcal{U}$, respectively, as in Definition 3.1, and are defined as[2]

$$\mathcal{F}_C := \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n_C)} \overset{\text{i.i.d}}{\sim} \mathbb{U}(C \cap \mathcal{U})\right\}, \tag{4a}$$

$$\mathcal{F}_D := \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n_D)} \overset{\text{i.i.d}}{\sim} \mathbb{U}(D \cap \mathcal{U})\right\}, \tag{4b}$$

which are collections of finitely many independent and identically distributed (i.i.d) samples from the corresponding set, namely, $C \cap \mathcal{U}$ and $D \cap \mathcal{U}$, respectively.

## 3.2 Computing a Sampled-Based Lyapunov Function via Learning

With the aim of guaranteeing asymptotic stability of the set $\mathcal{A}$ via learning a Lyapunov function for $\mathcal{H}$ on $\mathcal{U}$ with respect to $\mathcal{A}$ from sampled data, under Assumption 2.4, we propose an optimization program with conditions (2a) and (2b) as constraints, enforced at the points that define the sets of flow data $\mathcal{F}_C$ and of jump data $\mathcal{F}_D$, respectively.

We model the Lyapunov function candidate –see Section 2.2– as a neural network (NN). NNs are adaptive basis functions regressors, namely, a series of stacked generalized linear models (GLMs) [12], defined as

$$x \mapsto \widehat{V}_\theta(x) = \left(z^{(d)} \circ \cdots \circ z^{(1)}\right)(x), \tag{5}$$

where $d \in \mathbb{N}$ denotes the depth of the neural network (number of layers), and $w \mapsto z^{(m)}(w)$, with $m \in \{1, 2, \ldots, d\}$, describes the $m$-th hidden network layer, defined as

$$w \mapsto z^{(m)}(w) := \left(z_1^{(m)}(w), z_2^{(m)}(w), \cdots, z_{q_m}^{(m)}(w)\right), \tag{6}$$

with dimension $q_m \in \mathbb{N}$. Given a nonlinear activation function[3] $\varphi : \mathbb{R} \to \mathbb{R}$, the neurons are defined as

$$w \mapsto z_i^{(m)}(w) = \varphi\left(\left\langle \theta_i^{(m)}, w \right\rangle\right), \qquad \begin{array}{l} \forall i \in \{1, 2, \ldots, q_m\}, \\ \forall m \in \{1, 2, \ldots, d\} \end{array} \tag{7}$$

where $\theta_i^{(m)} \in \mathbb{R}^{q_{m-1}}$ are design parameters, with $q_0 = n$. The dimension of the resulting network parameter vector

$$\theta = \left(\theta_1^{(1)}, \cdots, \theta_{q_1}^{(1)}, \theta_1^{(2)}, \cdots, \theta_{q_2}^{(2)}, \cdots, \theta_1^{(d)}, \cdots, \theta_{q_d}^{(d)}\right) \in \mathbb{R}^r, \tag{8}$$

which is the stack of vectors $\theta_i^{(m)}$, satisfies

$$r = \sum_{m=1}^{d} q_{m-1} q_m.$$

The design parameters $\theta \in \mathbb{R}^r$ are initialized at random values. Then, $\theta$ is updated based on the datasets $\mathcal{F}_C$ and $\mathcal{F}_D$ such that the candidate function $\widehat{V}_\theta$ satisfies the desired properties encoded in the optimization program (this process is known as learning). The final parameters are referred to as *learned parameters*.

[2] Note that each $x^{(i)} \in \mathcal{F}_\star$ is a point sampled from the uniform distribution over $\star \cap \mathcal{U}$, with $\star \in \{C, D\}$, not necessarily related to a particular solution $\phi$ to $\mathcal{H}$.

[3] The activation function computes the node's output by evaluating its inputs alongside their corresponding weights.

Assumption 3.3. *Given the sets $C, D, \mathcal{U} \subset \mathbb{R}^n$, compact $\mathcal{A} \subset \mathbb{R}^n$ with nonempty $\mathcal{A} \cap \mathcal{U} \cap (C \cup D)$, a network parameter vector $\theta$ as in* (8)*, and the candidate function $\widehat{V}_\theta$, suppose that $\theta$ is such that there exist $\alpha_1, \alpha_2 \in \mathcal{K}$ satisfying*

$$\alpha_1(|x|_{\mathcal{A}}) \leq \widehat{V}_\theta(x) \leq \alpha_2(|x|_{\mathcal{A}}) \quad \forall x \in (C \cup D) \cap \mathcal{U}. \tag{9}$$

Notice that if Assumption 3.3 holds, then the function $\widehat{V}_\theta$ is positive definite on $(C \cup D) \cap \mathcal{U}$ with respect to $\mathcal{A}$ for any nontrivial parameters $\theta$. We formally introduce the process of learning a Lyapunov function from data.

Proposition 3.4. (Robust Program for Stability) *Consider a hybrid system $\mathcal{H}$ as in* (1) *described by $(C, F, D, G)$, satisfying Assumption 2.4, a compact set $\mathcal{A} \subset \mathbb{R}^n$ and a set $\mathcal{U} \subset \mathbb{R}^n$ that contains an open neighborhood of $\mathcal{A} \cap (C \cup D \cup G(D))$, and a function $\widehat{V}_\theta$ as in* (5) *satisfying Assumption 3.3. If the optimization program*

$$\begin{aligned} &\min_{\theta \in \mathbb{R}^r} \quad |\theta|_2 \\ &\text{s.t.} \quad \left\langle \nabla \widehat{V}_\theta(x), F(x) \right\rangle < 0 \qquad \forall x \in (C \cap \mathcal{U}) \setminus \mathcal{A}, \\ &\qquad \widehat{V}_\theta(G(x)) - \widehat{V}_\theta(x) < 0 \qquad \forall x \in (D \cap \mathcal{U}) \setminus \mathcal{A} \end{aligned} \tag{10}$$

*is feasible, then $\widehat{V}_\theta$ is a Lyapunov function on $\mathcal{U}$ with respect to $\mathcal{A}$ for $\mathcal{H}$ and $\mathcal{A}$ is pre-asymptotically stable for $\mathcal{H}$.*

Proof. Assume (10) is feasible. Then, there exist $\theta \in \mathbb{R}^r$ defining the function $\widehat{V}_\theta$ satisfying

$$\begin{aligned} \left\langle \nabla \widehat{V}_\theta(x), F(x) \right\rangle < 0 \quad \forall x \in (C \cap \mathcal{U}) \setminus \mathcal{A}, \\ \widehat{V}_\theta(G(x)) - \widehat{V}_\theta(x) < 0 \quad \forall x \in (D \cap \mathcal{U}) \setminus \mathcal{A} \end{aligned}$$

Thus, such $\widehat{V}_\theta$ satisfies (2). Given that by design, dom $\widehat{V}_\theta := \mathbb{R}^n$, $\widehat{V}_{\theta^*}$ is continuous on $\mathcal{U}$ and locally Lipschitz on an open set containing $\overline{C} \cap \mathcal{U}$ [8], and given that, thanks to Assumption 3.3, $\widehat{V}_{\theta^*}$ is positive definite on $(C \cup D \cup G(D)) \cap \mathcal{U}$ with respect to $\mathcal{A}$, then $\widehat{V}_{\theta^*}$ is a Lyapunov function on $\mathcal{U}$ with respect to $\mathcal{A}$ for $\mathcal{H}$, as in Definition 2.6, and thanks to Theorem 2.7, we have that $\mathcal{A}$ is pAS for $\mathcal{H}$. □

