Trajectory-based Barrier Certificates for Monotone Systems: Extended Version

Felipe Galarza-Jimenez, Majid Zamani, and Saber Jafarpour*†

Abstract

Barrier certificates are a powerful tool for providing safety guarantees in dynamical systems. However, most existing approaches for constructing barrier certificates either require a complete system model or rely on extensive simulated data. In many real-world applications, the system model is unknown, and obtaining sampled data of the system dynamics is challenging. Instead, it is often more feasible to access a limited number of system trajectories.

In this work, we propose a data-driven approach to construct barrier certificates for unknown monotone systems using a finite number of system trajectories, while ensuring formal correctness guarantees. Given multiple system trajectories, we construct a family of monotone basis functions that remain non-increasing along all trajectories. Using this class of basis functions, we develop a sampling-based optimization approach to construct barrier certificates, establishing system safety with formal guarantees without requiring additional simulated data or assuming Lipschitz continuity of the system.

Furthermore, for the class of polynomial barrier functions, we show that the scenario-based approach can be efficiently reformulated as a linear program with significantly fewer constraints. Finally, we demonstrate the effectiveness of our approach through numerical examples.

1 Introduction

The increasing prevalence of autonomous systems in safety-critical applications such as transportation, robotics, and industrial automation has intensified the need to provide guarantees for their reliability and safety. Ensuring safety, loosely defined as preventing a system from entering unsafe or undesirable states, is a fundamental requirement in the design and deployment of such systems. Traditional methods for providing safety guarantees include abstraction-based techniques, where a simplified (finite-state) model is used for verification [1], and invariant-based proofs, where safety is ensured by demonstrating that the system remains within a safe set throughout its evolution - as discussed in [2, 3]. However, as systems become more complex and high-dimensional, these analytical approaches may become intractable due to the lack of an accurate model, necessitating data-driven methods for safety verification.

Data-driven techniques have emerged as a practical alternative to traditional model-based verification methods, primarily due to the increasing complexity of modern systems [4]. These techniques can be categorized into two main approaches. The first, an indirect method, relies on collecting system trajectories to reconstruct a model, which is then used to derive a safety certificate. This approach requires that the collected trajectories satisfy the fundamental lemma of persistence of excitation to ensure accurate model reconstruction [5, 6]. The second, a direct method, uses a black box model or a "digital twin" to generate samples of the evolution of the system in different regions of state space, which are then used to build a safety certificate [7, 8, 9]. Although both approaches have shown promise, they present inherent challenges: the indirect method depends on sufficient excitation of the system for effective model reconstruction, which may not be possible or even necessary to synthesize safety certificates, while the direct method is data intensive and often relies on local Lipschitz continuity assumptions to provide 100% safety guarantees [10]. These challenges motivate the exploration of system properties that can mitigate the limitations of existing data-driven methods.

Monotone systems have recently attracted significant attention for their potential to alleviate these challenges. These systems evolve while preserving a partial order, enabling more efficient analysis and certification [11, 12]. Studies have demonstrated that monotonicity can facilitate the

*This work was supported by NSF grants CNS-2039062 and CNS-2111688.

†F. Galarza-Jimenez, M. Zamani, and S. Jafarpour are with the Department of Computer Science at the University of Colorado Boulder, USA. Emails:{felipe.galarzajimenez, majid.zamani, saber.jafarpour}@colorado.edu

computation of stability and reachability certificates with reduced data requirements [13, 14, 15]. Robust forward invariance has also been explored for mixed-monotone systems [16, 17, 18], suggesting the feasibility of safety verification with fewer data points. However, despite these promising developments, the use of monotone system properties in data-driven synthesis of barrier certificates remains unexplored.

Contribution: In this paper, we propose a novel approach for safety verification of unknown monotone discrete-time dynamical systems using only a few collected trajectories. First, using a single trajectory of the system, we construct a *dissipation function* that is non-increasing along the system's evolution. A key property of this function is that its sublevel sets form forward invariant sets for the system. Second, using multiple collected trajectories, we treat their corresponding dissipation functions as a "basis" to establish safety certificates, referred to as *dissipation barrier functions (DBFs)*, for the unknown monotone system. We propose a sufficient condition for recovering a barrier certificate from the DBF. However, due to the data-driven nature of the dissipation functions, this condition cannot be verified using conventional off-the-shelf methods, such as Satisfiability Modulo Theories (SMT) solvers. To overcome this challenge, we develop a data-driven methodology that ensures the correctness of DBFs over compact state sets and provides a structured approach for constructing polynomial DBFs. Finally, we validate our approach on two monotone systems, successfully deriving polynomial DBFs using multiple trajectory data.

