



**HAL**  
open science

# Robustly Learning Regions of Attraction from Fixed Data

Matteo Tacchi, Yingzhao Lian, Colin N. Jones

► **To cite this version:**

Matteo Tacchi, Yingzhao Lian, Colin N. Jones. Robustly Learning Regions of Attraction from Fixed Data. 2024. hal-04101941v2

**HAL Id: hal-04101941**

**<https://hal.univ-grenoble-alpes.fr/hal-04101941v2>**

Preprint submitted on 11 Sep 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Robustly Learning Regions of Attraction from Fixed Data

Matteo Tacchi<sup>1</sup>, Yingzhao Lian<sup>2</sup>, Colin Jones<sup>2</sup>

September 11, 2024

## Abstract

While stability analysis is a mainstay for control science, especially computing regions of attraction of equilibrium points, until recently most stability analysis tools always required explicit knowledge of the model or a high-fidelity simulator representing the system at hand. In this work, a new data-driven Lyapunov analysis framework is proposed. Without using the model or its simulator, the proposed approach can learn a piece-wise affine Lyapunov function with a finite and fixed off-line dataset. The learnt Lyapunov function is robust to any dynamics that are consistent with the off-line dataset, and its computation is based on second order cone programming. Along with the development of the proposed scheme, a slight generalization of classical Lyapunov stability criteria is derived, enabling an iterative inference algorithm to augment the region of attraction.

## Keywords

Stability of nonlinear systems, Optimization, Uncertain systems, Machine learning, Robust control.

## Acknowledgments

This work was supported by the Swiss National Science Foundation under the “NCCR Automation” grant number 51NF40.180545.

The work of M. Tacchi was also supported by the French company RTE, under the RTE-EPFL partnership n°2022-0225.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Learning regions of attraction from data</b>	<b>5</b>
2.1	Unknown dynamic system, fixed data . . . . .	5
2.2	Some set-theoretical considerations . . . . .	7
<b>3</b>	<b>Piecewise affine set membership for Lyapunov inference</b>	<b>7</b>
3.1	Lyapunov inference . . . . .	7
3.2	Piecewise affine Lyapunov function . . . . .	9

---

<sup>1</sup>Univ. Grenoble Alpes, CNRS, Grenoble INP (Institute of Engineering Univ. Grenoble Alpes), GIPSA-lab, 38000 Grenoble, France. e-mail: matteo.tacchi@gipsa-lab.fr

<sup>2</sup>Laboratoire d’Automatique, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland. e-mail: {yingzhao.lian, colin.jones}@epfl.ch

1	INTRODUCTION	2
4	<b>Learning robust Lyapunov functions</b>	<b>12</b>
4.1	Robustly distributing information . . . . .	12
4.2	A Convex tractable case: Lipschitz bounds . . . . .	13
4.3	Comparison with related works . . . . .	16
5	<b>Algorithm Development</b>	<b>17</b>
5.1	Learnability . . . . .	19
5.2	Computationally efficient recasting . . . . .	20
5.3	Algorithms . . . . .	22
6	<b>Numerical Results</b>	<b>23</b>
6.1	Non Polynomial dynamic system . . . . .	24
6.2	Reverse Time Van Del Pol Oscillator . . . . .	26
7	<b>Conclusion</b>	<b>26</b>
A	<b>Proof of Theorem 8</b>	<b>30</b>
B	<b>Assessing conservatism</b>	<b>32</b>

## 1 Introduction

Stability analysis is a major research topic in control science. For instance, given a closed-loop control system and its current configuration (initial condition), deciding whether the trajectories will converge towards a stable equilibrium or not is a crucial problem that control engineers are required to solve in real time on a regular basis. Among various stability criteria, Lyapunov analysis [1] plays a key role in this field. In this framework, stability analysis is reformulated into the search for a Lyapunov function (LF), and all initial conditions of trajectories converging towards the stable equilibrium are gathered in the so-called region of attraction (RoA), often approximated from inside with sublevel sets of LFs.

Lyapunov analysis has been widely studied in model-based and data-driven setups, where the user is assumed to have direct access to the model or a high-fidelity simulator. While a system model or simulator may not always be available, it is usually possible and much simpler to measure a finite number of system responses offline. Therefore, stability analysis methods based on fixed measured data become desirable although often challenging. Motivated by this need, this work studies Lyapunov analysis based on a given finite set of measurements of the system response. The proposed approach can learn a piecewise affine (PWA) LF on a compact set without access to a system model or simulator. The contributions of this work are summarized as follows:

- We formulate and prove a Lyapunov inference theorem, which generalizes existing Lyapunov stability analysis methods. This generalization is used to expand a prior inner estimate of the region of attraction to a larger set.
- We specify our Lyapunov stability criterion for a PWA LF on a compact set, to make it verifiable locally on this compact set rather than at all points.
- We make the Lyapunov criterion robust to all models that are consistent with the measured dataset. This criterion is defined on general bounded evaluation function spaces, and has a convex form for Lipschitz functions.

- We develop an algorithm to learn a LF, robust to all models that are consistent with the measured dataset. The proposed algorithm only needs to solve a convex second-order cone program, regardless of the underlying unknown dynamics' (non-)linearity.
- We discuss numerical results and properties of the proposed algorithm, mostly focusing on the improvement of its computational efficiency.

## Previous Work

Nonlinear Lyapunov stability analysis has been widely studied, where model-based approaches and sampling-based approaches form two main categories. In each approach, a Lyapunov candidate is optimized or synthesized by verifying the Lyapunov stability conditions. In model-based approaches, the knowledge of the underlying model is explicitly used in the search of the LF. In contrast, data-driven approaches train an LF by penalizing the violation of the Lyapunov stability conditions on a dataset. Even though a standard Monte-Carlo sampling scheme can also give a probabilistic guarantee [2], it is always preferable to give a strict qualification in stability analysis. In this case, model-based and most data-driven approaches require explicit knowledge of the model. In particular, verification of the Lyapunov stability condition usually resorts to nonlinear optimization or satisfiability modulo theory (SMT) solvers, such as dReal [3]. Note that when smooth dynamics are considered, one can write the Lyapunov stability condition with respect to any Lyapunov candidate into an explicit algebraic form (see e.g [4–6]). The SMT solver is accordingly used to check whether these algebraic inequalities are satisfied up to some user-defined tolerance [7].

To the best of the authors' knowledge, the first numerical method to find an LF solves the Zubov equation [8], which models an LF as the solution to a linear partial differential equation (PDE). The approximation of this PDE is then solved via a series expansion [8], a collocation method [9], etc. One main advantage of the model-based approach is that the a-priori knowledge about the model can be used to reformulate the Lyapunov learning problem into a simpler problem. When polynomial dynamics are considered, a sum of square (SoS) programming relaxation can be used to search for polynomial LFs [10]. Due to the nice algebraic properties of polynomials, the SoS framework has been further used to find the region of attraction [11, 12] and its sparsity structure has been used to improve its scalability [13, 14]. Parallel to the studies in polynomial dynamics, PWA dynamics are another active area of research interest [15, 16], which comes as a result of the ubiquitous appearance of PWA functions in various controllers, such as ReLU-neural-network-based controllers and linear MPC [17]. For the PWA setup, optimization based approaches play a central role, such as linear matrix inequalities [12, 18, 19] and mixed integer programming [20].

Unlike model-based approaches, sampling-based methods rely on an efficient strategy for generating informative samples. Counter-example guided inductive synthesis (CEGIS) [21, 22] is a commonly applied concept in many sample-based approaches (see e.g. [19, 23, 24]) when direct access to the model or its simulator is available. During the learning process, they iteratively augment the sample dataset by adding counter examples to the Lyapunov candidate proposed in the current iteration. These algorithms train the LF by penalizing the violation of the Lyapunov stability condition on the samples, and they converge when no further counter example can be generated [5, 23].

The search for an LF is usually confined to a specific function class, such as a generalized quadratic form [25] or a positive definite kernel regressor [26]. In this work, we will focus on PWA LFs defined on a compact set. Besides the advantages mentioned in the model-based approach paragraph, PWA Lyapunov candidates have shown nice interplay with Lipschitz dynamics. In particular, when the samples of system dynamics are assigned to the vertices of a grid, a robust

Lyapunov stability condition on each simplex can be verified by only considering a tightened Lyapunov condition defined on its vertices. This family of methods is called the continuous piece-wise affine (CPA) method [27, 28]. The CPA method has been extended to more general problem setups: differential inclusions [29], switched systems [30], etc. In this work, we also consider Lipschitz dynamics, but we do not assume that the data are located on the vertices. Therefore, we do not term our method a CPA method to avoid unnecessary confusion. A more detailed comparison is presented in Section 4.3.

Our method can also be related to [31], which leverages partial knowledge on the system dynamics (such as bounds on its Lipschitz constant) to learn candidate LFs from data and provide probabilistic guarantees for this candidate to generalize to unobserved states. Similarly, we use such knowledge to certify Lyapunov conditions globally, although we choose a robust certification rather than probabilistic, as we aim at avoiding any false positives. Moreover, while [31] learns stability certificates, it does not explicitly provide methods for computing the region of attraction, which is the goal of our contribution. Even closer to our work is the reference [32], where knowledge on the Lipschitz constant is complemented with bounds on higher order derivatives, in order to compute Taylor approximations of the dynamics with deterministic error bounds. This results in polynomial sector conditions and SoS relaxations of Lyapunov constraints. Although less conservative than our method, this framework resorts to semidefinite programming (SDP), while we provide certificates computed from second order cone programming (SOCP) problems.

The rest of this paper is organized as follows: Section 2 states the problem setup. In Section 3, we will first generalize the Lyapunov theorem in Section 3.1, this generalization will later be used to develop a local Lyapunov condition with PWA Lyapunov candidate in Section 3.2. In the sequel, Section 4.1 applies this local condition to a set of uncertain function defined by data, whose robust satisfaction is summarized in Theorem 7. This theorem is later used to define a convex inequality condition for the class of Lipschitz function in Section 4.2, where the learning problem will be summarized. A comparison between the proposed learning problem and other related works are given in Section 4.3. The learnability of the proposed scheme is studied in Section 5.1, after which the proposed learning problem is recast into an equivalent form to enable higher computational efficiency in Section 5.2. The general learning algorithm are summarized in 5.3 with a numerical validation in Section 6. A conclusion wraps up this paper in Section 7.

**Notation:**  $\{\mathbf{x}_i\}_{i \in \mathcal{I}}$  is a set indexed by  $\mathcal{I}$ , and when there is no confusion, we drop the index set with  $\{\mathbf{x}_i\}$ .  $\mathbb{N}_a$  denotes the set of positive integer less than  $a$ .  $\mathbb{R}_+$  denotes the set of non-negative real numbers.  $\mathbf{0}$  is a zero vector.  $\mathbf{B}(\mathbf{x}, r)$  denotes an open euclidean ball centred at  $\mathbf{x}$  with radius  $r$ .  $\text{co}(\mathbf{X})$  denotes the convex hull of  $\mathbf{X} \subset \mathbb{R}^d$ .  $\mathcal{L}_d(\mathbf{X})$  is the set of Lipschitz functions from  $\mathbf{X}$  to  $\mathbb{R}^d$ .  $\mathbf{X} \setminus \mathbf{Y} := \{\mathbf{x} \in \mathbf{X} \mid \mathbf{x} \notin \mathbf{Y}\}$  for all  $\mathbf{Y} \subset \mathbf{X}$ .  $\|\mathbf{x}\| := (|x_1|^q + \dots + |x_d|^q)^{1/q}$  denotes the  $q$ -norm of  $\mathbf{x} \in \mathbb{R}^d$  for some fixed  $q \in [1, \infty]$  (with convention  $q = \infty \implies \|\mathbf{x}\| = \max_{1 \leq j \leq d} |x_j|$ ), whose dual norm is denoted by  $\|\mathbf{x}\|^*$ .  $|\mathbf{S}|$  denotes the cardinal of a set  $\mathbf{S}$ . If  $\mathbf{X}$  is a topological space, then  $\text{int}(\mathbf{X})$ ,  $\overline{\mathbf{X}}$  and  $\partial\mathbf{X}$  denote its interior, closure and boundary, respectively.

Finally, due to ambiguity in the literature, we indicate our definition of polyhedra and polytopes:

- A polyhedron  $\mathbf{P} \subset \mathbb{R}^d$  is an intersection of finitely many half spaces:  $\exists c \in \mathbb{N}, \mathbf{A} \in \mathbb{R}^{c \times d}, \mathbf{b} \in \mathbb{R}^c$  such that

$$\mathbf{P} = \{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{A}\mathbf{x} - \mathbf{b} \in (\mathbb{R}_+)^c\}.$$

- A polytope is a bounded, finite union of polyhedra (which is not necessarily convex). In particular, a convex polytope is a bounded polyhedron, and is the convex hull of its vertices.

## 2 Learning regions of attraction from data

### 2.1 Unknown dynamic system, fixed data

In our problem setup, we consider an **unknown** continuous time dynamic system of dimension  $d$  on a bounded open set  $\mathbf{X} \subset \mathbb{R}^d$ :

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) \quad (1a)$$

$$\forall t \geq 0, \quad \mathbf{x}(t) \in \mathbf{X} \quad (1b)$$

whose known behavior is summarized as follows:

**Assumption I.**

1.  $\mathbf{f}$  is Lipschitz continuous, so that the solution to (1) with initial condition  $\mathbf{x}(0) = \mathbf{x}_0$  exists and is unique, and we denote by  $\mathbf{x}(t|\mathbf{x}_0)$  the corresponding flow at time  $t$ .
2.  $\mathbf{f}$  has a single equilibrium point (EP) in  $\mathbf{X}$ , and this EP is in  $\mathbf{0}$  and is locally asymptotically stable (LAS).

Assumption I implies that the hypothesis space  $\mathcal{F}$  for the unknown dynamics  $\mathbf{f}$  is at most the space of Lipschitz continuous vector fields:  $\mathcal{F} \subset \mathcal{L}_d(\mathbf{X})$  with

$$\mathcal{L}_d(\mathbf{X}) := \left\{ \mathbf{h} : \mathbf{X} \rightarrow \mathbb{R}^d \mid \sup_{\mathbf{x} \neq \mathbf{y}} \frac{\|\mathbf{h}(\mathbf{x}) - \mathbf{h}(\mathbf{y})\|}{\|\mathbf{x} - \mathbf{y}\|} < \infty \right\}.$$

The goal of this work is to algorithmically learn an inner approximation of the RoA for this LAS EP:

**Problem 1.** Find the largest possible set of initial conditions of trajectories converging to  $\mathbf{0}$  when  $t$  goes to infinity:

$$\mathbf{R} := \left\{ \mathbf{x}_0 \in \mathbf{X} \mid \lim_{t \rightarrow \infty} \|\mathbf{x}(t|\mathbf{x}_0)\| = 0 \right\}.$$

Even when a model is available, such task is hard in practice, see e.g. [33] and the references therein. When there is no model, some alternative information is needed in order to address Problem 1, summarized in Assumptions II and III.

**Assumption II.** A fixed dataset  $\mathbf{D}$  of size  $n$  sampled from system (1) is available:

$$\mathbf{D} := \{(\mathbf{x}_i, \mathbf{f}_i = \mathbf{f}(\mathbf{x}_i))\}_{i=1}^n \subset \mathbf{X} \times \mathbb{R}^d.$$

Assumption II is the only direct observation that we have of the system behavior, and only covers a finite number of configurations, among the infinite possible samples. Hence, we also need some additional knowledge to be able to generalize from this dataset  $\mathbf{D}$  to the whole set  $\mathbf{X} \times \mathbf{f}(\mathbf{X})$ .