As stated in the previous section, (10) requires constraints satisfaction for infinitely many points in $C \cap \mathcal{U}$, which is computationally intractable. Therefore, we compute a tractable approximation to the optimization program in (10) through a scenario[4] program in which only finitely many samples are considered. Given design parameters $\tau_C, \tau_D > 0$, and $\mu > \varepsilon > 0$:

$$\begin{aligned} &\min_{\theta \in \mathbb{R}^r} \quad |\theta|_2 \\ &\text{s.t.} \quad \left\langle \nabla \widehat{V}_\theta(x'), F(x') \right\rangle \leq -\tau_C \quad \forall x' \in \mathcal{F}_C \setminus (\mathcal{A} + \mu\mathbb{B}), \\ &\qquad \widehat{V}_\theta(G(x')) - \widehat{V}_\theta(x') \leq -\tau_D \quad \forall x' \in \mathcal{F}_C \setminus (\mathcal{A} + \mu\mathbb{B}) \end{aligned} \tag{11}$$

Notice that if we allow $\mu \leq \varepsilon$ or define the constraints in (11) at $(\mathcal{F}_C \cup \mathcal{F}_D) \setminus \mathcal{A}$, generalizing such conditions to every $\varepsilon-$ball with center at $(\mathcal{F}_C \cup \mathcal{F}_D) \setminus \mathcal{A}$ will impose undesired conditions on $\mathcal{A}$. This justifies enforcing the constraints only outside a $\mu-$ball around $\mathcal{A}$. Naturally, this does not entail a cost-free implementation, and a discussion on its implications is included after Theorem 3.10.

[4]Referring to the fact that (10) will be solved at finitely many state values [19].

## 3.3 Generalizing Lyapunov Conditions from Sampled Data

We aim to generalize the conditions enforced at the points in the flow and jump data sets to every point in $(C \cup D) \cap \mathcal{U}$. Thus, taking advantage of the fact that an $\varepsilon-$net can be constructed with centers at the points in $\mathcal{F}_C, \mathcal{F}_D$ and covering $(C \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$ and $(D \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$, respectively, the parameters $\tau_C$ and $\tau_D$ in the constraints in (11) can be conveniently chosen such that the Lipschitz continuity[5] of $\widehat{V}_{\theta^*}$ as in (5), of its gradient, and of its time derivative guarantee that the Lyapunov conditions (2a) and (2b) hold at all points in $(C \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$ and $(D \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$, respectively.

### 3.3.1 *Lipschitz Continuity of the derivative of $\widehat{V}_\theta$.*

Sufficient conditions to guarantee Lipschitz continuity of $\widehat{V}_\theta$ include Lipschitz continuity of the activation function $\varphi$ defining $\widehat{V}_\theta$, as follows.

Lemma 3.5. (Lipschitz continuity of the Lyapunov function candidate) *Consider a compact set the function $\widehat{V}_\theta$ as in* (5) *with $d$ layers and network parameter vector $\theta$. If the activation function $\varphi$ defining $\widehat{V}_\theta$ is $L_\varphi$-Lipschitz continuous, then $\widehat{V}_\theta$ is $L_{\widehat{V}_\theta}-$Lipschitz continuous.*

Lemma 3.6. (Lipschitz continuity of the gradient of $\widehat{V}_\theta$) *Consider a hybrid system $\mathcal{H}$ as in* (1)*, described by $(C, F, D, G)$, a compact set $\mathcal{U} \subset \mathbb{R}^n$, and a function $\widehat{V}_\theta$ as in* (5)*. Assume that the activation function $\varphi$ defining $\widehat{V}_\theta$ is $C^2$. Then, the gradient of $\widehat{V}_\theta$, namely $\nabla\widehat{V}_\theta$, is locally $L_{\nabla\widehat{V}_\theta}-$Lipschitz on $(C \cup D \cup G(D)) \cap \mathcal{U}$.*

Finally, we will leverage these results to prove Lipschitz continuity of the time derivative of $\widehat{V}_\theta$.

Proposition 3.7. (Lipschitz continuity of $\dot{\widehat{V}}_\theta$) *Consider the function $\widehat{V}_\theta$ as in* (5) *and a hybrid system $\mathcal{H} = (C, F, D, G)$ as in* (1)*. Assume that the flow map $F : C \to \mathbb{R}^n$ is locally $L_F$-Lipschitz on $C \cap \mathcal{U}$, and there exists $\eta_F > 0$ such that $|F(x)| \leq \eta_F$ for all $x \in C \cap \mathcal{U}$, and the conditions in Lemma 3.5 and Lemma 3.6 hold, namely the activation function $\varphi$ defining $\widehat{V}_\theta$ is $L_\varphi$-Lipschitz continuous and its gradient $\nabla\varphi$ is $L_{\nabla\varphi}-$Lipschitz continuous. Then, the function $\dot{\widehat{V}}_\theta(x) := \langle \nabla\widehat{V}_\theta(x), F(x) \rangle$ is Lipschitz continuous with constant $L_{\dot{\widehat{V}}_\theta} := L_{\nabla\widehat{V}_\theta}\eta_F + L_{\widehat{V}_\theta} L_F$.*

Proposition 3.8. (Generalized Lyapunov Conditions) *Given compact sets $\mathcal{U}, \mathcal{A} \subset \mathbb{R}^n$, consider the hybrid system $\mathcal{H} = (C, F, D, G)$ as in* (1)*, with $F$ locally $L_F-$Lipschitz on $C \cap \mathcal{U}$ and $G$ locally $L_G-$Lipschitz on $D \cap \mathcal{U}$, a Lipschitz function $\widehat{V}_\theta$ as in* (5) *with constant $L_{\widehat{V}_\theta}$ over $(C \cup D) \cap \mathcal{U}$, and $L_{\dot{\widehat{V}}_\theta}-$Lipschitz time derivative on $C \cap \mathcal{U}$. Given $\varepsilon > 0$ defining $\mathcal{F}_C$ and $\mathcal{F}_D$ as $\varepsilon-$nets over $C \cap \mathcal{U}$ and over $D \cap \mathcal{U}$,*

[5]We follow the definition of Lipschitz continuity in [26, Definition A.21], and use interchangeably the terms $L$-Lipschtiz continuous and Lipschitz continuous with constant $L$.

*respectively, if, for some* $\tau_C > L_{\dot{\widehat{V}}_\theta}\varepsilon$, $\tau_D > L_{\widehat{V}_\theta}(1+L_G)\varepsilon$, $\mu > \varepsilon$, *we have*

$$\left\langle \nabla \widehat{V}_\theta(x'), F(x')\right\rangle \le -\tau_C \quad \forall x' \in \mathcal{F}_C \setminus (\mathcal{A}+\mu\mathbb{B}), \tag{12a}$$

$$\widehat{V}_\theta(G(x')) - \widehat{V}_\theta(x') \le -\tau_D \quad \forall x' \in \mathcal{F}_D \setminus (\mathcal{A}+\mu\mathbb{B}), \tag{12b}$$

*then,*

$$\left\langle \nabla \widehat{V}_\theta(x), F(x)\right\rangle < 0 \quad \forall x \in (C\cap\mathcal{U}) \setminus (\mathcal{A}+\mu\mathbb{B}), \tag{13a}$$

$$\widehat{V}_\theta(G(x)) - \widehat{V}_\theta(x) < 0 \quad \forall x \in (D\cap\mathcal{U}) \setminus (\mathcal{A}+\mu\mathbb{B}). \tag{13b}$$

Proposition 3.8 implies that, as the chosen $\mu$ is closer to zero, the number of closed balls needed to cover $(C\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B})$ and $(D\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B})$ increases, which relaxes (12) since the right-hand sides in conditions $\tau_C > L_{\dot{\widehat{V}}_\theta}\varepsilon$ and $\tau_D > L_{\widehat{V}_\theta}(1+L_G)\varepsilon$ become smaller.