2 Notation and Preliminaries

We denote by $\mathbb{N}$ and $\mathbb{R}$ the set of non-negative integers and the set of real numbers, respectively. Given $a \in \mathbb{N}$ ($a \in \mathbb{R}$), we use $\mathbb{N}_{\geq a}$ (resp. $\mathbb{R}_{\geq a}$) to denote all values in $\mathbb{N}$ (resp. $\mathbb{R}$) greater than or equal to a. For $a, b \in \mathbb{R}$, $[a, b]$, $]a, b[$, $[a, b[$, and $]a, b]$ denote closed, open, and half-open interval in $\mathbb{R}$. Likewise, $[a; b]$, $]a; b[$, $[a; b[$ and $]a; b]$ denote closed, open, and half-open sets in $\mathbb{N}$. Given sets $X_1, \ldots, X_n$, for some $n \in \mathbb{N}_{\geq 1}$, we denote the product by $X_1 \times X_2 \times \ldots \times X_n$, or more compactly by $\prod_{j=1}^n X_j$. Moreover, we denote an element $(x_1, \ldots, x_n) \in \prod_{j=1}^n X_j$ of the product set by $\prod_{j=1}^n x_j$, where $x_j \in X_j$ for all $j \in [1; n]$. We define $\mathbb{1}_n := (1, \ldots, 1) \in \mathbb{R}^n$. Given sets A and B, we represent the set difference as $A \backslash B := \{a \in A \mid a \notin B\}$, the Minkowski sum as $A \oplus B := \{a + b \mid a \in A, b \in B\}$, and $|A|$ denotes the cardinality of the set A. We use $f : A \to B$ to denote a function from A to B. We use $f(A)$ to denote the set $\{f(a) \in B \mid \text{ for all } a \in A\}$ and given $b \in B$, we define its preimage by $f^{-1}(b) = \{a \in A \mid f(a) = b\}$. Given a differentiable map $f : \mathbb{R}^n \to \mathbb{R}^m$, its derivative is the map $Df : \mathbb{R}^n \to \mathbb{R}^{m \times n}$ defined by $Df(x)_{ij} = \frac{\partial f_i}{\partial x_j}$ for every $i \in [1; m]$ and $j \in [1; n]$.

2.1 Interval Analysis

Given $x, y \in \mathbb{R}^n$, we write $x \leq y$ ($x \ll y$) if $x_j \leq y_j$ ($x_j < y_j$), for every $j \in [1; n]$. We denote the set $\{z \in \mathbb{R}^n \mid x \leq z \leq y\}$ by $[x, y]$. Given a map $f : \mathbb{R}^n \to \mathbb{R}^m$, the *inclusion function* of f is a map $\mathsf{F} = \big(\underline{\mathsf{F}}, \overline{\mathsf{F}}\big) : \mathbb{R}^{2n} \to \mathbb{R}^{2m}$ such that

$$\underline{\mathsf{F}}(\underline{x}, \overline{x}) \leq f(x) \leq \overline{\mathsf{F}}(\underline{x}, \overline{x}), \qquad \text{for all } x \in [\underline{x}, \overline{x}].$$

For a map $f : \mathbb{R}^n \to \mathbb{R}^m$, there exist different approaches to construct inclusion functions of f. We refer to [19] for more details about inclusion functions and how to construct them.

2.2 Discrete-time Monotone Systems

In this part, we introduce discrete-time monotone systems and characterize their properties.

Definition 1 (Discrete-time Monotone System [11]) *A discrete-time dynamical system $\mathcal{S}$ is a tuple $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$, where $\mathcal{X} \subseteq \mathbb{R}^n$ is the state set, $\mathcal{X}_0 \subseteq \mathcal{X}$ is the initial state set, and $f : \mathcal{X} \to \mathcal{X}$ is the state transition map. The discrete-time dynamical system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ is monotone, if the state transition map f is monotone*[1]*, i.e.,*

$$x \leq y \implies f(x) \leq f(y), \qquad \textit{for all } x, y \in \mathcal{X}. \tag{1}$$

The evolution of the states of a discrete-time system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ is given by

$$x(t+1) = f(x(t)), \tag{2}$$

[1] The results in this manuscript can be extended to monotone systems with respect to any *proper pointed cone* $\mathcal{C}$ that induces a partial order on $\mathbb{R}^n$.

for all $t \in \mathbb{N}$. Given $x_0 \in \mathcal{X}_0$, the trajectory of the discrete-time system $\mathcal{S}$ starting from x_0 is defined by $\mathbf{f}^{(t)}(x_0) := (f^{(0)}(x_0), f^{(1)}(x_0), \ldots)$, with the convention that $f^{(0)}(x_0) = x_0$. For a given $T \in \mathbb{N}$, the T-truncated trajectory of the discrete-time system $\mathcal{S}$ starting from x_0 is defined by $\mathbf{f}_T^{(t)}(x_0) := (f^{(0)}(x_0), \ldots, f^{(T)}(x_0))$. One can show that the discrete-time dynamical system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ is monotone if and only if, for all $x_0, y_0 \in \mathcal{X}$, we have $f^{(t)}(x_0) \leq f^{(t)}(y_0)$, for every $t \geq 0$ [11].