**Assumption III.** From the dataset  $\mathbf{D}$  and side information  $(\Sigma)$ , a set-valued map  $\mathbf{F} : \mathbf{X} \rightrightarrows \mathbb{R}^d$  exists, such that for all  $\mathbf{x} \in \mathbf{X}$ ,  $\mathbf{F}(\mathbf{x})$  is compact and convex, and it holds that

$$\mathbf{f}(\mathbf{x}) \in \mathbf{F}(\mathbf{x}) \quad (2)$$

*Example.* The side information  $(\Sigma)$  can take various forms and is essentially a bound on the complexity of the model  $\mathbf{f}$  outside of the datapoints  $\mathbf{x}_i$ . Here are some examples of such side information:

1. Lipschitz bound:  $\exists M > 0$  such that  $\forall \mathbf{x}, \mathbf{y} \in \mathbf{X}$ ,

$$\|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})\| \leq M\|\mathbf{x} - \mathbf{y}\|.$$

In such case, the uncertainty set is trivially given by

$$\mathbf{F}(\mathbf{x}) = \bigcap_{i=1}^n \mathbf{B}(\mathbf{f}_i, M\|\mathbf{x} - \mathbf{x}_i\|).$$

2. Hilbert bounds [34]: there is a kernel function  $\kappa : \mathbf{X} \times \mathbf{X} \rightarrow \mathbb{R}$  generating a reproducing kernel Hilbert space (RKHS)  $\mathcal{H}$  of real functions over  $\mathbf{X}$  such that  $\mathbf{f} = (f_1, \dots, f_d) \in \mathcal{H}^d$ , as well as upper bounds  $M_1, \dots, M_d > 0$  such that

$$\forall j \in \mathbb{N}_d, \quad \|f_j\|_{\mathcal{H}} \leq M_j.$$

In this setting, it is possible to learn a representer  $\hat{\mathbf{f}} = (\hat{f}_1, \dots, \hat{f}_d) \in \mathcal{H}^d$  for  $\mathbf{f}$  from the dataset  $\mathbf{D}$ , and  $\mathbf{F}$  can be determined through the following kernel inputs:

$$\begin{aligned} \mathbf{K} &:= (\kappa(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^n \in \mathbb{R}^{n \times n}, \\ \boldsymbol{\kappa} &:= \mathbf{x} \mapsto (\kappa(\mathbf{x}, \mathbf{x}_i))_{i=1}^n \in \mathbb{R}^n, \\ P &:= \mathbf{x} \mapsto \sqrt{\kappa(\mathbf{x}, \mathbf{x}) - \boldsymbol{\kappa}(\mathbf{x})^\top \mathbf{K}^{-1} \boldsymbol{\kappa}(\mathbf{x})}, \end{aligned}$$

for which [34] provides a new uncertainty set for  $\mathbf{f}$  (denoting  $[\pm a] := [-a, a]$  for  $a \geq 0$ ):

$$\mathbf{F}(\mathbf{x}) := \hat{\mathbf{f}}(\mathbf{x}) + P(\mathbf{x}) \prod_{j=1}^d \left[ \pm \sqrt{M_j^2 - \|\hat{f}_j\|_{\mathcal{H}}^2} \right].$$

3. Derivative bounds [32]:  $\mathbf{f} \in C^{k+1}(\mathbf{X})^d$  and for  $j \in \mathbb{N}_d$ ,  $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_d) \in \mathbb{N}^d$  such that  $|\boldsymbol{\alpha}| := \alpha_1 + \dots + \alpha_d = k + 1$ , there is an  $M_{j,\boldsymbol{\alpha}} > 0$  verifying

$$\forall \mathbf{x} \in \mathbf{X}, \quad \left| \frac{\partial^{k+1} f_j}{\partial \mathbf{x}^{\boldsymbol{\alpha}}}(\mathbf{x}) \right| \leq M_{j,\boldsymbol{\alpha}},$$

where  $\mathbf{x}^{\boldsymbol{\alpha}} := x_1^{\alpha_1} \dots x_d^{\alpha_d}$ . Here, [32] resorts to degree  $k$  Taylor approximations in  $\boldsymbol{\omega} \in \mathbf{X}$

$$\begin{aligned} \hat{f}_j(\mathbf{x}) &= \sum_{|\boldsymbol{\alpha}| \leq k} c_{j,\boldsymbol{\alpha}} (\mathbf{x} - \boldsymbol{\omega})^{\boldsymbol{\alpha}} \\ \hat{\mathbf{f}}(\mathbf{x}_i) &= \mathbf{f}_i, \end{aligned}$$

which we summarize into  $\hat{\mathbf{f}} \in \mathcal{F}_{\mathbf{D}}^{\boldsymbol{\omega}} \subset \mathbb{R}[\mathbf{x}]_k$ , to get a polynomial sum-of-squares error bound

$$\|\mathbf{f}(\mathbf{x}) - \hat{\mathbf{f}}(\mathbf{x})\|^2 \leq \sigma_{\boldsymbol{\omega}}(\mathbf{x}) := \sum_{j,\boldsymbol{\alpha}} \frac{M_{j,\boldsymbol{\alpha}}^2}{\boldsymbol{\alpha}!} (\mathbf{x} - \boldsymbol{\omega})^{2\boldsymbol{\alpha}},$$

with  $\boldsymbol{\alpha}! := \alpha_1! \dots \alpha_d!$ , and end up with the following uncertainty set, parameterized by a finite set  $\boldsymbol{\Omega}$  of approximation points :

$$\mathbf{F}(\mathbf{x}) = \bigcap_{\boldsymbol{\omega} \in \boldsymbol{\Omega}} \bigcup_{\hat{\mathbf{f}} \in \mathcal{F}_{\mathbf{D}}^{\boldsymbol{\omega}}} \mathbf{B}(\hat{\mathbf{f}}(\mathbf{x}), \sqrt{\sigma_{\boldsymbol{\omega}}(\mathbf{x})}).$$

It is worth noticing that when  $d = 1$ , the case  $k = 0$ ,  $\boldsymbol{\Omega} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$  recovers our first example as a special case; however, in higher dimension  $d$ , the two examples are slightly different, the former bounding norms while the latter bounds projections. Also, all these examples can be modified to account for bounded noise  $\mathbf{f}(\mathbf{x}_i) = \mathbf{f}_i + \mathbf{w}_i$ ,  $\mathbb{P}(\mathbf{w}_i \leq W) = 1$ .

## 2.2 Some set-theoretical considerations

We have defined our dynamic system (1) only on a bounded set  $\mathbf{X}$ . Indeed, in the three above examples, when  $\|\mathbf{x}\|$  goes to infinity, so does the size of the uncertainty set  $\mathbf{F}(\mathbf{x})$  (because  $\|\mathbf{x} - \mathbf{x}_i\|$ ,  $P(\mathbf{x})$  and  $\sigma_\omega(\mathbf{x})$  go to infinity), rendering any learning process ineffective. As a result, one can only aim at certifying stability on a bounded subset of the state space. However, in practice, such assumption is without loss of generality, as most control systems are designed with security constraints under the form of bounds on their state variables. In general, outside of these bounds, the integrity of the system at hand is compromised. For these reasons, we only consider bounded admissible state set  $\mathbf{X}$  in which we assume that the desired RoA  $\mathbf{R}$  is included:  $\mathbf{R} \subset \mathbf{X}$ . We also make an additional assumption on prior knowledge regarding the RoA:

**Assumption IV.** A compact subset of the RoA:  $\mathbf{A} \subset \text{int}(\mathbf{R})$ , is already known, with  $\mathbf{0} \in \text{int}(\mathbf{A})$ .

Assumption IV is a slight generalization of Assumption I (which yields an  $\mathbf{A} = \{\mathbf{0}\}$ ), required for practical learning of the RoA. Generally speaking,  $\mathbf{A}$  models a conservative prior about the region of attraction (RoA) of the unknown dynamic system, e.g. deduced from engineering practice. We stress here that such knowledge can be required in a non-trivial form to learn a finitely parameterized LF, either in a model-based or data-driven setting, as discussed in [35,36].

## 3 Piecewise affine set membership for Lyapunov inference

In this section, we first try to augment the prior knowledge of attractivity in  $\mathbf{A}$  to a larger set in Section 3.1. This result is refined to a Lyapunov candidate from the class of piece-wise affine (PWA) functions in Section 3.2.

### 3.1 Lyapunov inference

Before proceeding to the stability analysis, we introduce three additional concepts on functions  $V : \mathbb{R}^d \rightarrow \mathbb{R}$ .

**Definition 2.**

- The strict sub-level set of  $V$  with level  $a \in \mathbb{R}$  is

$$\mathbf{L}_a^V := \{\mathbf{x} \in \mathbb{R}^d \mid V(\mathbf{x}) < a\}.$$

- The Clarke generalized gradient of  $V$  at a point  $\mathbf{x} \in \mathbb{R}^d$  is the set given by

$$\partial_C V(\mathbf{x}) := \text{co} \left\{ \mathbf{g} \in \mathbb{R}^d \left| \begin{array}{l} \forall \varepsilon > 0, \exists \mathbf{x}_\varepsilon \in \mathbb{R}^d \text{ s.t.} \\ \|\mathbf{x} - \mathbf{x}_\varepsilon\| < \varepsilon, \\ V \text{ is differentiable at } \mathbf{x}_\varepsilon, \\ \|\mathbf{g} - \nabla V(\mathbf{x}_\varepsilon)\| < \varepsilon \end{array} \right. \right\}.$$

- The Clarke-Lie derivative of  $V$  along a set-valued map  $\mathbf{F} : \mathbb{R}^d \rightrightarrows \mathbb{R}^d$  at a point  $\mathbf{x} \in \mathbb{R}^d$  is

$$\dot{V}_{\mathbf{F}}(\mathbf{x}) := \left\{ \mathbf{f}_x^\top \mathbf{g}_x \mid \mathbf{f}_x \in \mathbf{F}(\mathbf{x}), \mathbf{g}_x \in \partial_C V(\mathbf{x}) \right\}.$$

In particular, if there is an  $\mathbf{f}$  such that for all  $\mathbf{x}$  it holds  $\mathbf{F}(\mathbf{x}) = \{\mathbf{f}(\mathbf{x})\}$ , we denote

$$\dot{V}_{\mathbf{f}}(\mathbf{x}) := \left\{ \mathbf{f}(\mathbf{x})^\top \mathbf{g} \mid \mathbf{g} \in \partial_C V(\mathbf{x}) \right\}.$$



*Remark.* The Clarke gradient is a generalized gradient in the sense that if  $V$  is continuously differentiable in a neighbourhood of  $\mathbf{x}$ , then trivially  $\partial_C V(\mathbf{x}) = \{\nabla V(\mathbf{x})\}$ , and if  $V$  is convex in a neighborhood of  $\mathbf{x}$ , then  $\partial_C V(\mathbf{x})$  coincides with the subdifferential of  $V$ .

In [37, Theorem 2.5.1] it is proven that if  $V$  is Lipschitz continuous in a neighbourhood of  $\mathbf{x}$ , then  $\partial_C V(\mathbf{x}) \neq \emptyset$ ; in such a case,  $\partial_C V(\mathbf{x})$  is compact and convex by definition and if  $\mathbf{F}$  also takes compact and convex values, it trivially follows that  $\dot{V}_{\mathbf{F}}$  has the following form:

$$\forall \mathbf{x} \in \mathbb{R}^d, \exists \ell_{\mathbf{x}} \leq u_{\mathbf{x}} \text{ s.t. } \dot{V}_{\mathbf{F}}(\mathbf{x}) = [\ell_{\mathbf{x}}, u_{\mathbf{x}}].$$

Then, we introduce a slight generalization of the Krasovky-LaSalle invariance principle [38].

**Lemma 3.** *Consider the dynamic system (1) and suppose that there is a Lipschitz continuous function  $V : \mathbb{R}^d \rightarrow \mathbb{R}$  s.t.*

$$\forall \mathbf{x} \in \mathbf{X} \setminus \mathbf{A}, \quad \max \dot{V}_{\mathbf{f}}(\mathbf{x}) < 0. \quad (3)$$

*Then, for all  $a \in \mathbb{R}$  such that  $\mathbf{L}_a^V \subset \mathbf{X}$ , it holds*

$$\forall \mathbf{x}_0 \in \mathbf{L}_a^V, \quad \lim_{t \rightarrow \infty} \text{dist}(\mathbf{x}(t|\mathbf{x}_0), \mathbf{A}) = 0, \quad (4)$$

where  $\text{dist}(\mathbf{p}, \mathbf{S}) = \inf_{\mathbf{s} \in \mathbf{S}} \|\mathbf{p} - \mathbf{s}\|$  denotes the distance between point  $\mathbf{p} \in \mathbb{R}^d$  and set  $\mathbf{S} \subset \mathbb{R}^d$ .

*Remark.* The only difference with the original Krasovky-LaSalle invariance principle is the relaxation of the original continuous differentiability assumption on  $V$  into a Lipschitz continuity assumption, and accordingly the use of the Clarke-Lie derivative. This slight modification does not drastically change the proof, thanks to [39, Lemma 2.15] which generalizes the chain rule to this setting. As usual, the case  $\mathbf{A} = \{\mathbf{0}\}$  covers standard Lyapunov local asymptotic stability theorems (constraints on the values of  $V$  being replaced by boundedness of  $\mathbf{X}$ ).

*Remark.* One can immediately notice that if moreover  $\mathbf{A}$  is as in Assumption IV (which is not required in Lemma 3), then condition (3) implies that  $\mathbf{f}$  does not vanish outside of the RoA of the LAS EP  $\mathbf{0}$ . In other words,  $\mathbf{0}$  is implicitly required to be the *only* equilibrium point in  $\mathbf{X}$ . To relax this implicit condition, a slight modification of Lemma 3 can be performed: instead of condition (3), one can ask for the existence of an  $a \in \mathbb{R}$  such that  $\mathbf{L}_a^V \subset \mathbf{X}$  and  $\max \dot{V}_{\mathbf{f}}(\mathbf{x}) < 0$  only holds for  $\mathbf{x} \in \mathbf{L}_a^V \cap (\mathbf{X} \setminus \mathbf{A})$  (instead of for all  $\mathbf{x} \in \mathbf{X} \setminus \mathbf{A}$ ). Then, conclusion (4) would hold only for such  $a \in \mathbb{R}$ . While allowing for the existence of alternative equilibria outside of  $\mathbf{L}_a^V$ , this condition is also more difficult to translate into a convex optimization constraint, hence we did not implement it in our numerical experiments, and focused on the case when  $\mathbf{X}$  is tailored to contain no EP other than  $\mathbf{0}$ .

In the case of unknown dynamics, condition (3) is impossible to check, as  $\dot{V}_{\mathbf{f}}$  takes unknown values. Hence, in our Lyapunov inference theorem we will replace it with a more conservative but certifiable condition in terms of the uncertainty set  $\mathbf{F}(\mathbf{x})$  defined in (2).

**Theorem 4.** *Consider the dynamic system (1) together with the set-valued map  $\mathbf{F}$  defined by (2), and suppose that there exists a Lipschitz continuous function  $V : \mathbb{R}^d \rightarrow \mathbb{R}$  satisfying*

$$\forall \mathbf{x} \in \mathbf{X} \setminus \mathbf{A}, \quad \max \dot{V}_{\mathbf{F}}(\mathbf{x}) < 0. \quad (5)$$

*Then, under Assumptions I to IV, for all  $a \in \mathbb{R}$  such that  $\mathbf{L}_a^V \subset \mathbf{X}$ , it holds  $\mathbf{L}_a^V \subset \mathbf{R}$ , i.e.*

$$\forall \mathbf{x}_0 \in \mathbf{L}_a^V, \quad \lim_{t \rightarrow \infty} \|\mathbf{x}(t|\mathbf{x}_0)\| = 0. \quad (6)$$

*Proof.* It follows from Assumption IV (compact  $\mathbf{A}$  included in open  $\text{int}(\mathbf{R})$ ) that there exists a  $\delta > 0$  such that, if  $\text{dist}(\mathbf{x}, \mathbf{A}) < \delta$ , then  $\mathbf{x} \in \mathbf{R}^1$ . Existence of a  $\tau > 0$  such that

$$\text{dist}(\mathbf{x}(\tau|\mathbf{x}_0), \mathbf{A}) < \delta$$

follows from the fact that for all  $\mathbf{x} \in \mathbf{X}$ ,  $\mathbf{f}(\mathbf{x}) \in \mathbf{F}(\mathbf{x})$ , so that  $\dot{V}_{\mathbf{f}}(\mathbf{x}) \subset \dot{V}_{\mathbf{F}}(\mathbf{x})$  and hence  $\max \dot{V}_{\mathbf{f}}(\mathbf{x}) \leq \dot{V}_{\mathbf{F}}(\mathbf{x}) < 0$  recovers (3) from (5) and we can apply Lemma 3. As a result, using the semigroup property, it holds

$$\begin{aligned} \lim_{t \rightarrow \infty} \|\mathbf{x}(t|\mathbf{x}_0)\| &= \lim_{t \rightarrow \infty} \|\mathbf{x}(t + \tau|\mathbf{x}_0)\| \\ &= \lim_{t \rightarrow \infty} \|\mathbf{x}(t|\mathbf{x}(\tau|\mathbf{x}_0))\| = 0. \end{aligned}$$

□

*Remark.* The point for working with a neighborhood  $\mathbf{A}$  of the equilibrium in Theorem 4 is that this allows us to exclude  $\mathbf{A}$  from the set on which the Lyapunov condition (5) has to be checked. Indeed, on such neighborhood the function  $\mathbf{f}$  becomes vanishingly small, so that in practice, enforcing condition (3) on a finitely parameterized Lyapunov candidate is more challenging (if not impossible) in this region of the state space, see e.g. [35, 36].