Remark 3.9. (Bootstrap Evaluation) *The conditions in Proposition 3.8 can be used to iteratively find a learning-based Lyapunov function that satisfies* (13). *Following [25], given initial parameters* $r, d$, $\tau_C, \tau_D > 0$, *and* $\theta \in \mathbb{R}^r$, *first we solve* (11) *if feasible (if not, choose new initial* $\tau_C, \tau_D$*). Then, take* $L_{\dot{\widehat{V}}_\theta}(x) \approx |\nabla\dot{\widehat{V}}_\theta(x)|$ *for all* $x \in C\cap\mathcal{U}$ *and* $L_{\Delta\widehat{V}_\theta}(x) \approx |\nabla\Delta\widehat{V}(x)|$ *for all* $x \in D\cap\mathcal{U}$, *where* $\Delta\widehat{V}_\theta(x) = \widehat{V}_\theta(G(x)) - \widehat{V}_\theta(x)$, *and verify*

$$\varepsilon - \tau_C/L_{\dot{\widehat{V}}_\theta}(x) < 0 \qquad \forall x \in \mathcal{F}_C \setminus (\mathcal{A}+\mu\mathbb{B}), \tag{14}$$

$$\varepsilon - \tau_D/L_{\Delta\widehat{V}_\theta}(x) < 0 \qquad \forall x \in \mathcal{F}_D \setminus (\mathcal{A}+\mu\mathbb{B}). \tag{15}$$

*If either* (14) *or* (15) *do not hold, choose new hyperparameters* $r, d$, $\tau_C$, *and* $\tau_D$, *then solve* (11), *and verify* (14) *and* (15) *again, iterating until a feasible set of hyperparameters is found.*

## 3.4 Learning-Based Sufficient Conditions for Stability

In this section, we show that under suitable assumptions, the solution to (11) satisfying conditions in Proposition 3.8 allows to learn a Lyapunov function for $\mathcal{H}$ on $\mathcal{U}$ with respect to $\mathcal{A}$ that satisfies sufficient conditions to guarantee practical pre-asymptotic stability of $\mathcal{A}$.

The generalization to $(C\cup D)\cap\mathcal{U}$ of the conditions that are enforced on the sets $\mathcal{F}_C, \mathcal{F}_D$ in (11) depends on an adequate construction of the $\varepsilon$−nets defined by $\mathcal{F}_C$ and $\mathcal{F}_D$. Specifically, for each $x' \in \mathcal{F}_C \setminus (\mathcal{A}+\mu\mathbb{B})$, define

$$\mathcal{E}_C(x') := \max\Big\{\varepsilon > 0 : x' + \varepsilon\mathbb{B} \subset (C\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B}), \quad \dot{\widehat{V}}_\theta(x) < 0 \quad \forall x \in x'+\varepsilon\mathbb{B}\Big\}$$

and for each $x' \in \mathcal{F}_D \setminus (\mathcal{A}+\mu\mathbb{B})$, define

$$\mathcal{E}_D(x') := \max\Big\{\varepsilon > 0 : x' + \varepsilon\mathbb{B} \subset (D\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B}), \quad \widehat{V}_\theta(G(x)) - \widehat{V}_\theta(x) < 0 \quad \forall x \in x'+\varepsilon\mathbb{B}\Big\}$$

as the radii of the biggest balls around $x'$ over which $\dot{\widehat{V}}_\theta$ and $\Delta\widehat{V}_\theta$ are negative, respectively. Then, we select $\varepsilon_C = \min_{x' \in \mathcal{F}_C\setminus(\mathcal{A}+\mu\mathbb{B})} \mathcal{E}_C(x')$, such that the set

$$\mathcal{G}_C := \bigcup_{x' \in \mathcal{F}_C\setminus(\mathcal{A}+\mu\mathbb{B})} x' + \varepsilon_C\mathbb{B}$$

is an $\varepsilon$−net over $(C\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B})$, and $\dot{\widehat{V}}_\theta(x) < 0$ for all $x \in (C\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B}) \subset \mathcal{G}_C$. Likewise, we select $\varepsilon_D = \min_{x' \in \mathcal{F}_D\setminus(\mathcal{A}+\mu\mathbb{B})} \mathcal{E}_D(x')$, such that the set

$$\mathcal{G}_D := \bigcup_{x' \in \mathcal{F}_D\setminus(\mathcal{A}+\mu\mathbb{B})} x' + \varepsilon_D\mathbb{B}$$

is an $\varepsilon$−net over $(D\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B})$, and $\Delta\widehat{V}_\theta(x) < 0$ for all $x \in (D\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B}) \subset \mathcal{G}_D$. Notice we can conveniently define a single size for the balls of both sets as $\varepsilon := \max\{\varepsilon_C, \varepsilon_D\}$. In the following result, we state sufficient conditions to guarantee that, under a proper definition of the $\varepsilon$−nets covering $(C\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B})$ and $(D\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B})$, the conditions over the derivative and the change of $\widehat{V}_\theta$ are satisfied at every point in $((C\cup D)\cap\mathcal{U})\setminus(\mathcal{A}+\mu\mathbb{B})$.

Theorem 3.10. (Practical pre-asymptotic stability) *Given compact sets* $\mathcal{U}, \mathcal{A} \subset \mathbb{R}^n$, *consider a hybrid system* $\mathcal{H}$ *as in* (1) *described by* $(C, F, D, G)$, *with* $F$ *locally* $L_F$−*Lipschitz on* $C\cap\mathcal{U}$ *and* $G$ *locally* $L_G$−*Lipschitz on* $D\cap\mathcal{U}$. *Given the sets* $\mathcal{F}_C, \mathcal{F}_D$ *as in* (4) *that are* $\varepsilon$−*nets over* $C\cap\mathcal{U}$ *and* $D\cap\mathcal{U}$, *respectively, with* $\varepsilon > 0$, *assume there exists a Lipschitz function* $\widehat{V}_\theta$ *as in* (5) *with constant* $L_{\widehat{V}_\theta}$ *over* $(D\cup C)\cap\mathcal{U}$ *and with* $L_{\dot{\widehat{V}}_\theta}$*-Lipschitz time derivative over* $C\cap\mathcal{U}$ *that satisfies Assumption 3.3,* $\widehat{V}_\theta(\mathcal{A}) = \{0\}$, *and, for* $\mu > \varepsilon$, *satisfies*

$$\left\langle \nabla \widehat{V}_\theta(x'), F(x')\right\rangle \le -\tau_C \quad \forall x' \in \mathcal{F}_C \setminus (\mathcal{A}+\mu\mathbb{B}), \tag{16a}$$

$$\widehat{V}_\theta(G(x')) - \widehat{V}_\theta(x') \le -\tau_D \quad \forall x' \in \mathcal{F}_D \setminus (\mathcal{A}+\mu\mathbb{B}), \tag{16b}$$