Moreover, similarly to Lemma 3, Theorem 4 also includes an implicit condition: in order to allow for the existence of a Lyapunov candidate  $V$  satisfying condition (5), i.e.

$$\forall \mathbf{x} \in \mathbf{X} \setminus \mathbf{A}, \quad 0 > \max \dot{V}_{\mathbf{F}}(\mathbf{x}) = \max_{\substack{\mathbf{f} \in \mathbf{F}(\mathbf{x}) \\ \mathbf{g} \in \partial_{\mathbf{C}} V(\mathbf{x})}} \mathbf{f}^{\top} \mathbf{g},$$

$\mathbf{F}(\mathbf{x})$  should not contain  $\mathbf{0}$  when  $\mathbf{x} \in \mathbf{X} \setminus \mathbf{A}$ , because if  $\mathbf{0} \in \mathbf{F}(\mathbf{x})$  then  $0 \in \dot{V}_{\mathbf{F}}(\mathbf{x})$  and condition (5) cannot hold, regardless of  $V$ . In particular, considering the fact that  $\mathbf{F}$  models uncertainty over  $\mathbf{f}$ , it is reduced to a point only on the finite dataset, where  $\mathbf{F}(\mathbf{x}_i) = \{\mathbf{f}_i\}$  (assuming perfect measurements), and grows in size with the distance to datapoints. Hence, it is very likely that on a neighbourhood of  $\mathbf{0}$ , and given that  $\mathbf{F}(\mathbf{0}) = \{\mathbf{0}\}$ , it will hold  $\mathbf{0} \in \mathbf{F}(\mathbf{x})$  (this happens for example in the very simple case of a Lipschitz bound  $M > 1$ ). This is another justification for excluding a whole neighbourhood  $\mathbf{A}$  of  $\mathbf{0}$  in condition (5).

### 3.2 Piecewise affine Lyapunov function

PWA functions have strong modelling capability because they are dense in the space of continuous functions with a compact domain [40, Chapter 7.4]. This section will refine Theorem 4 to Lipschitz continuous PWA Lyapunov candidates. For the sake of simplicity, we further assume

**Assumption V.**  $\bar{\mathbf{X}}$  and  $\mathbf{A}$  are polytopes.

When a set is not a polytope, it can be inner-approximated by a polytope up to arbitrary accuracy, thus this assumption will not limit the application of the proposed analysis. Recall that the definition of polytope used in this paper is not necessarily convex, but rather is a finite union of polyhedra (Section 2.1).

<sup>1</sup>take e.g.  $\delta = \min_{\mathbf{x} \in \mathbf{A}, \mathbf{y} \in \partial \mathbf{R}} \|\mathbf{x} - \mathbf{y}\|$ ; the min is well-defined as the minimum of a continuous function over the compact set  $\mathbf{A} \times \partial \mathbf{R}$ , whose positivity is given by  $\mathbf{A} \cap \partial \mathbf{R} = \emptyset$ .

We now introduce our Lyapunov candidate under the form of a Lipschitz continuous PWA function. We first consider an  $m$ -piece tessellation of  $\mathbf{X} \setminus \mathbf{A}$ :

$$\overline{\mathbf{X} \setminus \mathbf{A}} = \bigcup_{k=1}^m \mathbf{Y}_k \quad (7a)$$

$$k \neq k' \implies \text{int}(\mathbf{Y}_k) \cap \text{int}(\mathbf{Y}_{k'}) = \emptyset \quad (7b)$$

where the  $\mathbf{Y}_k$  are convex polytopes (in addition we define  $\mathbf{Y}_0 = \mathbf{A} \ni \mathbf{0}$ , not necessarily convex). For  $k \in \{1, \dots, m\}$ , we denote the vertices of  $\mathbf{Y}_k$  by  $\{\mathbf{y}_{k,l}\}_{l=1}^{\nu_k}$  ( $\nu_k$  hence denoting the number of vertices of  $\mathbf{Y}_k$ ). Using this structure, a PWA Lyapunov candidate  $V$  is defined on  $\mathbf{X}$  by

$$\forall k \in \{0, \dots, m\}, \mathbf{x} \in \mathbf{Y}_k, \quad V(\mathbf{x}) := \mathbf{g}_k^\top \mathbf{x} + b_k. \quad (8a)$$

Obviously, our Lyapunov candidate  $V$  should be continuous on  $\mathbf{X}$ : for any common vertex  $\mathbf{y} \in \mathbf{Y}_k \cap \mathbf{Y}_{k'}$  (i.e.  $\exists l \in \mathbb{N}_{\nu_k}, l' \in \mathbb{N}_{\nu_{k'}}$  such that  $\mathbf{y} = \mathbf{y}_{k,l} = \mathbf{y}_{k',l'}$ ), the condition

$$(\mathbf{g}_k - \mathbf{g}_{k'})^\top \mathbf{y} = b_{k'} - b_k \quad (8b)$$

should hold. Then,  $V$  is Lipschitz continuous on  $\mathbf{X}$  with Lipschitz constant

$$\sup_{\mathbf{x} \neq \mathbf{y}} \frac{|V(\mathbf{x}) - V(\mathbf{y})|}{\|\mathbf{x} - \mathbf{y}\|} = \max_{0 \leq k \leq m} \|\mathbf{g}_k\| < \infty.$$

*Remark.* Regardless of the tessellation (7), system (8) always admits  $\{\mathbf{g}_k, b_k\}_{k=0}^m = \mathbf{0}$  as a solution. More specific tessellation methods are discussed later in the paper for which non-trivial solutions exist.

$V$  can then be extended to the whole of  $\mathbb{R}^d$  while keeping the same Lipschitz constant, using Kirszbraun's theorem [41]. Regarding our PWA Lyapunov candidate, the stability condition (5) in Theorem 4 can be restated as follows:

**Lemma 5.** *Given the tessellation (7) of  $\mathbf{X}$ , for  $\mathbf{x} \in \mathbf{X} \setminus \mathbf{A}$  define the index set*

$$\mathbb{K}(\mathbf{x}) := \{k \in \mathbb{N}_m \mid \mathbf{x} \in \mathbf{Y}_k\}.$$

*Then, any Lipschitz extension of the function  $V$  defined by (8) to the whole  $\mathbb{R}^d$  satisfies the following condition:*

$$\forall \mathbf{x} \in \mathbf{X}, \quad \partial_C V(\mathbf{x}) = \text{co}\{\mathbf{g}_k \mid k \in \mathbb{K}(\mathbf{x})\}. \quad (9)$$

*Proof.* Let  $\mathbf{x} \in \mathbf{X} \setminus \mathbf{A}$ . If  $V$  is differentiable in  $\mathbf{x}$ , then for any  $k \in \mathbb{K}(\mathbf{x})$  it holds

$$\partial_C V(\mathbf{x}) = \{\nabla V(\mathbf{x})\} = \{\mathbf{g}_k\}$$

(which proves that for any  $k, k' \in \mathbb{K}(\mathbf{x})$ ,  $\mathbf{g}_k = \mathbf{g}_{k'}$  and thus by continuity  $b_k = b_{k'}$ ). Else,  $V$  is not differentiable in  $\mathbf{x}$ , and we need the broader definition of the Clarke gradient:

$$\partial_C V(\mathbf{x}) = \text{co} \left\{ \mathbf{g} \in \mathbb{R}^d \left| \begin{array}{l} \forall \varepsilon > 0, \exists \mathbf{x}_\varepsilon \in \mathbb{R}^d \text{ s.t.} \\ \|\mathbf{x} - \mathbf{x}_\varepsilon\| < \varepsilon, \\ V \text{ is differentiable in } \mathbf{x}_\varepsilon, \\ \|\mathbf{g} - \nabla V(\mathbf{x}_\varepsilon)\| < \varepsilon \end{array} \right. \right\}.$$

In our case, there is an  $\varepsilon > 0$  small enough such that  $\|\mathbf{x} - \mathbf{x}_\varepsilon\| < \varepsilon$  implies the existence of a  $k \in \mathbb{K}(\mathbf{x})$  with  $\mathbf{x}_\varepsilon \in \mathbf{Y}_k$  (because  $\mathbf{X}$  is open, so that  $\mathbf{x} \notin \partial \mathbf{X}$ ). Then, it follows from the previous

argument that if  $V$  is differentiable in  $\mathbf{x}_\varepsilon$  then  $\nabla V(\mathbf{x}_\varepsilon) = \mathbf{g}_k$ . Conversely, for any  $k \in \mathbb{K}(\mathbf{x})$  there is an  $\mathbf{x}_\varepsilon \in \mathbf{Y}_k \cap \mathbf{B}(\mathbf{x}, \varepsilon)$  in which  $V$  is differentiable, with again  $\nabla V(\mathbf{x}_\varepsilon) = \mathbf{g}_k$ . This allows to recast the Clarke gradient in our special case as

$$\partial_C V(\mathbf{x}) = \text{co} \left\{ \mathbf{g} \in \mathbb{R}^d \mid \begin{array}{l} \forall \varepsilon > 0, \exists k \in \mathbb{K}(\mathbf{x}) \text{ s.t.} \\ \|\mathbf{g} - \mathbf{g}_k\| < \varepsilon \end{array} \right\},$$

which is exactly the set announced in (9).  $\square$

**Proposition 6.** *Let Assumptions I to V hold, and consider the function  $V$  defined by (7), (8). If  $V$  moreover satisfies the following condition*

$$\forall k \in \mathbb{N}_m, \mathbf{x} \in \mathbf{Y}_k, \mathbf{f} \in \mathbf{F}(\mathbf{x}), \quad \mathbf{f}^\top \mathbf{g}_k < 0 \quad (10)$$

Then for all  $a \in \mathbb{R}$  such that  $\mathbf{L}_a^V \subset \mathbf{X}$ , it holds  $\mathbf{L}_a^V \subset \mathbf{R}$ , i.e.

$$\forall \mathbf{x}_0 \in \mathbf{L}_a^V, \quad \lim_{t \rightarrow \infty} \|\mathbf{x}(t|\mathbf{x}_0)\| = 0. \quad (6)$$

*Proof.* Assuming that condition (10) is satisfied, we want to prove (5) in order to apply Theorem 4. Let  $\mathbf{x} \in \mathbf{X} \setminus \mathbf{A}$ ,  $\mathbf{f} \in \mathbf{F}(\mathbf{x})$ ,  $\mathbf{g} \in \partial_C V(\mathbf{x})$ , and let us prove that  $\mathbf{f}^\top \mathbf{g} < 0$ ; closedness of  $\dot{\mathbf{V}}_{\mathbf{F}}(\mathbf{x})$  will conclude. From Lemma 5, we have access to a  $\boldsymbol{\theta} := (\theta_k)_{k \in \mathbb{K}(\mathbf{x})} \in [0, 1]^{\mathbb{K}(\mathbf{x})}$  such that

$$\begin{aligned} \sum_{k \in \mathbb{K}(\mathbf{x})} \theta_k &= 1 \\ \sum_{k \in \mathbb{K}(\mathbf{x})} \theta_k \mathbf{g}_k &= \mathbf{g} \end{aligned}$$

and hence

$$\mathbf{f}^\top \mathbf{g} = \sum_{k \in \mathbb{K}(\mathbf{x})} \theta_k \underbrace{\mathbf{f}^\top \mathbf{g}_k}_{< 0} < 0.$$

$\square$

Finally, we would wrap up this part by sorting out the logic flow in this theoretical Section 3 again. The ultimate goal is to extend some prior knowledge of RoA (i.e.  $\mathbf{A}$ ) to a larger set  $\mathbf{L}_a^V$  via PWA continuous function, which is not smooth. Theorem 4 gives this characterization with respect to a Lipschitz continuous Lyapunov candidate via its Clarke gradient evaluation within the set  $\mathbf{X} \setminus \mathbf{A}$ . A specific characterization based on a continuous PWA Lyapunov candidate is then summarized in Proposition 6. This result reformulates the RoA approximation into a negativity test on each cell  $\mathbf{Y}_k$ .

*Remark.* It is noteworthy that, with a fixed tessellation, the parameters of the Lyapunov candidate on each affine piece (i.e.  $\mathbf{g}_k, b_k$  on  $\mathbf{Y}_k$ ) can be uniquely determined by the function evaluation on the vertices  $\{V(\mathbf{g}_{k,l})\}_{l=1}^{\nu_k}$ .

Another main benefit of a fixed tessellation is that it allows a direct control over the model complexity of the Lyapunov candidate. In particular, consider two Lyapunov candidates  $V_1(\mathbf{x})$  and  $V_2(\mathbf{x})$  with their corresponding partitions  $\{\mathbf{Y}_{1,k}\}$  and  $\{\mathbf{Y}_{2,k}\}$ . Then, we can state that  $V_1(\mathbf{x})$  is a refinement of  $V_2(\mathbf{x})$  (i.e.  $V_1(\mathbf{x})$  has a higher degree of modelling capability than  $V_2(\mathbf{x})$ ) if  $\forall \mathbf{Y}_{2,k}, \exists \{\mathbf{Y}_{1,j}\}_{j \in \mathcal{I}_k}$  such that  $\cup_j \mathbf{Y}_{1,j} = \mathbf{Y}_{2,k}$ . As condition (10) is local to each cell, if one cell  $\mathbf{Y}_k$  violates the assumptions of Proposition 6, then we can refine the model locally by further partitioning  $\mathbf{Y}_k$ .

## 4 Learning robust Lyapunov functions

The developments of the previous section are similar in spirit to set membership methods for dynamical systems, such as [32, 34], although their specific application to computing a region of attraction is relatively new, and their combination with search spaces made of PWA Lyapunov candidates as proposed in Section 3 is original. In this section, we further specialize our approach by leveraging specific structures of the uncertainty set, in order to reach unprecedented scalability properties by writing the overall learning process as a second order cone program (SOCP).

Subsection 4.1 studies the negativity condition (10) for a general hypothesis space, and Subsection 4.2 studies the condition in the most basic hypothesis space, i.e. the Lipschitz function space.

### 4.1 Robustly distributing information

The key to certifying that a PWA Lyapunov candidate is indeed an LF in our setting, consists in verifying the negativity condition (10). This condition can be strengthened for  $k \in \mathbb{N}_m$  into

$$v_k^* := \sup_{\mathbf{x} \in \mathbf{Y}_k} \max_{\mathbf{f} \in \mathbf{F}(\mathbf{x})} \mathbf{f}^\top \mathbf{g}_k < 0,$$

whose left hand side is an optimization problem with linear cost  $\mathbf{f} \mapsto \mathbf{f}^\top \mathbf{g}_k$ . Hence, a natural approach would be to consider the feasible set

$$\mathbf{F}(\mathbf{Y}_k) := \bigcup_{\mathbf{x} \in \mathbf{Y}_k} \mathbf{F}(\mathbf{x}) \quad (11)$$

and ask two questions about it:

1. Is  $\mathbf{F}(\mathbf{Y}_k)$  compact and convex? If so, then  $v_k^*$  is attained on the set  $\mathbf{F}(\mathbf{Y}_k)^*$  made of its extreme points.
2. Can  $\mathbf{F}(\mathbf{Y}_k)^*$  be efficiently parameterized? If so, then it may be possible to compute  $v_k^*$  and test for its sign.

If  $\mathbf{F} : \mathbf{X} \rightrightarrows \mathbb{R}^d$  is determined through Hilbert bounds as in [34], then convexity of  $\mathbf{F}(\mathbf{Y}_k)$  is very difficult to decide. In contrast, if  $\mathbf{F}$  comes from Lipschitz bounds, then it is possible to prove that  $\mathbf{F}(\mathbf{Y}_k)$  is compact and convex. However, even in this simple case, the extreme points  $\mathbf{F}(\mathbf{Y}_k)^*$  are difficult to parameterize as soon as  $n > 1$ , unless one notices a specific structure on  $\mathbf{F}$ , which we will now assume.