*for some* $\tau_C > L_{\dot{\widehat{V}}_\theta}\varepsilon$ *and* $\tau_D > L_{\widehat{V}_\theta}(1+L_G)\varepsilon$. *Then,* $\mathcal{A}$ *is practically pre-asymptotically stable for* $\mathcal{H}$ *with respect to* $\varepsilon$, *i.e., there exists* $\beta \in \mathcal{KL}$ *such that each solution* $\phi$ *to* $\mathcal{H}$ *with* $\phi(0,0) \in (\overline{C}\cup D)\cap\mathcal{U}$ *that stays in* $(\overline{C}\cup D\cup G(D))\cap\mathcal{U}$, *satisfies*

$$|\phi(t,j)|_{\mathcal{A}} \le \beta(|\phi(0,0)|_{\mathcal{A}}, t+j) + \mu \quad \forall (t,j) \in \operatorname{dom}\phi. \tag{17}$$

# 4 COST UPPER BOUND FOR HYBRID SYSTEMS

## 4.1 Sufficient Conditions for Cost Upper Bound

Following the approach in [9, 14, 15], in this section, we derive an upper bound on the cost associated to a solution to a hybrid system $\mathcal{H}$ as in (1) without computing the solution itself.

Assumption 4.1. *The flow map* $F$ *and the flow set* $C$ *are such that solutions to* $\dot{x} = F(x)$ $x \in C$ *are unique. The jump map* $G$ *is single valued.*

Given $\xi \in C\cup D$, the stage cost for flows $\mathcal{L}_C : \mathbb{R}^n \to \mathbb{R}_{\ge 0}$, the stage cost for jumps $\mathcal{L}_D : \mathbb{R}^n \to \mathbb{R}_{\ge 0}$, and the terminal cost $q : \mathbb{R}^n \to \mathbb{R}$, we define the cost associated to the solution to $\mathcal{H}$ from the initial condition $\xi$, under Assumption 4.1, as

$$\mathcal{J}(\xi) := \sum_{j=0}^{\sup_j \operatorname{dom}\phi} \int_{t_j}^{t_{j+1}} \mathcal{L}_C(\phi(t,j))dt + \sum_{j=0}^{\sup_j \operatorname{dom}\phi - 1} \mathcal{L}_D(\phi(t_{j+1},j)) + \limsup_{\substack{(t,j)\to\sup\operatorname{dom}\phi \\ (t,j)\in\operatorname{dom}\phi}} q(\phi(t,j)), \tag{18}$$

where $\{t_j\}_{j=0}^{\sup_j \operatorname{dom}\phi}$ is a nondecreasing sequence associated to the definition of the hybrid time domain of $\phi$ – see Definition 2.2.

In the next result, following [9], we present sufficient conditions to compute an upper bound on the cost associated to a solution to $\mathcal{H}$. As a difference to [9], and similar to [14], note that (18) includes a terminal cost.

Proposition 4.2. (Cost Upper Bound) *Given a hybrid system $\mathcal{H}$ as in* (1)*, stage costs $\mathcal{L}_C : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ and $\mathcal{L}_D : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$, terminal cost $q : \mathbb{R}^n \to \mathbb{R}$, and the set $\mathcal{U} \subset \mathbb{R}^n$, suppose that there exists a function $V : \operatorname{dom} V \to \mathbb{R}$, $\operatorname{dom} V \supset \overline{C} \cap \mathcal{U}$, that is continuously differentiable on an open set containing $\overline{C} \cap \mathcal{U}$, and such that*

$$\mathcal{L}_C(x) + \langle \nabla V(x), F(x)\rangle \leq 0 \qquad \forall x \in C \cap \mathcal{U}, \tag{19a}$$

$$\mathcal{L}_D(x) + V(G(x)) - V(x) \leq 0 \qquad \forall x \in D \cap \mathcal{U}. \tag{19b}$$

*Let $\phi : \operatorname{dom}\phi \to \mathbb{R}^n$ be a solution to $\mathcal{H}$ from $\xi \in (\overline{C} \cup D) \cap \mathcal{U}$ and suppose that $(t,j) \mapsto V(\phi(t,j))$ is bounded on* $\operatorname{dom}\phi$ *and*

$$\limsup_{\substack{(t,j)\to\sup\operatorname{dom}\phi \\ (t,j)\in\operatorname{dom}\phi}} V(\phi(t,j)) = \limsup_{\substack{(t,j)\to\sup\operatorname{dom}\phi \\ (t,j)\in\operatorname{dom}\phi}} q(\phi(t,j)). \tag{20}$$

*Then, it follows that*

$$\mathcal{J}(\xi) \leq V(\xi). \tag{21}$$

By building a function $V$ that satisfies the conditions in Proposition 4.2, we provide an upper bound on the cost, which is computed by evaluating $V$ at the initial condition $\xi$.

## 4.2 Sets of Flow and Jump Data for Data-Driven Cost Upper Bound

Our data-driven approach relies on enforcing conditions on finitely many points and, under appropriate assumptions, characterize the behavior of all the points in the set. To provably extend the conditions from samples to the entire set, we use $\varepsilon$−nets, as in Definition 3.1, and guarantee the conditions of interest at every $\varepsilon$−ball. If a set of interest can be covered by an $\varepsilon$−net, the conditions enforced at the centers of every ball can be extended, under appropriate assumptions, to every point in the set of interest.

Consider a hybrid system $\mathcal{H}$ as in (1), and a set $\mathcal{U} \subset \mathbb{R}^n$ such that $(C \cup D) \cap \mathcal{U}$ is nonempty. For given $\varepsilon > 0$, the set of flow data $\mathcal{F}_C$ and the set of jump data $\mathcal{F}_D$ defined as in (4), are $\varepsilon$−nets, as in Definition 3.1, which are collections of finitely many samples from the corresponding set. Using $\mathcal{F}_C$ and $\mathcal{F}_D$, in the following sections, we propose a method to find an upper bound to the cost $\mathcal{J}$ associated to a solution to $\mathcal{H}$, by using learning-based methods.

## 4.3 Sampled-Based Cost Upper Bound Conditions via Learning

With the aim of learning an upper bound on the cost $\mathcal{J}$ associated to a solution to $\mathcal{H}$ from sampled data, under Assumption 2.4, we propose an optimization program with conditions (19a) and (19b) as constraints, enforced at the points that define the set of flow data $\mathcal{F}_C$ and of jump data $\mathcal{F}_D$, respectively. By properly choosing the points of each set, we guarantee a provable extension of the aforementioned conditions to all the points of a set of interest.

We model the function $V$ in Section 4.1 as a neural network as in Section 3.2, to learn the upper bound on the cost associated to a solution to $\mathcal{H}$. Thus, $V$ is an adaptive basis functions regressor as in (5), with network parameter vector $\theta \in \mathbb{R}^r$.