**Assumption VI.** There exists a finite set  $\Omega \subset \mathbf{X}$  and a family of set-valued maps  $\mathbf{F}_\omega : \mathbf{X} \rightrightarrows \mathbb{R}^d$  indexed by  $\omega \in \Omega$ , such that for all  $\mathbf{x} \in \mathbf{X}$ , it holds

$$\mathbf{F}(\mathbf{x}) = \bigcap_{\omega \in \Omega} \mathbf{F}_\omega(\mathbf{x}). \quad (12a)$$

and for  $\omega \in \Omega$  the following “local” feasible set is compact:

$$\mathbf{F}_\omega(\mathbf{Y}_k) := \bigcup_{\mathbf{x} \in \mathbf{Y}_k} \mathbf{F}_\omega(\mathbf{x}). \quad (12b)$$

*Remark.* Assumption VI rules out Hilbert bounds from our framework (in their current form). However, it still covers derivative bounds as in [32] as well as Lipschitz bounds.

An intuitive interpretation of Assumption VI would be that the uncertainty on  $\mathbf{f}(\mathbf{x})$  can be sequentially reduced using information related to parameterizing points  $\boldsymbol{\omega} \in \boldsymbol{\Omega}$ . From this observation, the information relative to each point  $\boldsymbol{\omega}$  can be distributed in the learning process, using the following result.

**Theorem 7.** *Under Assumption VI, for any  $k \in \mathbb{N}_m$ , (10) holds if there exists a map  $\boldsymbol{\gamma}_k : \boldsymbol{\Omega} \rightarrow \mathbb{R}^d$  such that*

$$\sum_{\boldsymbol{\omega} \in \boldsymbol{\Omega}} \boldsymbol{\gamma}_{k,\boldsymbol{\omega}} = \mathbf{g}_k \quad (13a)$$

$$\sum_{\boldsymbol{\omega} \in \boldsymbol{\Omega}} \max_{\mathbf{f} \in \mathbf{F}_\omega(\mathbf{Y}_k)} \mathbf{f}^\top \boldsymbol{\gamma}_{k,\boldsymbol{\omega}} < 0 \quad (13b)$$

*Proof.* Let  $\boldsymbol{\omega} \in \boldsymbol{\Omega}$ ,  $\mathbf{x} \in \mathbf{Y}_k$ . Then, by Assumption VI, it holds  $\mathbf{F}(\mathbf{x}) \subset \mathbf{F}_\omega(\mathbf{x})$ , so that taking the union over all possible  $\mathbf{x}$  yields  $\mathbf{F}(\mathbf{Y}_k) \subset \mathbf{F}_\omega(\mathbf{Y}_k)$ . Hence, for any  $\mathbf{f} \in \mathbf{F}(\mathbf{Y}_k)$  it holds

$$\mathbf{f}^\top \boldsymbol{\gamma}_{k,\boldsymbol{\omega}} \leq \max_{\mathbf{f}_\omega \in \mathbf{F}_\omega(\mathbf{Y}_k)} \mathbf{f}_\omega^\top \boldsymbol{\gamma}_{k,\boldsymbol{\omega}},$$

which can be summed over  $\boldsymbol{\omega} \in \boldsymbol{\Omega}$  to get

$$\begin{aligned} \mathbf{f}^\top \mathbf{g}_k &\stackrel{(13a)}{=} \sum_{\boldsymbol{\omega} \in \boldsymbol{\Omega}} \mathbf{f}^\top \boldsymbol{\gamma}_{k,\boldsymbol{\omega}} \\ &\leq \sum_{\boldsymbol{\omega} \in \boldsymbol{\Omega}} \max_{\mathbf{f}_\omega \in \mathbf{F}_\omega(\mathbf{Y}_k)} \mathbf{f}_\omega^\top \boldsymbol{\gamma}_{k,\boldsymbol{\omega}} \end{aligned}$$

and we conclude the proof using (13b).  $\square$

*Remark.* The key concept behind Theorem 7 is the decomposition of the uncertainty set  $\mathbf{F}(\mathbf{x})$  in (12a). In particular, if  $\boldsymbol{\Omega} = \{\mathbf{x}_i\}_{i=1}^n$  is made of the sampled data points, then the quantity  $\max_{\mathbf{f}_\omega \in \mathbf{F}_\omega(\mathbf{Y}_k)} \mathbf{f}_\omega^\top \boldsymbol{\gamma}_{k,\boldsymbol{\omega}}$  is related to the uncertainty quantified from one data point, which usually has an easy-to-evaluate explicit closed solution. In comparison, the explicit solution is usually not available or difficult to evaluate when the whole dataset  $\mathbf{D}$  is considered. For example, when considering Lipschitz bounds, the uncertainty boundary quantified by one data point defines a shifted cone. However, the uncertainty upper and lower bounds are PWA and non-trivial to evaluate [42] when the whole dataset  $\mathbf{D}$  is used. Moreover, it is also reasonable to consider an RKHS, which underpins various uncertainty quantification methods such as Gaussian process regression [43] and deterministic error bound methods [34]. All these methods require computing the inverse of the Gram matrix or solving a second order cone program, which has an easy-to-evaluate explicit solution only when one data point is considered. The consequence of Theorem 7 is that instead of studying compactness and convexity of the  $\mathbf{F}(\mathbf{Y}_k)$ , one is reduced to studying convexity of the  $\mathbf{F}_\omega(\mathbf{Y}_k)$  as well as their extreme points, which can be much simpler.

## 4.2 A Convex tractable case: Lipschitz bounds

Theorem 7 gives a representation of condition (10), but such a representation remains abstract and hard to check numerically; for instance, with derivative bounds it holds

$$\mathbf{F}_\omega(\mathbf{x}) = \bigcup_{\hat{\mathbf{f}} \in \mathcal{F}_\omega^{\mathbf{B}}} \mathbf{B}(\hat{\mathbf{f}}(\mathbf{x}), \sqrt{\sigma_\omega(\mathbf{x})}),$$

and although convexity of  $\mathbf{F}_\omega(\mathbf{Y}_k)$  can be deduced from the convexity of  $\mathcal{F}_\mathbf{D}^\omega$  and the cone of SoS polynomials, its extreme points remain very difficult to identify, so that using Theorem 7 does not actually simplify the computations: one still has to solve SDP problems. Thus, we will now recast this representation under a tractable form, in a specific case. Although we have discussed that it is possible to consider a more complex side information on top of Lipschitz bounds in Section 2.1, this section will show that a convex learning problem exists when we consider this most basic hypothesis. Indeed, a minimal assumption to deduce the general behavior of the dynamics  $\mathbf{f}$  on  $\mathbf{X}$  from local data  $\mathbf{D}$ , is formulated in terms of bounding the variations of  $\mathbf{f}$ .

**Assumption VII.** An upper bound  $M$  on the Lipschitz constant of the unknown vector field  $\mathbf{f}$  is known:

$$\exists M > 0; \quad \forall \mathbf{x}, \mathbf{y} \in \mathbf{X}, \quad \|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})\| \leq M\|\mathbf{x} - \mathbf{y}\|. \quad (14)$$

In this case, the following corollary to Proposition 6 and Theorem 7 allows one to drastically simplify the numerical treatment of constraint (10).

**Theorem 8.** *Let Assumptions I, II, IV, V and VII hold, and consider the function  $V$  defined by (7), (8). For  $k \in \mathbb{N}_m$ , let  $\{\mathbf{y}_{k,l}\}_{l=1}^{\nu_k}$  be the set of vertices of  $\mathbf{Y}_k$ , so that*

$$\mathbf{Y}_k = \text{co}\{\mathbf{y}_{k,l}\}_{l=1}^{\nu_k}. \quad (15a)$$

Suppose that  $\forall k \in \mathbb{N}_m$ , there is a set  $\{\gamma_{i,k}\}_{i=1}^n \subset \mathbb{R}^d$  with

$$\sum_{i=1}^n \gamma_{i,k} = \mathbf{g}_k \quad (15b)$$

and for all  $l \in \mathbb{N}_{\nu_k}$

$$\sum_{i=1}^n \mathbf{f}_i^\top \gamma_{i,k} + M\|\gamma_{i,k}\|^* \|\mathbf{x}_i - \mathbf{y}_{k,l}\| < 0, \quad (15c)$$

where we recall that there is a  $q \in [1, \infty]$ ,  $q' := q/(q-1)$  (with conventions  $1/\infty = 0$  and  $1/0 = \infty$ ) such that for  $\phi \in \mathbb{R}^d$ ,  $\|\phi\|^q = |\phi_1|^q + \dots + |\phi_d|^q$  and  $\|\phi\|^{*q'} = |\phi_1|^{q'} + \dots + |\phi_d|^{q'}$ .

Then,  $\forall a \in \mathbb{R}$  such that  $\mathbf{L}_a^V \subset \mathbf{X}$ , it holds  $\mathbf{L}_a^V \subset \mathbf{R}$ , i.e.

$$\forall \mathbf{x}_0 \in \mathbf{L}_a^V, \quad \lim_{t \rightarrow \infty} \|\mathbf{x}(t|\mathbf{x}_0)\| = 0. \quad (6)$$

*Proof.* See Appendix A. □

*Remark.* Theorem 8 exploits two features specific to Assumption VII, namely the fact that  $\mathbf{F}_i(\mathbf{x})$  is centered at  $\mathbf{f}_i$  (in contrast, with Hilbert or derivative bounds, it is centered at some approximant  $\widehat{\mathbf{f}}(\mathbf{x})$  of  $\mathbf{f}(\mathbf{x})$ ) and that for all  $\mathbf{x} \in \mathbf{X}$ ,  $\max_{\mathbf{f} \in \mathbf{F}_i(\mathbf{x})} \|\mathbf{f} - \mathbf{f}_i\|$  is upper bounded by a convex function of  $\mathbf{x}$ , which attains a maximum on  $\mathbf{Y}_k$  at a vertex  $\mathbf{y}_{k,l}$ . In comparison, the shapes of uncertainty sets related to Hilbert or derivative bounds are much more difficult to analyse.

Theorem 8 implies that, under Assumptions I, II, IV, V and VII, existence of an LF giving access to a robust inner approximation  $\mathbf{L}_a^V$  of the RoA  $\mathbf{R}$  of system (1), is obtained as the consequence of the feasibility of the system of constraints (7), (8), (15). Further assuming that the tessellation (7) is fixed and recalling again that for  $\phi \in \mathbb{R}^d$ ,

$$\|\phi\| = \left( \sum_{j=1}^d |x_j|^q \right)^{1/q},$$

this corresponds to:

- linear (polytopic) constraints if  $q = 1$ :  $\forall j, k, l$

$$\sum_{i=1}^n \mathbf{f}_i^\top \boldsymbol{\gamma}_{i,k} + M \gamma_{i,j,k} \sum_{r=1}^d |x_{i,r} - y_{k,l,r}| < 0$$

- second order cone constraints if  $q = 2$ :  $\forall k, l$

$$\sum_{i=1}^n \mathbf{f}_i^\top \boldsymbol{\gamma}_{i,k} + M \|\boldsymbol{\gamma}_{i,k}\| \cdot \|\mathbf{x}_i - \mathbf{y}_{k,l}\| < 0.$$

In contrast, contributions like [32] or [31] resort to the more expensive framework of sum-of-squares programming, which belongs to the class of semidefinite programming and whose complexity grows quickly with the state space dimension and degree of the considered certificates, while [23] is based on the nonconvex framework of mixed-integer programming.

Introducing a fixed negativity tolerance  $\varepsilon > 0$  as well as slack variables  $s_{k,l} \in \mathbb{R}$ ,  $k \in \mathbb{N}_m$ ,  $l \in \mathbb{N}_{\nu_k}$ , we replace the strict inequality constraint (15c) with a large inequality constraint, and learn an LF through solving the following optimization problem.

**Problem 9.** Find  $\{b_k\}_{k=0}^m \subset \mathbb{R}$ ,  $\{\boldsymbol{\gamma}_{i,k}\}_{1 \leq k \leq m} \subset \mathbb{R}^d$ ,  $\mathbf{g}_0 \in \mathbb{R}^d$ ,  $\{\{s_{k,l}\}_{l=1}^{\nu_k}\}_{k=1}^m \subset \mathbb{R}$  solution to the program

$$s_\varepsilon^* := \min \sum_{k=1}^m \sum_{l=1}^{\nu_k} s_{k,l}$$

if  $\mathbf{y}_{k,l} \in \mathbf{Y}_0$ ,  $\sum_{i=1}^n \boldsymbol{\gamma}_{i,k}^\top \mathbf{y}_{k,l} = \mathbf{g}_0^\top \mathbf{y}_{k,l} + b_0 - b_k$  (8b)

if  $\mathbf{y}_{k,l} \in \mathbf{Y}_{k'}$ ,  $\sum_{i=1}^n (\boldsymbol{\gamma}_{i,k} - \boldsymbol{\gamma}_{i,k'})^\top \mathbf{y}_{k,l} = b_{k'} - b_k$  (13a)

$\forall k \in \mathbb{N}_m, l \in \mathbb{N}_{\nu_k}, s_{k,l} \geq -\varepsilon$  (16a)

and  $\sum_{i=1}^n \mathbf{f}_i^\top \boldsymbol{\gamma}_{i,k} + M \|\boldsymbol{\gamma}_{i,k}\|^* \|\mathbf{x}_i - \mathbf{y}_{k,l}\| \leq s_{k,l}$ . (16b)

Optimization problem (16) comes with the following result:

**Proposition 10.** Under Assumption V and tessellation given by (7), for any  $\mathbf{f}$  satisfying Assumptions I, II, IV, and VII and any optimal solution to Problem 9 satisfying

$$s_\varepsilon^* = -\varepsilon \sum_{k=1}^m \nu_k, \tag{17}$$

the function  $V$  defined for  $k \in \mathbb{N}_m$ ,  $\mathbf{x} \in \mathbf{Y}_k$  by

$$V(\mathbf{x}) = \sum_{i=1}^n \boldsymbol{\gamma}_{i,k}^\top \mathbf{x} + b_k \tag{8a+15b}$$

defines a generalized Lyapunov function, such that for all  $a \in \mathbb{R}$  satisfying  $\mathbf{L}_a^V \subset \mathbf{X}$ , it holds  $\mathbf{L}_a^V \subset \mathbf{R}$ , i.e.

$$\forall \mathbf{x}_0 \in \mathbf{L}_a^V, \quad \lim_{t \rightarrow \infty} \|\mathbf{x}(t|\mathbf{x}_0)\| = 0. \tag{6}$$



*Proof.*  $s_\varepsilon^* = -\varepsilon \sum_{k=1}^m \nu_k$  in Problem 9 directly implies that for all  $k \in \mathbb{N}_m$ ,  $l \in \mathbb{N}_{\nu_k}$ , it holds  $s_{k,l} = -\varepsilon$ , i.e. constraint (16a) is saturated. Reinjecting in constraint (16b) yields (15c) as  $\varepsilon > 0$ . The conclusion then follows directly from Theorem 8.  $\square$

The implication of this result is strong as it states that under the assumption of Lipschitz dynamics, we can learn/validate a robust estimate of the RoA  $\mathbf{R}$  through convex programming (16) even when the unknown underlying dynamic system  $\mathbf{f}$  is nonlinear.

*Remark.* It is possible to adapt the proposed scheme (16) to the case where the measurements are contaminated by bounded measurement noise. More specifically, if one has only access to a dataset  $\tilde{\mathbf{D}} := \{(\tilde{\mathbf{x}}_i, \tilde{\mathbf{f}}_i)\}_{i=1}^n \subset (\mathbb{R}^d)^2$  and bounds  $W, W' > 0$  such that there exists  $\mathbf{D} = \{(\mathbf{x}_i, \mathbf{f}_i)\}_{i=1}^n \subset \mathbf{X} \times \mathbb{R}^d$  and a set of random variables  $\mathbf{W} = \{(\mathbf{w}_i, \mathbf{w}'_i)\}_{i=1}^n \subset (\mathbb{R}^d)^2$  satisfying  $\mathbf{f}(\mathbf{x}_i) = \mathbf{f}_i$  and

$$\begin{aligned} \tilde{\mathbf{x}}_i &= \mathbf{x}_i + \mathbf{w}_i, & \mathbb{P}(\|\mathbf{w}_i\| \leq W) &= 1 \\ \tilde{\mathbf{f}}_i &= \mathbf{f}_i + \mathbf{w}'_i, & \mathbb{P}(\|\mathbf{w}'_i\| \leq W') &= 1 \end{aligned}$$

The constraint (16b) is accordingly modified to

$$\sum_{i=1}^n \mathbf{f}_i^\top \boldsymbol{\gamma}_{i,k} + \|\boldsymbol{\gamma}_{i,k}\|^* \left( M (\|\mathbf{y}_{k,l} - \mathbf{x}_i\| + W) + W' \right) \leq s_{k,l}.$$

For the sake of a clear presentation, we only consider noise-free measurements in the rest of this work.