First, to introduce an optimization program enforcing conditions (19a) and (19b), we consider the hybrid system $\mathcal{H}$ as in (1) described by $(C, F, D, G)$, under Assumption 2.4, and use $\widehat{V}_\theta$ as in (5) to learn $V$. Given the stage costs $\mathcal{L}_C : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ and $\mathcal{L}_D : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$, and the set $\mathcal{U} \subset \mathbb{R}^n$, define the terminal cost $q : \mathbb{R}^n \to \mathbb{R}$ as $q(x) = \widehat{V}_\theta(x)$ for each $x \in (C \cup D) \cap \mathcal{U}$. We formulate the following optimization program to compute and evaluate a cost upper bound as in (21):

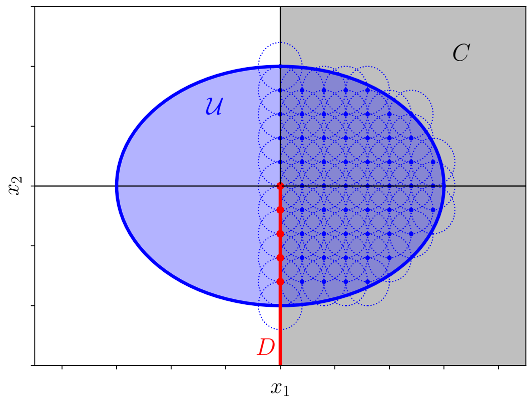
$$\begin{aligned} \min_{\theta\in\mathbb{R}^r} \quad & |\theta|_2 \\ \text{s.t.} \quad & \mathcal{L}_C(x) + \left\langle \nabla\widehat{V}_\theta(x), F(x)\right\rangle \leq 0 \qquad \forall x \in C \cap \mathcal{U}, \\ & \mathcal{L}_D(x) + \widehat{V}_\theta(G(x)) - \widehat{V}_\theta(x) \leq 0 \qquad \forall x \in D \cap \mathcal{U} \end{aligned} \tag{22}$$

Notice that the constraints in (22) are enforced at all (likely infinitely many) points in $(C \cup D) \cap \mathcal{U}$, which is computationally intractable. Therefore, we propose solving a relaxed version of (22) using a scenario[6] program, given by

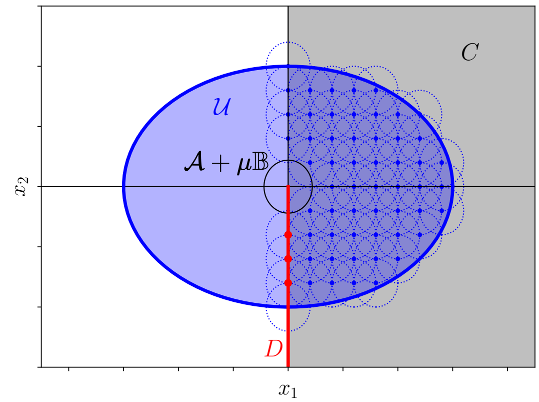
$$\begin{aligned} \min_{\theta\in\mathbb{R}^r} \quad & |\theta|_2 \\ \text{s.t.} \quad & \mathcal{L}_C(x') + \left\langle \nabla\widehat{V}_\theta(x'), F(x')\right\rangle \leq -\eta_C \qquad \forall x' \in \mathcal{F}_C, \\ & \mathcal{L}_D(x') + \widehat{V}_\theta(G(x')) - \widehat{V}_\theta(x') \leq -\eta_D \qquad \forall x' \in \mathcal{F}_D \end{aligned} \tag{23}$$

where $\mathcal{F}_C$, $\mathcal{F}_D$, and $\eta_C, \eta_D > 0$ are given. In the next section, we provide sufficient conditions to guarantee that if (23) is feasible, then we can provide an upper bound on the cost associated to a solution that starts and remains in $(\overline{C} \cup D) \cap \mathcal{U}$, under appropriate assumptions on $F$, $G$, and the elements in $\mathcal{F}_C, \mathcal{F}_D$. Such upper bound can be computed without computing solutions to $\mathcal{H}$.

[6]Referring to the fact that (22) will be solved at samples of the state space [19].




(a)




(b)

**Figure 1: Flow set $C$, jump set $D$, sampling set $\mathcal{U} \subset \mathbb{R}^2$ for the oscillator with impacts. In (a), we show the coverings over $C \cap \mathcal{U}$ and $D \cap \mathcal{U}$. We enforce conditions at the sampled points (centers of the balls), and under certain assumptions, generalize them to the set $\mathcal{U}$ to upper bound the cost associated to solutions to (28). In (b), the $\varepsilon$-nets cover the sets $(C \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$ and $(D \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$, respectively, and by extending the conditions enforced at the centers of the balls to the set $\mathcal{U}$, we guarantee practical asymptotic stability of $\mathcal{A}$ as in Section 3.2.**

## 4.4 Sufficient Conditions for Design of Learning-Based Cost Upper Bound

In this section, we show that under suitable assumptions, the solution to (23) allows to construct the function $V$ in Section 4.1 with a structure as in (5), which guarantees the existence of an upper bound on the cost associated to a solution to $\mathcal{H}$.

Similar to Section 3.2, we aim to extend the conditions enforced at the points in the flow and jump data sets, $\mathcal{F}_C$ and $\mathcal{F}_D$, to every point in $(\overline{C} \cup D) \cap \mathcal{U}$. The parameters $\eta_C$ and $\eta_D$ in the constraints in (23) can be conveniently chosen such that the Lipschitz continuity[7] of $\widehat{V}_\theta$ as in (5), its gradient, and its time derivative guarantee that the cost upper bound conditions (19a) and (19b) hold at all points in $C \cap \mathcal{U}$ and $D \cap \mathcal{U}$, respectively.

The extension to $(C \cup D) \cap \mathcal{U}$ of the conditions that are enforced on the sets $\mathcal{F}_C, \mathcal{F}_D$ in (23) depends on an adequate construction of the $\varepsilon$−nets defined by $\mathcal{F}_C$ and $\mathcal{F}_D$. Specifically, for each $x' \in \mathcal{F}_C$, define

$$\mathcal{E}_C^{\mathcal{J}}(x') \coloneqq \max \Big\{ \varepsilon > 0 : x' + \varepsilon\mathbb{B} \subset C \cap \mathcal{U}, \\ \dot{\widehat{V}}_\theta(x) + \mathcal{L}_C(x) \leq 0 \quad \forall x \in x' + \varepsilon\mathbb{B} \Big\}$$

and for each $x' \in \mathcal{F}_D$, define

$$\mathcal{E}_D^{\mathcal{J}}(x') \coloneqq \max \Big\{ \varepsilon > 0 : x' + \varepsilon\mathbb{B} \subset D \cap \mathcal{U}, \\ \widehat{V}_\theta(G(x)) - \widehat{V}_\theta(x) + \mathcal{L}_D(x) \leq 0 \quad \forall x \in x' + \varepsilon\mathbb{B} \Big\}$$

as the radii of the biggest balls around $x'$ over which $\dot{\widehat{V}}_\theta + \mathcal{L}_C$ and $\Delta\widehat{V}_\theta + \mathcal{L}_D$ are nonpositive, respectively. Then, we choose $\varepsilon = \min_{x' \in \mathcal{F}_C} \mathcal{E}_C^{\mathcal{J}}(x')$, such that the set

$$\mathcal{G}_C^{\mathcal{J}} \coloneqq \bigcup_{x' \in \mathcal{F}_C} x' + \varepsilon\mathbb{B}$$

is an $\varepsilon$−net over $C \cap \mathcal{U}$, and $\dot{\widehat{V}}_\theta(x) + \mathcal{L}_C(x) \leq 0$ for all $x \in (C \cap \mathcal{U}) \subset \mathcal{G}_C^{\mathcal{J}}$. Likewise, we choose $\varepsilon = \min_{x' \in \mathcal{F}_D} \mathcal{E}_D^{\mathcal{J}}(x')$, such that the set

$$\mathcal{G}_D^{\mathcal{J}} \coloneqq \bigcup_{x' \in \mathcal{F}_D} x' + \varepsilon\mathbb{B}$$

is an $\varepsilon$−net over $D \cap \mathcal{U}$, and $\Delta\widehat{V}_\theta(x) + \mathcal{L}_D(x) \leq 0$ for all $x \in (D \cap \mathcal{U}) \subset \mathcal{G}_D^{\mathcal{J}}$. In the following result, we state the sufficient conditions to guarantee that, under a proper definition of the $\varepsilon$−nets covering $C \cap \mathcal{U}$ and $D \cap \mathcal{U}$, the conditions over the derivative and the change of $\widehat{V}_\theta$ are satisfied at every point in in $C \cap \mathcal{U}$ and $D \cap \mathcal{U}$, respectively.