### 4.3 Comparison with related works

We would like to wrap up this subsection by comparing the proposed learning scheme with other existing methods. In comparison with other PWA LF based methods (see e.g. [27, 29]), the proposed scheme shows two major differences. First, the location of the samples and the tessellation of the PWA Lyapunov candidate are decoupled in our scheme. In contrast, in existing PWA LF based methods, the data are sampled on the vertices of the tessellation (i.e.  $\forall k, l, \exists i$  such that  $\mathbf{y}_{k,l} = \mathbf{x}_i$ ), therefore, the data locations are usually structural due to the choice of the tessellation. Second, and this is a consequence of the first point, the robust Lyapunov stability conditions derived from (5) in the existing methods only consider the model uncertainty quantified by one data point: for instance, in [29], instead of our condition (15c), the authors propose to prove exponential stability with the condition  $\mathbf{f}_i^\top \mathbf{g}_k + A_k \|\mathbf{g}_k\|_1 \leq -\|\mathbf{x}_i\|$  for  $i \in \mathbb{N}_n$  and some well-chosen generalization constant  $A_k$  (playing the same role as our  $M$ ),  $\|\cdot\|_1$  denoting the choice  $q = 1$  for our norm  $\|\cdot\|$ . On the contrary, the scheme we propose synthetically makes use of the uncertainty quantified by each data point while maintaining a convex tractable structure.

Another framework related to the proposed approach is the set-membership method [42]. In short, the set-membership method consists in looking for a stability certificate that will be a Lyapunov function for any Lipschitz vector fields  $\hat{\mathbf{f}}$  satisfying  $\hat{\mathbf{f}}(\mathbf{x}) \in \mathbf{F}(\mathbf{x})$  for the uncertainty set-valued map  $\mathbf{F}$  from Assumption III. In [44, 45], methods are developed for the special case  $d = 1$  (in [45] the case  $d > 1$  is deduced through assuming component-wise Lipschitz bound, i.e.  $\forall j \in \mathbb{N}_d, \exists M_j > 0$  with  $|\mathbf{f}_j(\mathbf{x}) - \mathbf{f}_j(\mathbf{y})| \leq M_j \|\mathbf{x} - \mathbf{y}\|$ ), in which case  $f(\mathbf{x}) \in F(\mathbf{x}) = [\underline{f}(\mathbf{x}), \bar{f}(\mathbf{x})]$  with PWA envelope functions  $\underline{f}$  and  $\bar{f}$ .

Now, we will show the difference between our approach and standard set membership regarding real-valued functions. To better demonstrate the difference, we consider a specific example  $f$  (Figure 1), whose Lipschitz overestimate is set to  $M = 1$  and the data points are:

$$\{(0, f(0) = -0.4), (0.3, f(0.3) = -0.5), (1, f(1) = -0.6)\}$$

Now, consider an LF candidate  $V(x) = gx$  with  $g = 0.9$  within interval  $[0, 1]$ . If the evaluation bounds of the set-membership method are used, the Lyapunov decreasing condition needs to be examined in all the sub-intervals generated by the PWA bounds (plotted as two-headed arrow in Figure 1). These intervals are  $[0, 0.1]$ ,  $[0.1, 0.2]$ ,  $[0.2, 0.3]$ ,  $[0.3, 0.6]$ ,  $[0.6, 0.7]$ ,  $[0.7, 1]$ ; in all generality, determining such intervals is computationally heavy. Instead, if we hope to simplify the analysis by only taking one data point into consideration as done in [27, 29], none of the simple models generated by one data point can justify the Lyapunov decreasing condition, as each cone spanned by a single data point intersects the half space  $gf(x) \geq 0$  (see black lines of different markers in Figure 1): the use of the whole dataset is necessary. The proposed scheme synthesizes the knowledge of simple models deduced from each datapoint via convex optimization. One optimal solution to the proposed scheme is plotted as a blue line in Figure 1, which only utilizes the last two points  $(0.3, -0.5)$  and  $(1, -0.6)$  (i.e.  $\gamma_1 = 0$ ). This plot shows that the approach we propose induces a relaxation gap between the bounds it provides and the tight set-membership bounds; what we obtain in exchange for this lower accuracy is a much more tractable convex optimization scheme. It is worth noting that, in this example, if we only consider the left data and the right data point, the Lyapunov decreasing condition will fail even when the set-membership method is used. In such case, the proposed method is able to search for the additional data point locations that are relevant to the Lyapunov decreasing condition. Additionally, this process is done by polynomial time convex optimization algorithms [46]. On the contrary, even though the set-membership method gives the tightest bound, checking the Lyapunov decreasing condition with these bounds is NP-hard, as it requires vertex elimination of the Voronoi cells.

Recently, the set-membership approach was extended by Martin and Allgöwer in [32]. In their contribution, the authors extend Lipschitz bounds to derivative bounds (see Example in Section 2.1), that are used to deduce a functional set-membership parameterization, namely they look for certificates (for dissipativity rather than attractivity) that are consistent with any vector field  $\hat{\mathbf{f}} \in \mathcal{F}_{\mathbf{B}}^{\omega}$ , for some approximation point  $\omega \in \mathbb{R}^d$ . The key idea in their methodology is that  $\mathcal{F}_{\mathbf{B}}^{\omega}$  can be represented in terms of polynomial sums of squares, that are themselves parameterized by linear matrix inequalities (LMI). This very general framework competes with other set membership methods in terms of accuracy, and successfully rids itself of the corresponding NP-hardness by approximating positivity constraints with LMIs through the celebrated Positivstellensatz. However, one of the involved LMIs has size  $2\binom{d+k}{d} + d$  (see  $\Xi_i$  in LMI (8) of [32]), i.e. combinatorial in the number of state variables and the maximal order of the derivative bounds, hence preventing the method from scaling to high dimensions. In contrast, the approach we propose relies on SOCP and LP formulations, that are known to scale much better than SDP.

To summarize this discussion, the approach we propose can be seen as a trade-off between computationally efficient methods based on PWA LFs such as [27, 29] on one hand, and more accurate set-membership based methods as [32, 42, 44, 45] on the other hand. As a result, it achieves better accuracy and sample-efficiency than the former (see the example illustrated on Figure 1) while being theoretically more scalable than the latter.

## 5 Algorithm Development

After the introduction of the Lyapunov learning problem 9, we will discuss its learnability in Section 5.1. The original learning problem will be recast to an equivalent but numerically more efficient form in Section 5.2. In the end, the main algorithms are summarized in Section 5.3.

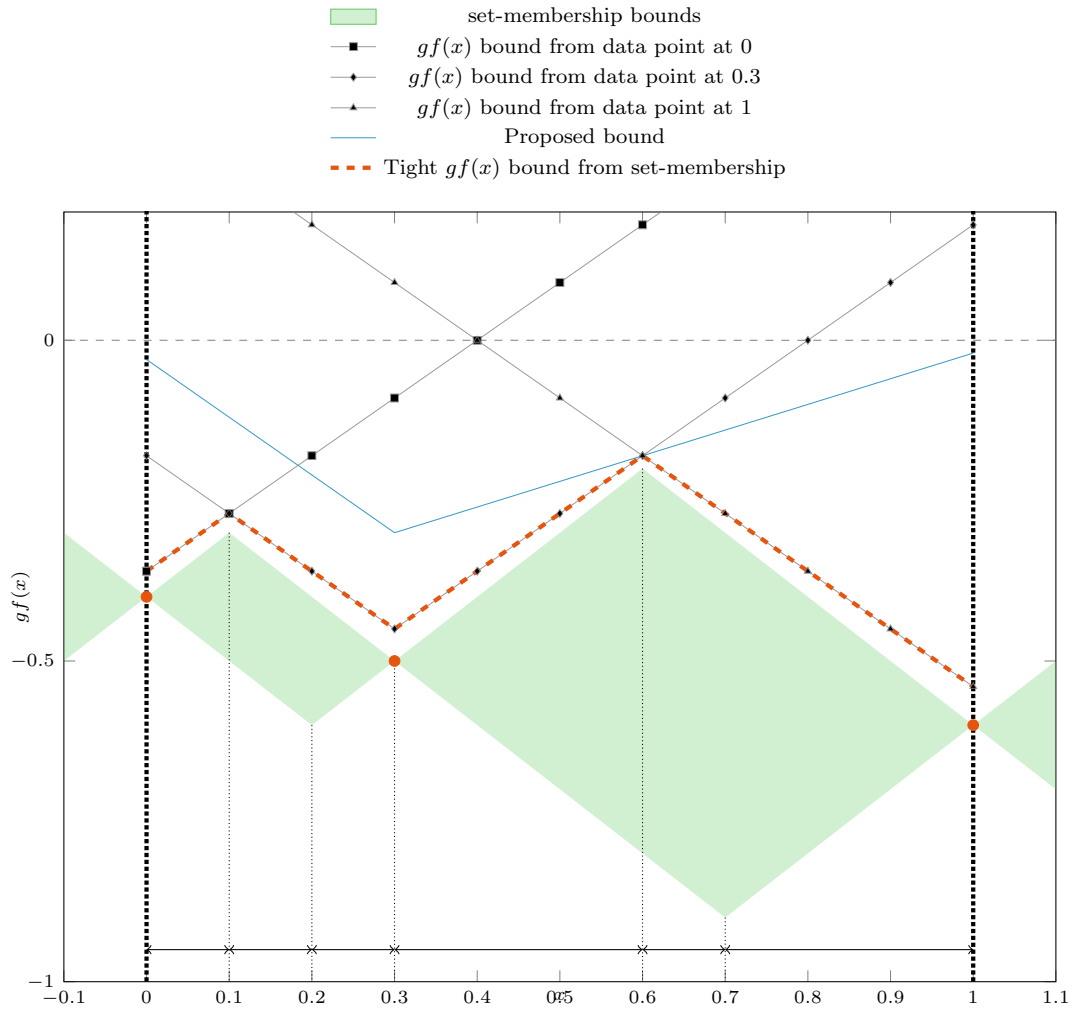


Figure 1: Comparison between set membership method and the proposed method. The proposed scheme is evaluated by  $\tilde{g}_1 = 0, \tilde{g}_2 = 0.65, \tilde{g}_3 = 0.35$ .

## 5.1 Learnability

First, due to the introduction of the slack variables  $\{s_{k,l}\}$ , the learning problem 9 is always feasible. It is natural to ask the following core question in the limiting case:

*Given an RoA prior estimate  $\mathbf{A}$ , what condition on dataset  $\mathbf{D}$  should hold to enable learning the PWA Lyapunov candidate on  $\mathbf{X} \setminus \mathbf{A}$ ?*

Obviously, it is impossible to answer this question with a sufficient condition, regarding the arbitrariness of the unknown dynamic system  $\mathbf{f}$ . However, if we only assume a bound on the Lipschitz constant of  $\mathbf{f}$  as in Assumption VII, we can still give an initial check of the learnability of problem 9, based on the remark we already made on the fact that  $\mathbf{F}(\mathbf{x})$  should not contain  $\mathbf{0}$  for  $\mathbf{x} \in \mathbf{X} \setminus \mathbf{A}$ . In order to discuss this necessary condition, we first define

$$\rho_i := \frac{\|\mathbf{f}_i\|}{M}.$$

We state the necessary condition as follows:

**Lemma 11.** *With tessellation (7) and under Assumptions I, II, IV, V and VII, if solutions to problem 9 define an LF as in Proposition 10, then it holds*

$$\mathbf{X} \setminus \mathbf{A} \subset \bigcup_{i=1}^n \mathbf{B}(\mathbf{x}_i, \rho_i). \quad (18)$$

*Proof.* As we already proved, under our assumptions, the existence of an optimal solution to problem 9 satisfying condition (17) implies that the Lyapunov condition (10) is satisfied on  $\mathbf{X} \setminus \mathbf{A}$ , i.e.

$$\forall k \in \mathbb{N}_m, \mathbf{x} \in \mathbf{Y}_k, \mathbf{f} \in \mathbf{F}(\mathbf{x}), \quad \mathbf{f}^\top \mathbf{g}_k < 0.$$

In particular, for all  $k \in \mathbb{N}_m, \mathbf{x} \in \mathbf{Y}_k$ , it holds  $\mathbf{0} \notin \mathbf{F}(\mathbf{x})$ , otherwise one would get the contradiction  $0 = \mathbf{0}^\top \mathbf{g}_k < 0$ . Now, from Assumption VII we know that

$$\mathbf{F}(\mathbf{x}) = \bigcap_{i=1}^n \mathbf{B}(\mathbf{f}_i, M\|\mathbf{x} - \mathbf{x}_i\|),$$

so that  $\mathbf{0} \notin \mathbf{F}(\mathbf{x})$  means existence of an  $i \in \mathbb{N}_n$  such that  $\mathbf{0} \notin \mathbf{B}(\mathbf{f}_i, M\|\mathbf{x} - \mathbf{x}_i\|)$ , i.e.

$$M\rho_i = \|\mathbf{f}_i\| = \|\mathbf{f}_i - \mathbf{0}\| > M\|\mathbf{x} - \mathbf{x}_i\|,$$

and hence  $\mathbf{x} \in \mathbf{B}(\mathbf{x}_i, \rho_i)$ . This yields the announced result:

$$\mathbf{X} \setminus \mathbf{A} \subset \bigcup_{k=1}^m \mathbf{Y}_k \subset \bigcup_{i=1}^n \mathbf{B}(\mathbf{x}_i, \rho_i).$$

□

*Remark.* Again, notice that the necessary condition (18) for learnability implies the Borel-Lebesgue property, i.e. that  $\overline{\mathbf{X} \setminus \mathbf{A}}$  is compact, and hence that  $\mathbf{X}$  is bounded, which further supports our assumption in that direction.

Lemma 11 shows the connection between learning a PWA LF and the set covering problem, which was proved to be equivalent to a non-convex semi-infinite problem [47], and thus one should not try to check the condition in Lemma 11 numerically. Recall a key idea behind the problem 9: the global analysis on  $\mathbf{X} \setminus \mathbf{A}$  is reduced to the analysis on the vertices. This inspires us to relax the continuous set covering problem to the covering problem of the vertices and we state this condition in the following Corollary:

**Corollary 12.** *With tessellation (7) and under Assumptions I, II, IV, V and VII, if solutions to problem 9 define an LF as in Proposition 10, then it holds*

$$\{\mathbf{y}_{k,l}\}_{\substack{1 \leq k \leq m \\ 1 \leq l \leq \nu_k}} \subset \bigcup_{i=1}^n \mathbf{B}(\mathbf{x}_i, \rho_i). \quad (19)$$

This necessary condition (19) can be checked in polynomial time. If this test fails, it means that there exist a  $\mathbf{y}_{k,l} \notin \bigcup_{i=1}^n \mathbf{B}(\mathbf{x}_i, \rho_i)$ , and therefore, the dataset  $\mathbf{D}$  is not informative enough to learn a PWA LF by only assuming the Lipschitz condition (14). Accordingly, the learning process will be aborted. Intuitively, the vertices  $\mathbf{y}_{k,l} \notin \bigcup_{i=1}^n \mathbf{B}(\mathbf{x}_i, \rho_i)$  should suggest the location where additional samples are required. We leave the investigation about this aspect for future work.

## 5.2 Computationally efficient recasting

In this part, we will discuss how we recast the original problem 9 to an equivalent problem that can be handled numerically more efficiently.

### Data Refinement

One main computational bottleneck for the original problem 9 comes from the number of decision variables. Without loss of generality, we consider an affine piece  $\mathbf{Y}_k$  and look for indices  $i$  that could be removed from  $\mathbb{N}_n$  without jeopardizing the tessellation validation condition (19). In other words, we look for  $i \in \mathbb{N}_n$  such that, for all  $l \in \mathbb{N}_{\nu_k}$ ,  $\mathbf{y}_{k,l} \notin \mathbf{B}(\mathbf{x}_i, \rho_i)$ . For such index  $i$ , and for any  $\boldsymbol{\gamma} \in \mathbb{R}^d$ , it holds  $\|\mathbf{x}_i - \mathbf{y}_{k,l}\| > \rho_i$  and hence, using Hölder's inequality:

$$\begin{aligned} \mathbf{f}_i^\top \boldsymbol{\gamma} + M \|\boldsymbol{\gamma}\|^* \|\mathbf{x}_i - \mathbf{y}_{k,l}\| &> \mathbf{f}_i^\top \boldsymbol{\gamma} + M \rho_i \|\boldsymbol{\gamma}\|^* \\ &= \mathbf{f}_i^\top \boldsymbol{\gamma} + \|\boldsymbol{\gamma}\|^* \|\mathbf{f}_i\| \\ &\geq \mathbf{f}_i^\top \boldsymbol{\gamma} - \boldsymbol{\gamma}^\top \mathbf{f}_i = 0. \end{aligned}$$

Hence, giving a nonzero value to the  $\gamma_{i,k}$  corresponding to this choice of  $k$  and  $i$  cannot help enforcing the strict negative Lyapunov condition (16b), so that it is optimal to fix such  $\gamma_{i,k}$  to  $\mathbf{0}$ . Such choice is equivalent to removing the index  $i$  from the sum in the Lyapunov condition (16b)

$$\sum_{i=1}^n \mathbf{f}_i^\top \gamma_{i,k} + M \|\gamma_{i,k}\|^* \|\mathbf{x}_i - \mathbf{y}_{k,l}\| \leq s_{k,l}.$$

Hence, we can define the set of data indices relevant to  $\mathbf{Y}_k$ :

$$\mathbb{I}_k := \{i \in \mathbb{N}_n \mid \mathbf{B}(\mathbf{x}_i, \rho_i) \cap \{\mathbf{y}_{k,l}\}_{l=1}^{\nu_k} \neq \emptyset\} \quad (20a)$$

and replace the dataset  $\mathbf{D} = \{(\mathbf{x}_i, \mathbf{f}_i)\}_{i=1}^n$  with  $\mathbf{D}_k := \{(\mathbf{x}_i, \mathbf{f}_i)\}_{i \in \mathbb{I}_k} \subset \mathbf{D}$  when working on the polytopic cell  $\mathbf{Y}_k$ , recasting condition (16b) as

$$\sum_{i \in \mathbb{I}_k} \mathbf{f}_i^\top \gamma_{i,k} + M \|\gamma_{i,k}\|^* \|\mathbf{x}_i - \mathbf{y}_{k,l}\| \leq s_{k,l}. \quad (20b)$$

The condition defining the new dataset  $\mathbf{D}_k$  essentially states that each datapoint should at least bring useful information to enforce strict negativity on constraint (16b) at one vertex. This technique can significantly reduce the computational cost. To see this, we consider a homogeneous tessellation within a unit hypercube centered at  $\mathbf{0}$  within which data points scatter uniformly. We further assume that the cells of the tessellation are hypercubes with edge width  $\eta$ . Then, considering a datapoint  $\mathbf{x}_i$ , the ball  $\mathbf{B}(\mathbf{x}_i, \rho_i)$  intersects at most  $O((\rho_i/\eta)^d) \ll m$  hypercubic cells. Notice that in that specific case, it holds

$$\rho_i = \frac{\|\mathbf{f}_i\|}{M} = \frac{\|\mathbf{f}_i - \mathbf{0}\|}{M} \leq \frac{M\|\mathbf{x}_i - \mathbf{0}\|}{M} = \|\mathbf{x}_i\| \leq 0.5,$$

so that the number of decision variables are reduced to roughly  $O(\frac{1}{2^d})$  of the problem defined by the whole data set. In the numerical example we consider in Section 6, we observe on average an 81% reduction in the number of decision variables, which makes the problem tractable on a Laptop without memory overflow.

### Explicit SOCP formulations

When dealing with the euclidean norm (i.e.  $q = 2$ ), in the numerical implementation, it is critical to convert the inequality constraint (20b) into a set of second order cone constraints [48]:

$$s_{k,l} \geq \sum_{i \in \mathbb{I}_k} \mathbf{f}_i^\top \boldsymbol{\gamma}_{i,k} + z_{i,k,l} \quad (21a)$$

$$z_{i,k,l} \geq M \|\boldsymbol{\gamma}_{i,k}\| \|\mathbf{y}_{k,l} - \mathbf{x}_i\|, \quad (21b)$$

where  $|\mathbb{I}_k|$  auxiliary decision scalar variables  $\{z_{i,k,l}\}$  are introduced per vertex  $\mathbf{y}_{k,l}$ . More precisely, the complexity for solving an SOCP scales like  $O(N + K + L)^{3.5}$  where  $N$  is the number of variables,  $K$  the cumulated dimension of the cone constraints, and  $L$  the number of linear constraints, while solving an SDP scales like  $O(N + K^2 + L)^{3.5}$ , where  $K$  denotes the size of the involved LMI. In our case,  $N$ ,  $K$  and  $L$  all scale linearly with the sample size  $n$ , resp. state dimension  $d$ , resp. number of tessellation vertices  $\nu = \nu_1 + \dots + \nu_m$ , so that the overall complexity would be  $O(P^7)$  ( $P \in \{n, d, \nu\}$  being the varying parameter) without the SOCP reformulation, versus  $O(P^{3.5})$  with the SOCP recasting.

### The Recast Problem

After the above reformulations, one ends up numerically solving the following problem:

**Problem 13.** Find  $\{\{s_{k,l}\}_{l=1}^{\nu_k}\}_{k=1}^m$ ,  $\{\{z_{i,k,l}\}_{1 \leq l \leq \nu_k}^{i \in \mathbb{I}_k}\}_{k=1}^m$ ,  $\{b_k\}_{k=0}^m \subset \mathbb{R}$ ,  $\{\{\boldsymbol{\gamma}_{i,k}\}_{i \in \mathbb{I}_k}\}_{k=1}^m \subset \mathbb{R}^d$ ,  $\mathbf{g}_0 \in \mathbb{R}^d$  solution to

$$s_\varepsilon^* := \min \sum_{k=1}^m \sum_{l=1}^{\nu_k} s_{k,l} \quad (8b)$$

$$\text{if } \mathbf{y}_{k,l} \in \mathbf{Y}_0, \quad \sum_{i \in \mathbb{I}_k} \boldsymbol{\gamma}_{i,k}^\top \mathbf{y}_{k,l} = \mathbf{g}_0^\top \mathbf{y}_{k,l} + b_0 - b_k \quad (8b)$$

$$\text{if } \mathbf{y}_{k,l} \in \mathbf{Y}_{k'}, \quad \sum_{i \in \mathbb{I}_k} (\boldsymbol{\gamma}_{i,k} - \boldsymbol{\gamma}_{i,k'})^\top \mathbf{y}_{k,l} = b_{k'} - b_k \quad (13a)$$

$$\forall k \in \mathbb{N}_m, \quad i \in \mathbb{I}_k, \quad l \in \mathbb{N}_{\nu_k}, \quad s_{k,l} \geq -\varepsilon \quad (16a)$$

$$s_{k,l} \geq \sum_{i \in \mathbb{I}_k} \mathbf{f}_i^\top \boldsymbol{\gamma}_{i,k} + z_{i,k,l} \quad (21a)$$

$$\text{and } z_{i,k,l} \geq M \|\boldsymbol{\gamma}_{i,k}\|^* \|\mathbf{y}_{k,l} - \mathbf{x}_i\|. \quad (21b)$$

*Remark. Learning without knowing  $M$ :* It is possible to consider  $M$  as a decision variable, and determine the largest Lipschitz constant for which an LF can be found given a particular set of data. More specifically, constraint (21b) is recast to a positive semi-definite constraint by

$$\begin{bmatrix} MI_d & \gamma_{i,k} \\ \gamma_{i,k}^\top & z_{i,k,l} \end{bmatrix} \in \mathcal{S}^+, \quad (23)$$

where  $\mathcal{S}^+$  denotes the set of positive semi-definite matrices. However, once this formulation is used, the resulting optimization problem becomes an SDP and the data refinement technique proposed at the beginning of this Section 5.2 cannot be applied. Even though the particular sparsity structure in (23) can be exploited to improve the computational efficiency, the computational cost of this optimization still drastically increase in comparison with the SOCP problem 13.

### 5.3 Algorithms

A first learning scheme is summarized in Algorithm 1. Even though we use a standard, non-tailored tessellation methodology in algorithm 1, generating a good tessellation is vital but non-trivial. Existing works mostly focus on the link between a convex liftable tessellation and the power diagram (see e.g. [49,50]). However, as the LF studied in this paper is not necessarily convex, hence we leave the study of this topic in the future research and we use the standard Delaunay triangulation in this work [51].

---

#### Algorithm 1

---

**Input:** RoA prior  $\mathbf{A}$ , negativity tolerance  $-\varepsilon$ , Lipschitz overestimate  $M$

**Output:** LF  $V(\mathbf{x})$ , RoA inner estimate  $\widehat{\mathbf{R}} = \mathbf{L}_a^V$

Refine a tessellation  $\{\mathbf{Y}_k\}_{k=1}^m$  until it satisfies (19)

**if** tessellation is valid **then**

Solve optimization problem 13

**if** Optimal solution satisfies  $s_\varepsilon^* = -\varepsilon \sum_{k=1}^m \nu_k$  **then**

Find the largest  $a$  such that  $\mathbf{L}_a^V \subset \mathbf{X}$

Return RoA estimate  $\widehat{\mathbf{R}} = \mathbf{L}_a^V$

**end if**

**else**

Return cannot learn  $V(\mathbf{x})$ .

**end if**

---

**Proving that  $\mathbf{X}$  is the full RoA** Up to this point, we only assumed knowledge of a subset  $\mathbf{A}$  of the RoA and focused on finding an RoA estimate based on the dataset  $\mathbf{D}$  and the Lipschitz bound  $M$ ,  $\mathbf{X}$  being the admissible state set in which the considered system should evolve. However, in some cases, one could define  $\mathbf{X}$  as the a priori candidate for the RoA estimate. In that case, one wants to certify that the whole  $\mathbf{X}$  is equal to the RoA  $\mathbf{R}$ , by finding a PWA LF  $V(\mathbf{x})$  and level value  $a \in \mathbb{R}$  such that  $\mathbf{X} = \mathbf{L}_a^V$ . This is done by adding the following level conditions to problem 13 (recalling that  $\mathbf{X}$  is *open*, i.e.  $\mathbf{X} \cap \partial\mathbf{X} = \emptyset$ ):

$$\mathbf{y}_{k,l} \in \partial\mathbf{X} \implies \sum_{i \in \mathbb{I}_k} \gamma_{i,k}^\top \mathbf{y}_{k,l} + b_k = a \quad (24a)$$

$$\mathbf{y}_{k,l} \in \mathbf{X} \implies \sum_{i \in \mathbb{I}_k} \gamma_{i,k}^\top \mathbf{y}_{k,l} + b_k \leq a - \varepsilon. \quad (24b)$$

Based on the solution to the resulting problem, it is possible to determine an RoA estimate using the following Corollary:

**Corollary 14.** *Under Assumption V and tessellation given by (7), for any  $\mathbf{f}$  satisfying Assumptions I, II, IV, and VII, if the solution to problem 13 further satisfies (17) i.e.  $s_\varepsilon^* = -\varepsilon \sum_{k=1}^m \nu_k$ , as well as level condition (24) for some  $a \in \mathbb{R}$ , then  $\mathbf{X} = \mathbf{A}$  i.e.*

$$\forall \mathbf{x}_0 \in \mathbf{X}, \quad \lim_{t \rightarrow \infty} \|\mathbf{x}(t|\mathbf{x}_0)\| = 0.$$

*Proof.* Trivially follows from Proposition 10 applied to  $\mathbf{L}_a^V \stackrel{(24)}{=} \mathbf{X}$ .  $\square$

With this corollary, we are able to certify whether  $\mathbf{X}$  is the full RoA, by applying the scheme summarized in Algorithm 2.

---

**Algorithm 2**


---

**Input:** RoA prior  $\mathbf{A}$ , negativity tolerance  $-\varepsilon$ , Lipschitz overestimate  $M$ , set level  $a$

**Output:** LF  $V(\mathbf{x})$

Refine a tessellation  $\{\mathbf{Y}_k\}_{k=1}^m$  until it satisfies (19)

**if** tessellation is valid **then**

Solve optimization problem 13 with constraint (24)

**if** Optimal value solution satisfies  $s_\varepsilon^* = -\varepsilon \sum_{k=1}^m \nu_k$  **then**

Return

**end if**

**else**

Return cannot learn  $V(\mathbf{x})$ .

**end if**

---

It is possible to sequentially decompose the learning of the RoA: in addition to increasing the size of the RoA prior  $\mathbf{A}$ , we propose to sequentially increase the size of the RoA candidate  $\mathbf{X}$ , i.e. we come up with a sequence  $(\mathbf{X}_r)_{1 \leq r \leq r_{\max}}$  of RoA candidates, such that

$$\mathbf{A} \subset \mathbf{X}_1 \subset \mathbf{X}_2 \dots \subset \mathbf{X}_{r_{\max}} = \mathbf{X}.$$

Now, at iteration  $r$ , we prove that  $\mathbf{X}_r$  is positively invariant and attracted to  $\mathbf{0}$ , after which we set  $\mathbf{A} = \mathbf{X}_r$  and work on  $\mathbf{X}_{r+1}$ ; to do so, it is only required to learn the LF on the set  $\mathbf{X}_r \setminus \mathbf{A}$ , as summarised in Algorithm 3. At the end of the iteration process, and if the algorithm did not fail, one gets an algorithmic proof that  $\mathbf{X}_{r_{\max}} = \mathbf{X} = \mathbf{R}$ . If the algorithm failed at iteration  $r$ , then one still gets an algorithmic proof that  $\mathbf{X}_{r-1} \subset \mathbf{R}$  (with convention  $\mathbf{X}_0 = \mathbf{A}$  at the first iteration). It is noteworthy that, if one needs to recover the whole LF on  $\mathbf{X} \setminus \mathbf{A}$ , then it is necessary to impose the continuity condition on the boundary of  $\partial \mathbf{X}_r$  between the  $r$ -th iteration and the  $(r+1)$ -th iteration, as the tessellation of  $\mathbf{X} \setminus \mathbf{A}$  is the union of tessellations of the  $\mathbf{X}_{r+1} \setminus \mathbf{X}_r$ , so that the continuity condition (8b) applies on the boundary  $\partial \mathbf{X}_r$ .

## 6 Numerical Results

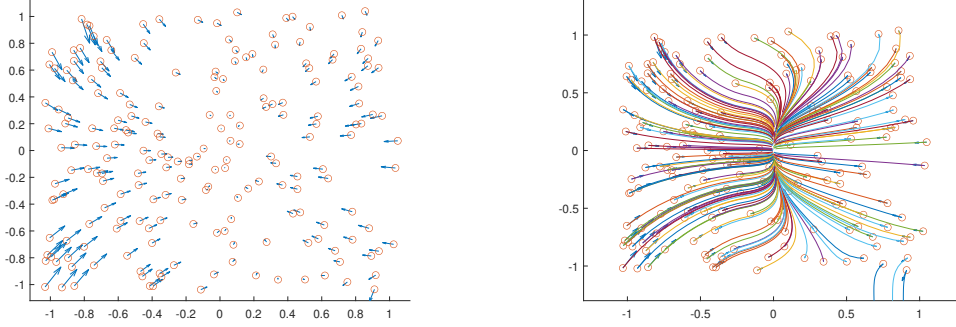
In this part, the proposed learning schemes are evaluated in two different examples. In particular, we will make use of Algorithm 1, 2 and 3 in the first example and use Algorithm 1 in the second one. All the following results are implemented on a laptop with Intel i7-11800H and 32G memory, and the Mosek numerical solver is used to get optimal solutions of the SOCP problems.



**Algorithm 3****Input:** RoA prior  $\mathbf{A}$ , RoA candidates  $(\mathbf{X}_r)_{1 \leq r \leq r_{\max}}$ **Output:** LF  $V(\mathbf{x})$ , RoA inner estimate  $\widehat{\mathbf{R}} = \mathbf{L}_a^V$  $r = 1$ **while**  $\{r \leq r_{\max}\}$  **do**  Run Algorithm. 2 with  $\mathbf{X} = \mathbf{X}_r$   **if** Algorithm 2 failed **then**    Return cannot learn  $V(\mathbf{x})$ .  **end if**   $\mathbf{A} \leftarrow \mathbf{X}_r$    $r \leftarrow r + 1$ **end while****6.1 Non Polynomial dynamic system**

We consider a two-dimensional nonlinear dynamic system:

$$\begin{aligned} \dot{x}_1(t) &= -0.9 \sin(x_1(t)) \cos(x_2(t)) + 0.2x_1(t)x_2(t) + 0.25x_2(t)^2 \\ \dot{x}_2(t) &= -1 \sin(x_2(t))(|x_1(t) + 0.2|) + 0.5 \frac{x_1(t)x_2(t)}{\cos(x_2(t)) - 0.3x_1(t)} \end{aligned}$$



(a) The data used for the learning scheme.