Theorem 4.3. (Data-Driven Cost Upper Bound) *Given a compact set $\mathcal{U} \subset \mathbb{R}^n$, consider a hybrid system $\mathcal{H}$ as in* (1)*, with $F$ locally $L_F$−Lipschitz on $C \cap \mathcal{U}$ and $G$ locally $L_G$−Lipschitz on $D \cap \mathcal{U}$, locally Lipschitz continuous functions $\mathcal{L}_C : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ in $C \cap \mathcal{U}$, $\mathcal{L}_D : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ in $D \cap \mathcal{U}$, with constants $L_C$ and $L_D$, respectively, defining the stage cost for flows and jumps, and the terminal cost $q : \mathbb{R}^n \to \mathbb{R}$. Given the sets $\mathcal{F}_C, \mathcal{F}_D$ as in* (4)*, suppose that these sets are $\varepsilon$−nets over $C \cap \mathcal{U}$ and $D \cap \mathcal{U}$, respectively, with $\varepsilon > 0$, and suppose that there exists a parameter vector $\theta \in \mathbb{R}^r$, defining a Lipschitz function $\widehat{V}_\theta$ as in* (5) *with constant $L_{\widehat{V}_\theta}$ over $(C \cup D) \cap \mathcal{U}$ and with $L_{\dot{\widehat{V}}_\theta}$−Lipschitz time derivative over $C \cap \mathcal{U}$, that satisfies*

$$\mathcal{L}_C(x') + \left\langle \nabla\widehat{V}_\theta(x'), F(x') \right\rangle \leq -\eta_C \quad \forall x' \in \mathcal{F}_C, \tag{24}$$

$$\mathcal{L}_D(x') + \widehat{V}_\theta(G(x')) - \widehat{V}_\theta(x') \leq -\eta_D \quad \forall x' \in \mathcal{F}_D, \tag{25}$$

[7] Guaranteed according to Lemmas 3.5, 3.6, and Proposition 3.7.

*with $\eta_C, \eta_D$ satisfying*

$$\eta_C \geq \varepsilon(L_C + L_{\hat{\widehat{V}}_\theta}) \tag{26}$$

$$\eta_D \geq \varepsilon(L_{\widehat{V}_\theta}(1 + L_G) + L_D). \tag{27}$$

*Let $\phi : \mathrm{dom}\,\phi \to \mathbb{R}^n$ be a solution to $\mathcal{H}$ from $\xi \in (\overline{C} \cup D) \cap \mathcal{U}$ and suppose that $(t, j) \mapsto \widetilde{V}_\theta(\phi(t, j))$ is bounded on $\mathrm{dom}\,\phi$ and* (20) *holds. Then,*

$$\mathcal{J}(\xi) \leq \widehat{V}_\theta(\xi).$$

Remark 4.4. (Data Driven Cost Upper Bound with Asymptotic Stability) *There are results that connect cost evaluation and asymptotic stability for hybrid systems [9]. Accordingly, under additional conditions, the learning-based cost upper bound function presented in Theorem 4.3 can be rendered as a Lyapunov function to guarantee practical pAS of a set of interest $\mathcal{A}$.*

## 5 CASE OF STUDY: LYAPUNOV FUNCTION AND COST UPPER BOUND FOR OSCILLATOR WITH IMPACTS

To illustrate our proposed algorithm[8] to design Lyapunov functions and to upper bound the cost of solutions to a hybrid system via learning, consider the linear oscillator with impacts with dynamics given by

$$\mathcal{H} \begin{cases} \dot{x} = F(x) := \begin{pmatrix} x_2 \\ -x_1 - \lambda_C x_2 \end{pmatrix} & x \in C := \left\{x \in \mathbb{R}^2 : x_1 \geq 0\right\} \\ x^+ = G(x) := \begin{pmatrix} 0 \\ -\lambda_D x_2 \end{pmatrix} & x \in D := \left\{x \in \mathbb{R}^2 : x_1 = 0, \right. \\ & \qquad\qquad \left. x_2 \leq 0\right\} \end{cases} \tag{28}$$

On the other hand, to certify stability of the set $\mathcal{A}$ for the oscillator in (28) using learning-based methods, we follow a similar approach using separate coverings for the sets $(C \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$ and $(D \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$ by finitely many round balls (see Figure 1b). By enforcing conditions at the centers of such balls that can be generalized to every point in $((C \cup D) \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$, under proper assumptions, we guarantee practical asymptotic stability of $\mathcal{A}$ for $\mathcal{H}$ with respect to $\mu$.

Finally, for the remainder of the section, we consider the following sampling set

$$\mathcal{U} = \left\{x \in \mathbb{R}^2 \mid x_1^2/h_0^2 + x_2^2/v_0^2 \leq 1\right\}$$

where $h_0, v_0 > 0$.

### 5.1 Data-Driven Lyapunov Function

The samples are strategically chosen to form $\varepsilon$-nets over $C \cap \mathcal{U} \setminus (\mathcal{A} + \mu\mathbb{B})$ and $D \cap \mathcal{U} \setminus (\mathcal{A} + \mu\mathbb{B})$, with $\varepsilon = 0.01$ and $\mu = 1.1\varepsilon$. To design a learning-based Lyapunov function, we implement a

[8] The implementation code can be found at https://github.com/HybridSystemsLab/LearningLyapunovFunction-HybridOscillator.

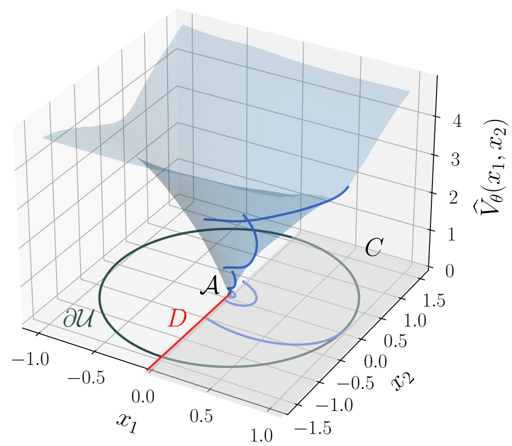



**Figure 2: Lyapunov function designed as a neural network trained on the set $\mathcal{U}$ by solving the SP in (11). A hybrid solution to $\mathcal{H}$ from $\xi = (1, 0)$ is shown (in dark blue), illustrating how $\widehat{V}_\theta$ decreases along flows and jumps.**

specific structure of a neural network that is positive definite with respect to the set $\mathcal{A} = \{0\}$ on $(C \cup D) \cap \mathcal{U}$, which is guaranteed in [23, Theorem 2], and it is shown in Figure 2. We solve the SP in (11) with $\tau_C = 0.037$ and $\tau_D = 0.049$ using JAX [4] while following the augmented Lagrangian method [7] to account for the constraints in the learning process. First, we verify that the chosen hyperparameters satisfy (14) and (15), and adjust them according to Remark 3.9 until a successful case is found.