(b) Underlying dynamics (not used for learning).

Figure 2: Representations of the considered nonpolynomial dynamic system.

We assume that we know an RoA prior  $\mathbf{A} = [-0.1, 0.1]^2$ . A dataset with only 200 samples within  $[-1, 1]^2 \subset \mathbb{R}^2$  is used to learn the underlying LF: the positions  $\{\mathbf{x}_i\}$  and speeds  $\{\mathbf{f}_i\}$  of these samples are plotted in Figure 2a, from which we can observe that this dataset is relatively sparse in  $[-1, 1]^2$ . Judging by the speed sample, the dynamic system seems stable within the box  $[-0.4, 0.4]^2$ , while stability within the region  $[-1, 1]^2 \setminus [-0.4, 0.4]^2$  is unclear because of the speed samples in the lower right corner in Figure 2a. Hence, we ran sequential space partition scheme (Algorithm 3). In particular, we first use Algorithm 1 in the region  $[-0.4, 0.4]^2$  with  $\mathbf{A} = [-0.1, 0.1]^2$ . After we justify that  $[-0.4, 0.4]^2$  is a positively invariant subset of the RoA, then we further apply Algorithm 2 to  $[-1, 1]^2$  with  $\mathbf{A} = [-0.4, 0.4]^2$ . In both sub-problems,

the negativity tolerances  $\varepsilon$  are set to  $10^{-3}$  and the tessellation are both randomly generated by Delaunay triangulation [51].

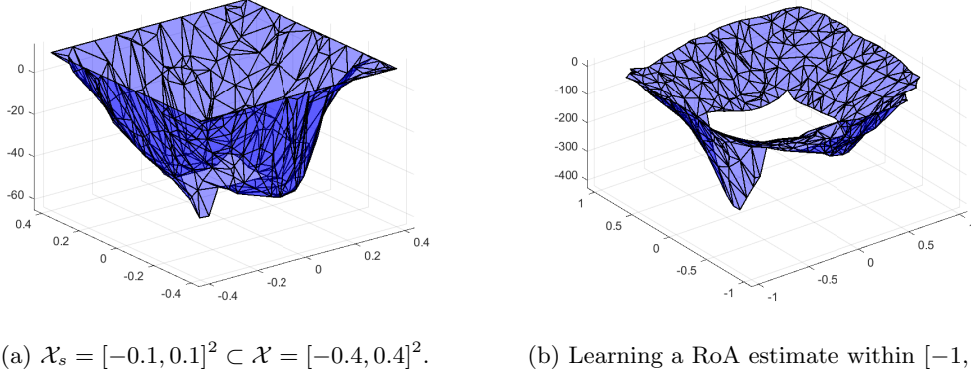


Figure 3: Visualization of the learnt LFs.

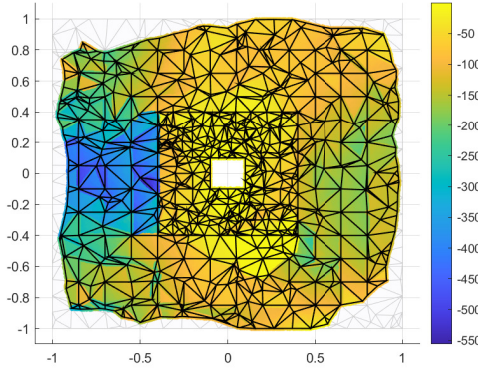


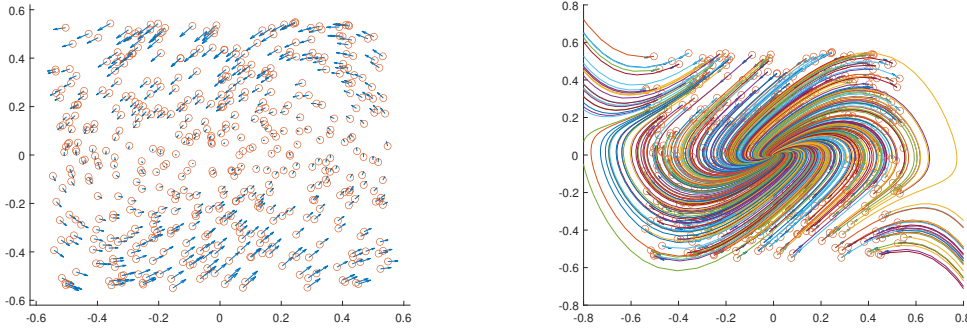
Figure 4: Evaluation of  $\mathbf{f}(x)^\top \partial_C V(\mathbf{x})$  on the learnt RoA, the gray triangulation in the background is the tessellation used to solve Problem 13, while the coloured region in the front is our RoA estimate.

The learnt Lyapunov function in  $[-0.4, 0.4]^2$  is shown in Figure 3a, while the RoA we finally end up with is shown in Figure 4. Moreover, the LF learnt from Algorithm 2 in  $\mathbf{A} = [-1, 1]^2 \setminus [-0.4, 0.4]^2$  is shown in Figure 3b. Figure 4 also shows the evaluation of  $\mathbf{f}(x)^\top \partial_C V(\mathbf{x})$  with respect to the underlying dynamic system, whose maximal evaluation is  $-1.525 \times 10^{-2} < 0$ , as expected from an LF. In accordance with our guess, the learnt RoA in Figure 4 cuts off the lower right corner, because this region does not seem to be stable. To see that, we simulate the underlying dynamic system by setting the initial states to points in our dataset. The simulated trajectories are plotted in Figure 2b; please note that these trajectories are not used in the learning scheme at all.

## 6.2 Reverse Time Van Del Pol Oscillator

In this part, we consider the reverse time Van Del Pol oscillator:

$$\begin{aligned}\dot{x}_1(t) &= -2x_2(t) \\ \dot{x}_2(t) &= -0.8x_1(t) - 10(x_1(t)^2 - 0.21)x_2(t).\end{aligned}$$



(a) The data used for the learning scheme. (b) Simulated trajectories (not used for learning).

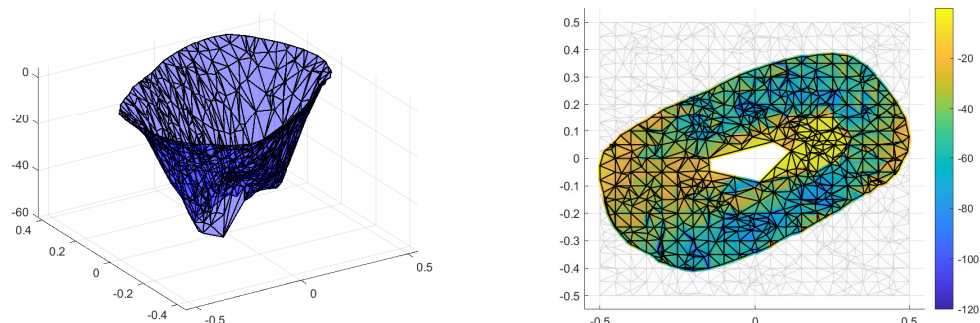
Figure 5: The reverse time Van Del Pol oscillator.

We know an a-priori polytopic RoA  $\mathbf{A}$ , which is plotted in the center of Figure 6b. A dataset with only 400 samples within  $[-0.5, 0.5]^2 \subset \mathbb{R}^2$  is used to learn the underlying LF: the positions  $\{\mathbf{x}_i\}$  and speeds  $\{\mathbf{f}_i\}$  of these samples are plotted in Figure 5a. Similar to what we did in the last example, we simulate these data forward in Figure 5b, while these trajectories are not used in the learning scheme. We can observe that both the lower right corner and the upper left corner in Figure 5b correspond to regions of unstable states. Even with only the access to the data in Figure 5a, we can not give a clear idea about which region is safe, hence we apply Algorithm 2 to  $[-0.5, 0.5]^2$ . In particular, the negativity tolerance  $\varepsilon$  is set to  $10^{-3}$  and the tessellation is randomly generated by Delaunay triangulation [51]. The learnt LF and the corresponding evaluation of  $\mathbf{f}(\mathbf{x})^\top \partial_C V(\mathbf{x})$  on the learnt RoA are respectively plotted in Figure 6a and Figure 6b. In particular, the maximal evaluation of  $\mathbf{f}(\mathbf{x})^\top \partial_C V(\mathbf{x})$  on the learnt RoA is  $-1.947 \times 10^{-2} < 0$ .

## 7 Conclusion

The paper proves a variant of a stability theorem with non-smooth Lyapunov functions (LF) and then implements an algorithm for data-based region of attraction (RoA) estimation of an unknown dynamic system. Through this process, a theorem for piecewise affine (PWA) LF computation was proven, robustly distributing the global dataset information and deriving a convex optimization program for computing such Lyapunov functions.

The originality of the method is that it only requires a fixed dataset to compute an estimate of the RoA, from which it allows the user to deduce global information from local data and knowledge of the Lipschitz constant of the unknown dynamic system. In more detail, the technique (1) decouples the PWA representation from the data point locations, and (2) decomposes the



(a) Visualization of the LF.

(b) Evaluation of  $\mathbf{f}(\mathbf{x})^\top \partial_C V(\mathbf{x})$  on the learnt RoA.

Figure 6: The Lyapunov function learnt from a Van der Pol dataset.

On Figure 6b, the gray triangulation in the background is the tessellation used to solve Problem 13, while the coloured region in the front is the RoA estimate. The polytopic hole in the middle represents the RoA prior  $\mathbf{A}$ .

corresponding uncertainty set containing the unknown dynamics as an intersection of sets, each one depending on a single data point, in order to derive a tractable criterion accounting for the global dataset. Hence, it can be used to study systems whose dynamics cannot be easily sampled at will, through a relatively simple optimization problem that can be handled with interior point methods.

## References

- [1] A. M. Lyapunov, “The general problem of the stability of motion,” *International journal of control*, vol. 55, no. 3, pp. 531–534, 1992.
- [2] M. Hertneck, J. Köhler, S. Trimpe, and F. Allgöwer, “Learning an approximate model predictive controller with guarantees,” *IEEE Control Systems Letters*, vol. 2, no. 3, pp. 543–548, 2018.
- [3] S. Gao, S. Kong, and E. M. Clarke, “dreal: An smt solver for nonlinear theories over the reals,” in *International conference on automated deduction*, pp. 208–214, Springer, 2013.
- [4] Y.-C. Chang, N. Roohi, and S. Gao, “Neural lyapunov control,” *Advances in neural information processing systems*, vol. 32, 2019.
- [5] H. Dai, B. Landry, M. Pavone, and R. Tedrake, “Counter-example guided synthesis of neural network lyapunov functions for piecewise linear systems,” in *2020 59th IEEE Conference on Decision and Control (CDC)*, pp. 1274–1281, IEEE, 2020.
- [6] J. Kapinski, J. V. Deshmukh, S. Sankaranarayanan, and N. Aréchiga, “Simulation-guided lyapunov analysis for hybrid dynamical systems,” in *Proceedings of the 17th international conference on Hybrid systems: computation and control*, pp. 133–142, 2014.

- [7] S. Gao, J. Avigad, and E. M. Clarke, “ $\delta$ -complete decision procedures for satisfiability over the reals,” in *International Joint Conference on Automated Reasoning*, pp. 286–300, Springer, 2012.
- [8] V. I. Zubov, *Methods of AM Lyapunov and their application*. P. Noordhoff, 1964.
- [9] P. Giesl, *Construction of global Lyapunov functions using radial basis functions*, vol. 1904. Springer, 2007.
- [10] P. A. Parrilo, *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. California Institute of Technology, 2000.
- [11] A. Oustry, M. Tacchi, and D. Henrion, “Inner approximations of the maximal positively invariant set for polynomial dynamical systems,” *IEEE Control Systems Letters*, vol. 3, no. 3, pp. 733–738, 2019.
- [12] D. Henrion and M. Korda, “Convex computation of the region of attraction of polynomial control systems,” *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 297–312, 2013.
- [13] M. Tacchi, C. Cardozo, D. Henrion, and J. B. Lasserre, “Approximating regions of attraction of a sparse polynomial differential system,” *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 3266–3271, 2020.
- [14] A. A. Ahmadi and A. Majumdar, “Dsos and sdsos optimization: more tractable alternatives to sum of squares and semidefinite optimization,” *SIAM Journal on Applied Algebra and Geometry*, vol. 3, no. 2, pp. 193–230, 2019.
- [15] Z. Sun, “Stability of piecewise linear systems revisited,” *Annual Reviews in Control*, vol. 34, no. 2, pp. 221–231, 2010.
- [16] H. Lin and P. J. Antsaklis, “Stability and stabilizability of switched linear systems: a survey of recent results,” *IEEE Transactions on Automatic control*, vol. 54, no. 2, pp. 308–322, 2009.
- [17] A. Alessio and A. Bemporad, “A survey on explicit model predictive control,” in *Nonlinear model predictive control*, pp. 345–369, Springer, 2009.
- [18] M. Johansson and A. Rantzer, “Computation of piecewise quadratic lyapunov functions for hybrid systems,” in *1997 European Control Conference (ECC)*, pp. 2005–2010, IEEE, 1997.
- [19] H. Ravanbakhsh and S. Sankaranarayanan, “Learning control lyapunov functions from counterexamples and demonstrations,” *Autonomous Robots*, vol. 43, no. 2, pp. 275–307, 2019.
- [20] R. Schwan, C. N. Jones, and D. Kuhn, “Stability verification of neural network controllers using mixed-integer programming,” *arXiv preprint arXiv:2206.13374*, 2022.
- [21] A. Solar-Lezama, L. Tancau, R. Bodik, S. Seshia, and V. Saraswat, “Combinatorial sketching for finite programs,” in *Proceedings of the 12th international conference on Architectural support for programming languages and operating systems*, pp. 404–415, 2006.
- [22] A. Solar-Lezama, *Program synthesis by sketching*. University of California, Berkeley, 2008.
- [23] S. Chen, M. Fazlyab, M. Morari, G. J. Pappas, and V. M. Preciado, “Learning lyapunov functions for hybrid systems,” in *Proceedings of the 24th International Conference on Hybrid Systems: Computation and Control*, pp. 1–11, 2021.

- [24] A. Abate, D. Ahmed, M. Giacobbe, and A. Peruffo, “Formal synthesis of lyapunov neural networks,” *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 773–778, 2020.
- [25] T. A. Johansen, “Computation of lyapunov functions for smooth nonlinear systems using convex optimization,” *Automatica*, vol. 36, no. 11, pp. 1617–1626, 2000.
- [26] P. Giesl, B. Hamzi, M. Rasmussen, and K. N. Webster, “Approximation of lyapunov functions from noisy data,” *arXiv preprint arXiv:1601.01568*, 2016.
- [27] S. F. Marinósson, “Lyapunov function construction for ordinary differential equations with linear programming,” *Dynamical Systems: An International Journal*, vol. 17, no. 2, pp. 137–150, 2002.
- [28] P. M. Julian, *A high-level canonical piecewise linear representation: Theory and applications*. PhD thesis, Universidad Nacional del Sur (Argentina), 1999.
- [29] R. Baier, L. Grüne, and S. F. Hafstein, “Linear programming based lyapunov function computation for differential inclusions,” *Discrete & Continuous Dynamical Systems-B*, vol. 17, no. 1, p. 33, 2012.
- [30] S. F. Hafstein, “An algorithm for constructing lyapunov functions.,” *Electronic Journal of Differential Equations*, vol. 2007, 2007.
- [31] N. Boffi, S. Tu, N. Matni, J.-J. Slotine, and V. Sindhwan, “Learning stability certificates from data,” *Proceedings of Machine Learning Research*, vol. 155, pp. 1341–1350, 2021.
- [32] T. Martin and F. Allgöwer, “Data-driven system analysis of nonlinear systems using polynomial approximation,” *IEEE Transactions on Automatic Control*, 2023.
- [33] M. Tacchi, B. Marinescu, M. Anghel, S. Kundu, and C. Cardozo, “Power system transient stability analysis using sum of squares programming,” in *Power Systems Computation Conference*, 2018.
- [34] P. Scharnhorst, E. T. Maddalena, Y. Jiang, and C. N. Jones, “Robust uncertainty bounds in reproducing kernel hilbert spaces: A convex optimization approach,” *arXiv preprint arXiv:2104.09582*, 2021.
- [35] A. A. Ahmadi, M. Krstić, and P. Parrilo, “A globally asymptotically stable polynomial vector field with no polynomial Lyapunov function,” in *IEEE Conference on Decision and Control*, 2011.
- [36] J. M. Montenbruck and F. Allgöwer, “Some problems arising in controller design from big data via input-output methods,” in *IEEE Conference on Decision and Control*, 2016.
- [37] F. H. Clarke, *Optimization and Nonsmooth Analysis*. Classics in Applied Mathematics, SIAM, 1990.
- [38] J. P. La Salle, *The stability of dynamical systems*. SIAM, 1976.
- [39] M. Della Rossa, *Non-Smooth Lyapunov Functions for Stability Analysis of Hybrid Systems*. PhD thesis, Institut National des Sciences Appliquées de Toulouse, 2020.
- [40] H. L. Royden and P. Fitzpatrick, *Real analysis*, vol. 32. Macmillan New York, 1988.
- [41] M. D. Kirszbraun, “Über die zusammenziehende und Lipschitzsche transformationen,” *Fundamenta Mathematicae*, vol. 22, pp. 77–108, 1934.

- [42] J.-P. Calliess, S. J. Roberts, C. E. Rasmussen, and J. Maciejowski, “Lazily adapted constant kinky inference for nonparametric regression and model-reference adaptive control,” *Automatica*, vol. 122, p. 109216, 2020.
- [43] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, vol. 2. MIT press Cambridge, MA, 2006.
- [44] L. Sabug Jr, F. Ruiz, and L. Fagiano, “Smgo: A set membership approach to data-driven global optimization,” *Automatica*, vol. 133, p. 109890, 2021.
- [45] M. Canale, L. Fagiano, and M. Milanese, “Set membership approximation theory for fast implementation of model predictive control laws,” *Automatica*, vol. 45, no. 1, pp. 45–54, 2009.
- [46] Y. Nesterov and A. Nemirovskii, *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.
- [47] H. Krieg, *Modeling and solution of continuous set covering problems by means of semi-infinite optimization*. Fraunhofer Verlag, 2019.
- [48] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [49] K. Rybnikov, *Polyhedral partitions and stresses*. Queen’s University at Kingston, 2000.
- [50] F. Aurenhammer, “Power diagrams: properties, algorithms and applications,” *SIAM Journal on Computing*, vol. 16, no. 1, pp. 78–96, 1987.
- [51] B. Delaunay *et al.*, “Sur la sphere vide,” *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, vol. 7, no. 793-800, pp. 1–2, 1934.

## A Proof of Theorem 8

*Theorem 8:* Let Assumptions I, II, IV, V and VII hold, and consider the function  $V$  defined by (7), (8). For  $k \in \mathbb{N}_m$ , let  $\{\mathbf{y}_{k,l}\}_{l=1}^{\nu_k}$  be the set of vertices of  $\mathbf{Y}_k$ , so that

$$\mathbf{Y}_k = \text{co}\{\mathbf{y}_{k,l}\}_{l=1}^{\nu_k}. \quad (15a)$$

Suppose that  $\forall k \in \mathbb{N}_m$ , there is a set  $\{\gamma_{i,k}\}_{i=1}^n \subset \mathbb{R}^d$  with

$$\sum_{i=1}^n \gamma_{i,k} = \mathbf{g}_k \quad (15b)$$

and for all  $l \in \mathbb{N}_{\nu_k}$

$$\sum_{i=1}^n \mathbf{f}_i^\top \gamma_{i,k} + M \|\gamma_{i,k}\|^* \|\mathbf{x}_i - \mathbf{y}_{k,l}\| < 0. \quad (15c)$$

Then,  $\forall a \in \mathbb{R}$  such that  $\mathbf{L}_a^V \subset \mathbf{X}$ , it holds  $\mathbf{L}_a^V \subset \mathbf{R}$ , i.e.

$$\forall \mathbf{x}_0 \in \mathbf{L}_a^V, \quad \lim_{t \rightarrow \infty} \|\mathbf{x}(t|\mathbf{x}_0)\| = 0. \quad (6)$$

*Proof.* By Assumption VII, for all  $\mathbf{x} \in \mathbf{X}$ ,  $i \in \mathbb{N}_n$ , it holds  $\|\mathbf{f}(\mathbf{x}) - \mathbf{f}_i\| \leq M\|\mathbf{x} - \mathbf{x}_i\|$ , i.e.  $\mathbf{f}(\mathbf{x}) \in \mathbf{B}(\mathbf{f}_i, M\|\mathbf{x} - \mathbf{x}_i\|) =: \mathbf{F}_i(\mathbf{x})$ . In particular,

$$\mathbf{f}(\mathbf{x}) \in \mathbf{F}(\mathbf{x}) = \bigcap_{i=1}^n \mathbf{F}_i(\mathbf{x}),$$

so that Assumptions III and VI hold. Hence, according to Proposition 6 and Theorem 7, with  $\Omega = \{\mathbf{x}_i\}_{i=1}^n$ , the proof of our statement boils down to verifying (13b), i.e.

$$\sum_{i=1}^n \max_{\mathbf{f} \in \mathbf{F}_i(\mathbf{Y}_k)} \mathbf{f}^\top \gamma_{i,k} < 0.$$

Let  $i \in \mathbb{N}_n$ ,  $k \in \mathbb{N}_m$ ,  $\mathbf{f} \in \mathbf{F}_i(\mathbf{Y}_k)$ . By construction, there exists an  $\mathbf{x} \in \mathbf{Y}_k$  such that  $\mathbf{f} \in \mathbf{F}_i(\mathbf{x})$ , i.e.  $\|\mathbf{f} - \mathbf{f}_i\| \leq M\|\mathbf{x} - \mathbf{x}_i\|$ , and the Hölder inequality  $\forall \phi, \psi \in \mathbb{R}^d, |\phi^\top \psi| \leq \|\phi\|^* \|\psi\|$ , yields

$$\begin{aligned} \mathbf{f}^\top \gamma_{i,k} &= \mathbf{f}_i^\top \gamma_{i,k} + \gamma_{i,k}^\top (\mathbf{f} - \mathbf{f}_i) \\ &\leq \mathbf{f}_i^\top \gamma_{i,k} + \|\gamma_{i,k}\|^* \|\mathbf{f} - \mathbf{f}_i\| \\ &\leq \mathbf{f}_i^\top \gamma_{i,k} + M\|\gamma_{i,k}\|^* \|\mathbf{x} - \mathbf{x}_i\|. \end{aligned}$$

Taking the maximum over  $\mathbf{f} \in \mathbf{F}_i(\mathbf{Y}_k)$ , one gets

$$\max_{\mathbf{f} \in \mathbf{F}_i(\mathbf{Y}_k)} \mathbf{f}^\top \gamma_{i,k} \leq \mathbf{f}_i^\top \gamma_{i,k} + M\|\gamma_{i,k}\|^* \left( \max_{\mathbf{x} \in \mathbf{Y}_k} \|\mathbf{x}_i - \mathbf{x}\| \right).$$

It remains to show that

$$\max_{\mathbf{x} \in \mathbf{Y}_k} \|\mathbf{x}_i - \mathbf{x}\| = \max_{1 \leq l \leq \nu_k} \|\mathbf{x}_i - \mathbf{y}_{k,l}\|,$$

which follows from the convexity of the objective function  $\mathbf{x} \mapsto \|\mathbf{x}_i - \mathbf{x}\|$  together with the convexity of  $\mathbf{Y}_k$ : the maximum of the optimization problem on the left hand side is attained on an extreme point of  $\mathbf{Y}_k$ , i.e. a  $\mathbf{y}_{k,l}$ . Finally, summing over  $i$  yields

$$\begin{aligned} \sum_{i=1}^n \max_{\mathbf{f} \in \mathbf{F}_i(\mathbf{Y}_k)} \mathbf{f}^\top \gamma_{i,k} &\leq \max_{1 \leq l \leq \nu_k} \sum_{i=1}^n \mathbf{f}_i^\top \gamma_{i,k} \\ &\quad + M\|\gamma_{i,k}\|^* \|\mathbf{x}_i - \mathbf{y}_{k,l}\| \stackrel{(15c)}{<} 0. \end{aligned}$$

□

*Remark.* If instead of Assumption VII, we suppose that  $\|\mathbf{f}\|_{\mathcal{H}^d} \leq M$  for some RKHS  $\mathcal{H}$  with kernel  $\kappa$ , then the reasoning given in the Example of Section 2.1 yields an uncertainty set

$$\mathbf{F}_i(\mathbf{x}) = \frac{\kappa(\mathbf{x}, \mathbf{x}_i)}{\kappa(\mathbf{x}_i, \mathbf{x}_i)} \mathbf{f}_i + \underbrace{\sqrt{\kappa(\mathbf{x}, \mathbf{x}) - \frac{\kappa(\mathbf{x}, \mathbf{x}_i)^2}{\kappa(\mathbf{x}_i, \mathbf{x}_i)}}}_{P_i(\mathbf{x})} [\pm \boldsymbol{\mu}_i]$$

for some fixed vector  $\boldsymbol{\mu}_i \in \mathbb{R}^d$  depending on  $M$  and the representer  $\hat{\mathbf{f}}$  of  $\mathbf{f}$  learnt from the single data  $\{\mathbf{x}_i, \mathbf{f}_i\}$  (but not on  $\mathbf{x}$ ), so that in the previous proof one has to replace  $M\|\mathbf{x} - \mathbf{x}_i\|$  with  $|\kappa(\mathbf{x}, \mathbf{x}_i)/\kappa(\mathbf{x}_i, \mathbf{x}_i) - 1| \cdot \|\mathbf{f}_i\| + 2P_i(\mathbf{x})\|\boldsymbol{\mu}_i\|$  (using the triangle inequality to bound  $\|\mathbf{f} - \mathbf{f}_i\|$  for  $\mathbf{f} \in \mathbf{F}_i(\mathbf{x})$  as above). However, the problem here is that this new function of  $\mathbf{x}$  is not convex: its maximum is not necessarily attained on a vertex  $\mathbf{y}_{k,l}$  of  $\mathbf{Y}_k$ , and we would need other arguments (out of the scope of this article) to obtain a tractable constraint.



## B Assessing conservatism

One can see from the proof of Theorem 7 that replacing (5) with (13) induces some conservatism. This subsection is devoted to proving that, in the special case of Lipschitz bounds, this conservatism ultimately vanishes when the data points and tessellation vertices form an appropriate covering of  $\mathbf{X} \setminus \mathbf{A}$ .

**Proposition 15.** *Let  $V : \mathbb{R}^d \rightarrow \mathbb{R}$  be a continuously differentiable LF with  $V(\mathbf{0}) = 0$ ,  $V > 0$  and  $\dot{V}_f < 0$  on  $\mathbf{X} \setminus \{\mathbf{0}\}$  (which is proved to exist under Assumption I) and suppose that there is an  $a \in \mathbb{R}$  with  $\mathbf{A} \subset \mathbf{L}_a^V \subset \mathbf{X}$ .*

*Under Assumptions II, IV, V and VII and up to adding samples to our dataset  $\mathbf{D}$ ,  $V$  can be approximated by a continuous PWA function  $\hat{V}$  whose parameters in the sense of (8) are solution to Problem 9 and consistent with condition (17), so that  $\hat{V}$  defines a generalized LF as per Proposition 10.*

*Proof.* Let  $\delta > 0$ . Using the universal approximation property of PWA functions [40, Chapter 7.4], there exists a tessellation (7) and a PWA function  $\hat{V}$  as in (8) such that for all  $k \in \{0, \dots, m\}$ ,  $\mathbf{x} \in \mathbf{Y}_k$ , it holds

$$|V(\mathbf{x}) - \mathbf{g}_k^\top \mathbf{x} - b_k| < \delta, \quad |\dot{V}_f(\mathbf{x}) - \mathbf{f}(\mathbf{x})^\top \mathbf{g}_k| < \delta. \quad (25a)$$

We choose  $\delta > 0$  small enough such that

$$\beta := - \max_{\mathbf{x} \in \mathbf{X} \setminus \mathbf{A}} \dot{V}_f(\mathbf{x}) - \delta > 0. \quad (25b)$$

Let  $\eta > 0$ . Up to adding samples to our dataset  $\mathbf{D}$  and further partitioning the tessellation cells  $\mathbf{Y}_k$ , we assume that

$$\mathbf{X} \setminus \mathbf{A} \subset \bigcup_{i=1}^n \mathbf{B}(\mathbf{x}_i, \eta/M) \quad (25c)$$

$$\forall k \in \mathbb{N}_m, \quad \max_{\mathbf{x}, \mathbf{y} \in \mathbf{Y}_k} \|\mathbf{x} - \mathbf{y}\| < \frac{\eta}{M} \quad (25d)$$

Let  $k \in \mathbb{N}_m$  and  $\mathbf{x} \in \mathbf{Y}_k$ . By (25c), there exists an  $i_k \in \mathbb{N}_n$  such that  $\|\mathbf{x}_{i_k} - \mathbf{x}\| < \eta/M$ . As a result, it holds

$$\begin{aligned} \mathbf{f}_{i_k}^\top \mathbf{g}_k &\leq \mathbf{f}(\mathbf{x})^\top \mathbf{g}_k + \|\mathbf{g}_k\|^* \|\mathbf{f}_{i_k} - \mathbf{f}(\mathbf{x})\| \\ &\stackrel{(25a)}{\leq} \dot{V}_f(\mathbf{x}) + \delta + \|\mathbf{g}_k\|^* \|\mathbf{f}_{i_k} - \mathbf{f}(\mathbf{x})\| \\ &\stackrel{(25b)}{\leq} \|\mathbf{g}_k\|^* \|\mathbf{f}_{i_k} - \mathbf{f}(\mathbf{x})\| - \beta \\ &\stackrel{(14)}{\leq} M \|\mathbf{g}_k\|^* \|\mathbf{x}_{i_k} - \mathbf{x}\| - \beta \\ &\stackrel{(25c)}{\leq} \eta \|\mathbf{g}_k\|^* - \beta. \end{aligned} \quad (25e)$$

We use this to define

$$\gamma_{i,k} = \begin{cases} \mathbf{g}_k & \text{if } i = i_k \\ \mathbf{0} & \text{else.} \end{cases}$$

Then, it holds for all  $l \in \mathbb{N}_{\nu_k}$

$$\begin{aligned}
& \sum_{i=1}^n \mathbf{f}_i^\top \boldsymbol{\gamma}_{i,k} + M \|\boldsymbol{\gamma}_{i,k}\|^* \|\mathbf{y}_{k,l} - \mathbf{x}_i\| \\
&= \mathbf{f}_{i_k}^\top \mathbf{g}_k + M \|\mathbf{g}_k\|^* \|\mathbf{y}_{k,l} - \mathbf{x}_{i_k}\| \\
&\stackrel{(25e)}{\leq} \|\mathbf{g}_k\|^* (\eta + M \|\mathbf{y}_{k,l} - \mathbf{x}_{i_k}\|) - \beta \\
&\stackrel{(*)}{\leq} 3\eta \|\mathbf{g}_k\|^* - \beta,
\end{aligned} \tag{25f}$$

where  $(*)$  is obtained through the simple decomposition

$$\begin{aligned}
\|\mathbf{y}_{k,l} - \mathbf{x}_{i_k}\| &\leq \|\mathbf{y}_{k,l} - \mathbf{x}\| + \|\mathbf{x} - \mathbf{x}_{i_k}\| \\
&\stackrel{(25c)}{\leq} \|\mathbf{y}_{k,l} - \mathbf{x}\| + \eta/M \stackrel{(25d)}{\leq} 2\eta/M
\end{aligned}$$

In particular, setting

$$\eta = \frac{\varepsilon + \beta}{3 \max_{k \in \mathbb{N}_m} \|\mathbf{g}_k\|^*}$$

yields, for all  $k \in \mathbb{N}_m$ ,  $l \in \mathbb{N}_{\nu_k}$ ,

$$\sum_{i=1}^n \mathbf{f}_i^\top \boldsymbol{\gamma}_{i,k} + M \|\boldsymbol{\gamma}_{i,k}\|^* \|\mathbf{y}_{k,l} - \mathbf{x}_i\| \stackrel{(25f)}{\leq} -\varepsilon$$

so that our choice for  $\{b_k\}_{k=0}^m$ ,  $\{\boldsymbol{\gamma}_{i,k}\}_{1 \leq i \leq n}^{1 \leq k \leq m}$ ,  $\mathbf{g}_0$  and  $\{\{s_{k,l}\}_{l=1}^{\nu_k}\}_{k=1}^m = \{-\varepsilon\}$  yields a feasible solution to Problem 9.  $\square$