Then, leveraging regularity conditions of the neural network and properties of the $\varepsilon$-nets of the sets of interest, following Proposition 3.8, we extend the pointwise conditions from samples to the set $\mathcal{U}$, such that the learned $\widehat{V}_\theta$ and its derivative satisfy (13). This is illustrated in Figures 3 and 4. Thanks to Theorem 3.10, we certify that $\mathcal{A} = \{0\}$ is practically pre-asymptotically stable for $\mathcal{H}$ as in (28) (see Figure 2).

### 5.2 Data-Driven Cost Upper Bound

Following a similar approach, the samples are strategically chosen to form $\varepsilon$-nets over $C \cap \mathcal{U}$ and $D \cap \mathcal{U}$ with $\varepsilon = 0.01$. We set $x \mapsto \mathcal{L}_C(x) = 0.5|x|^2$ and $x \mapsto \mathcal{L}_D(x) = 0.15|x|^2$, defining the stage costs as in the cost functional $\mathcal{J}$ in (18). To design a learning-based cost upper bound, we implement a neural network by solving the SP in (23), following the augmented Lagrangian approach to account for the constraints in the learning process. To tune the hyperparameters, namely. the number of neurons $r$, the number of layers $d$, the slack variable for flows $\eta_C$, and the slack variable for jumps $\eta_D$ , we also follow Remark 3.9 replacing (14) and (15) by

$$\varepsilon - \frac{\eta_C}{L_C(x) + L_{\hat{\widehat{V}}_\theta}(x)} \leq 0 \qquad \forall x \in \mathcal{F}_C,$$

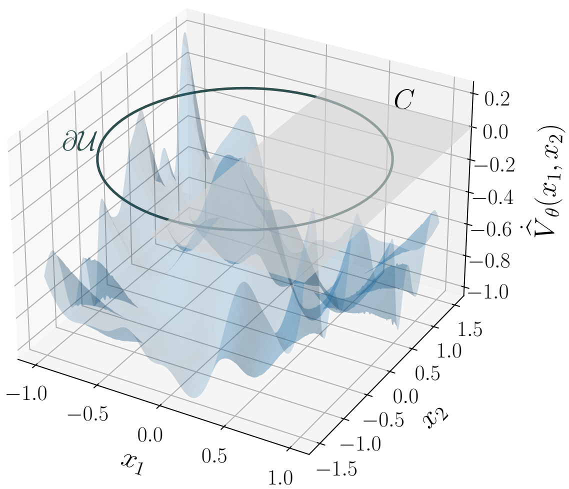



**Figure 3: Surface plot of the function $x \mapsto \dot{\widehat{V}}_\theta(x) := \langle \nabla \widehat{V}_\theta(x), F(x) \rangle$. The flow set $C$ and the boundary of the sampling set, namely $\partial\mathcal{U}$, are displayed. We show that $\dot{\widehat{V}}_\theta < 0$ at $(C \cap \mathcal{U}) \setminus (\mathcal{A} + \mu\mathbb{B})$ which is enforced via solving the SP$_\tau$ in (11) and by Proposition 3.8.**

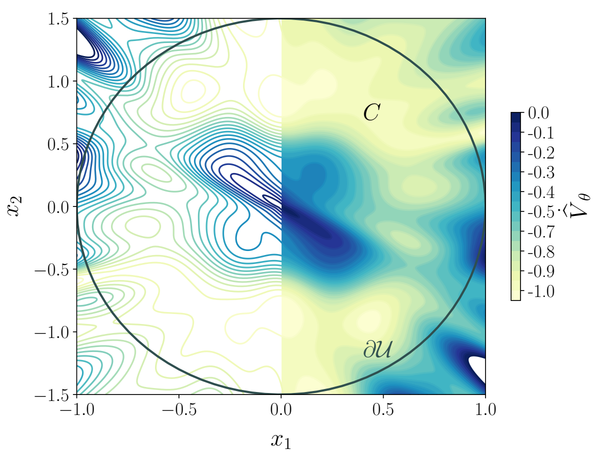



**Figure 4: Contour plot of the function $x \mapsto \dot{\widehat{V}}_\theta(x) := \left\langle \nabla \widehat{V}_\theta(x), F(x) \right\rangle$. Contours are filled only in the flow set $C$, which confirms $\dot{\widehat{V}}_\theta \leq 0$ at $C \cap \mathcal{U}$.**

$$\varepsilon - \frac{\eta_D}{L_{\widehat{V}_\theta}(x)(1 + L_G(x)) + L_D(x)} \leq 0 \qquad \forall x \in \mathcal{F}_D$$

therein, respectively, and iterating until a successful case was found, which yielded $\eta_C = 0.058$ and $\eta_D = 0.044$.

Therefore, leveraging regularity conditions of the neural network and the properties of the $\varepsilon$-nets covers at the sets of interest, following Section 4, we extend the pointwise conditions from samples

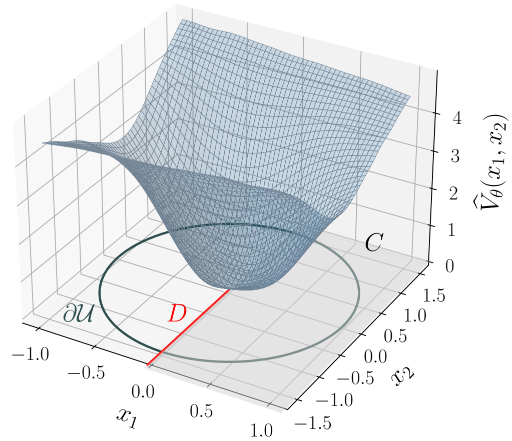



**Figure 5: Cost upper bound function designed as a neural network trained on the set $\mathcal{U}$ by solving the SP in (23) for the hybrid oscillator (28). An upper bound on the cost of evolving hybridly from any $\xi$ in $(C \cup D) \cap \mathcal{U}$ is displayed (in blue) with the neural network evaluated at such a point, which does not require computing solutions.**

to the set $\mathcal{U}$, and thanks to Theorem 4.3, we certify that $\widehat{V}_\theta$ (see Figure 5) defines an upper bound on the cost of solutions to the oscillator in (28).

## 6 CONCLUSIONS AND FUTURE WORK

In this work, we propose a data-driven algorithm to synthesize a Lyapunov function to guarantee asymptotic stability of a set of interest for a hybrid system. In addition, given a cost functional associated to solutions to a hybrid system, we propose a data-driven approach to obtain an upper bound on the cost, which does not require computing of solutions. Both approaches are based on strategically sampling points from a set of the state space and enforcing point-wise conditions at them, that under regularity properties, are generalized to every point of the set.

In future work, we will consider evaluating different data-driven methods to learn the Lyapunov and value functions and compare their scaling properties and repeatability. In addition, extending the results to hybrid inclusions will allow us to address scenarios with nondeterminism.

## AKNOWLEDGEMENTS

This research has been partially supported by NSF Grants no. CNS-2039054 and CNS-2111688, by AFOSR Grants nos. FA9550-19-1-0169, FA9550-20-1-0238, FA9550-23-1-0145, and FA9550-23-1-0313, by AFRL Grant nos. FA8651-22-1-0017 and FA8651-23-1-0004, by

ARO Grant no. W911NF-20-1-0253, and by DoD Grant no. W911NF-23-1-0158.

## REFERENCES

[1] A. Abate, D. Ahmed, M. Giacobbe, and A. Peruffo. 2020. Formal Synthesis of Lyapunov Neural Networks. *IEEE Control Systems Letters* 5, 3 (2020), 773–778.

[2] Aaron D. Ames, K. Galloway, K. Sreenath, and J. W. Grizzle. 2014. Rapidly Exponentially Stabilizing Control Lyapunov Functions and Hybrid Zero Dynamics. *IEEE Trans. Automat. Control* 59, 4 (2014), 876–891.

[3] Felix Berkenkamp, Riccardo Moriconi, Angela P. Schoellig, and Andreas Krause. 2016. Safe Learning of Regions of Attraction for Uncertain, Nonlinear Systems with Gaussian Processes. In *Proceedings of the 55th Conference on Decision and Control*. 4661–4666. https://doi.org/10.1109/CDC.2016.7798979

[4] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. 2018. *JAX: composable transformations of Python+NumPy programs*. http://github.com/google/jax

[5] H. Dai, B. Landry, Marco Pavone, and Russ Tedrake. 2020. Counter-example guided synthesis of neural network lyapunov functions for piecewise linear systems. In *Proceedings of the 59th Conference on Decision and Control*. Jeju, Korea (South), 1274–1281.

[6] Hongkai Dai, Benoit Landry, Marco Pavone, and Russ Tedrake. 2020. Counter-Example Guided Synthesis of Neural Network Lyapunov Functions for Piecewise Linear Systems. In *Proceeding of the 59th Conference on Decision and Control*. Jeju, Korea (South), 1274–1281.

[7] Franck Djeumou, Cyrus Neary, Eric Goubault, Sylvie Putot, and Ufuk Topcu. 2022. Neural Networks with Physics-Informed Architectures and Constraints for Dynamical Systems Modeling. In *Learning for Dynamics and Control Conference*. Proceedings of Machine Learning Research, 263–277.

[8] Mahyar Fazlyab, Alexander Robey, Hamed Hassani, Manfred Morari, and George Pappas. 2019. Efficient and accurate estimation of lipschitz constants for deep neural networks. *Advances in Neural Information Processing Systems* 32 (2019).

[9] Francesco Ferrante and Ricardo G. Sanfelice. 2018. Cost Evaluation for Hybrid Inclusions: A Lyapunov Approach. In *Proceedings of the 57th Conference on Decision and Control*. Miami Beach, FL, USA, 855–860.

[10] Peter Giesl and Sigurdur Hafstein. 2015. Review on computational methods for Lyapunov functions. *Discrete and Continuous Dynamical Systems-B* 20, 8 (2015), 2291–2331.

[11] Rafal Goebel, Ricardo G Sanfelice, and Andrew R Teel. 2012. *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Press, New Jersey.

[12] Peter J Green and Bernard W Silverman. 1994. *Nonparametric Regression and Generalized Linear Models*. Vol. 58. Chapman and Hall.

[13] Lars Grüne. 2021. Computing Lyapunov Functions using Deep Neural Networks. *Journal of Computational Dynamics* 8, 2 (2021), 131–152.

[14] Santiago Jimenez Leudo and Ricardo G Sanfelice. 2022. Sufficient conditions for optimality and asymptotic stability in two-player zero-sum hybrid games. In *Proceedings of the 25th International Conference on Hybrid Systems: Computation and Control*. 1–11.

[15] Santiago Jimenez Leudo and Ricardo G Sanfelice. 2022. Sufficient Conditions for Optimality in Finite-Horizon Two-Player Zero-Sum Hybrid Games. In *Proceedings of the 2022 61th IEEE Conference on Decision and Control (CDC)*. IEEE, 3268–3273. https://doi.org/10.1109/CDC51059.2022.9992905

[16] Ivan Dario Jimenez Rodriguez, Aaron D. Ames, and Yisong Yue. 2022. LyaNet: A Lyapunov Framework for Training Neural ODEs. In *Proceedings of the International Conference on Machine Learning*. 18687–18703.

[17] Hassan K Khalil. 2002. *Nonlinear systems* (3rd ed.). Prentice Hall.

[18] Lars Lindemann, Haimin Hu, Alexander Robey, Hanwen Zhang, Dimos Dimarogonas, Stephen Tu, and Nikolai Matni. 2021. Learning Hybrid Control Barrier Functions from Data. In *Conference on Robot Learning*. Proceedings of Machine Learning Research, 1351–1370.

[19] Peyman Mohajerin Esfahani, Tobias Sutter, and John Lygeros. 2015. Performance Bounds for the Scenario Approach and an Extension to a Class of Non-Convex Programs. *IEEE Trans. Automat. Control* 60, 1 (1 2015), 46–58. https://doi.org/10.1109/TAC.2014.2330702

[20] S. Mohammad Khansari-Zadeh and Aude Billard. 2014. Learning Control Lyapunov Function to Ensure Stability of Dynamical System-based Robot Reaching Motions. *Robotics and Autonomous Systems* 62, 6 (2014), 752–765. https://doi.org/10.1016/j.robot.2014.03.001

[21] Antonis Papachristodoulou and Stephen Prajna. 2002. On the construction of Lyapunov functions using the sum of squares decomposition. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, Vol. 3. IEEE, 3482–3487.

[22] Hadi Ravanbakhsh and Sriram Sankaranarayanan. 2015. Counter-example guided synthesis of control Lyapunov functions for switched systems. In *2015 54th IEEE conference on decision and control (CDC)*. IEEE, 4232–4239.

[23] Spencer M Richards, Felix Berkenkamp, and Andreas Krause. 2018. The Lyapunov Neural Network: Adaptive Stability Certification for Safe Learning of Dynamical Systems. In *Conference on Robot Learning*. Proceeding of Machine Learning Research, 466–476.

[24] Spencer M. Richards, Felix Berkenkamp, and Andreas Krause. 2018. The Lyapunov Neural Network: Adaptive Stability Certification for Safe Learning of Dynamical Systems. In *Proceedings of Machine Learning Research*. 466–476.

[25] Alexander Robey, Haimin Hu, Lars Lindemann, Hanwen Zhang, Dimos V. Dimarogonas, Stephen Tu, and Nikolai Matni. 2020. Learning Control Barrier Functions from Expert Demonstrations. In *Proceedings of the 59th Conference on Decision and Control*. Jeju, Korea (South), 3717–3724.

[26] Ricardo G Sanfelice. 2021. *Hybrid Feedback Control*. Princeton University Press, New Jersey.

[27] Sriram Sankaranarayanan, Xin Chen, et al. 2013. Lyapunov Function Synthesis using Handelman Representations. *IFAC Proceedings Volumes* 46, 23 (2013), 576–581.

[28] Roman Vershynin. 2018. *High-Dimensional Probability : An Introduction with Applications in Data Science*. Cambridge University Press.