2.3 Safety via Barrier Certificates

Given a discrete-time dynamical system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ and a set of unsafe states $\mathcal{X}_u \subseteq \mathcal{X}$, we say that $\mathcal{S}$ is *safe* if there is no initial condition $x_0 \in \mathcal{X}_0$ and no time $t \in \mathbb{N}$ such that $f^{(t)}(x_0) \in \mathcal{X}_u$. In other words, system $\mathcal{S}$ is safe if starting from *any* x_0, the evolution of $\mathcal{S}$ never enters the unsafe region $\mathcal{X}_u$. One common approach to ensuring the safety of discrete-time systems is the use of barrier certificates [2, 3]. In this paper, we use the following definition of barrier certificates.

Definition 2 (Barrier Certificate) *Consider a system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ and an unsafe set $\mathcal{X}_u \subseteq \mathcal{X}$. Then, a function $\mathbb{B} : \mathcal{X} \to \mathbb{R}$ is called a Barrier certificate if:*

$$\mathbb{B}(x) \leq 0, \qquad \text{for all } x \in \mathcal{X}_0, \tag{3a}$$

$$\mathbb{B}(x) > 0, \qquad \text{for all } x \in \mathcal{X}_u, \tag{3b}$$

$$\mathbb{B}(f(x)) \leq 0, \qquad \text{for every } x \in \mathcal{X} \text{ s.t. } \mathbb{B}(x) \leq 0. \tag{3c}$$

For a discrete-time system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$, it can be shown that the existence of a barrier certificate $\mathbb{B}$ is sufficient and necessary for the existence of an *inductive proof of safety* of the system with respect to the unsafe set $\mathcal{X}_u$ [20]. In some works, condition (3c) is replaced by a more conservative but easier-to-check condition $\mathbb{B}(f(x)) \leq \lambda \mathbb{B}(x)$, for some $\lambda \in \mathbb{R}_{>0}$ uniformly for all $x \in \mathcal{X} \setminus \mathcal{X}_u$.

3 Trajectory-based Safety Certificates for Monotone Systems

In this section, we first formalize the main problem we tackle in this paper. Next, leveraging the information from each collected trajectory of the system, we introduce a scalar *dissipation function* and analyze its properties. Finally, we use this class of dissipation functions as a basis for constructing safety certificates for discrete-time monotone systems.

3.1 Problem Formulation

Consider a monotone discrete-time system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ with a set of unsafe states $\mathcal{X}_u \subseteq \mathcal{X}$. We assume that the state transition map f is *unknown* but have access to data from a few system trajectories, as formalized in the following assumption.

Assumption 1 *For the monotone discrete-time system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$, we have access to N collected T-truncated trajectories of the system $\mathbf{f}_T^{(t)}(x_{0,1})$, $\mathbf{f}_T^{(t)}(x_{0,2})$, $\ldots$, $\mathbf{f}_T^{(t)}(x_{0,N})$ starting from $x_{0,1}$, $x_{0,2}$, $\ldots$, $x_{0,N} \in \mathcal{X}$, respectively.*

Our goal is to utilize these truncated trajectory data to certify the system's safety with respect to the unsafe set $\mathcal{X}_u$.

3.2 A Family of Dissipation Functions

In this section, we use data from each collected trajectory of the monotone discrete-time dynamical system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ to construct a scalar function. We show that this function is non-increasing along the system's evolutions and is monotone on the state set $\mathcal{X}$; thus, we refer to it as a *dissipation function.*

Definition 3 (Dissipation Function) *Consider a monotone system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ with an initial condition $x_0 \in \mathcal{X}$, and a parameter $\alpha \in \mathbb{R}_{>1}$. The dissipation function $P^{x_0} : \mathcal{X} \to \mathbb{R}_{\geq 0}$ is defined by*

$$Q^{x_0}(x, t) = \begin{cases} \frac{1}{t+1} & \text{If } x \leq f^{(t)}(x_0) \\ \alpha & \text{otherwise,} \end{cases} \tag{4a}$$

$$P^{x_0}(x) = \inf_{t \geq 0} \{Q^{x_0}(x, t)\}. \tag{4b}$$

In other words, the map $P^{x_0}(x)$ returns $\frac{1}{t^*+1}$, where t^* is the largest time $t \geq 0$ for which the inequality $x \leq f^{(t)}(x_0)$ holds. If this inequality does not hold for any $t \geq 0$, then $P^{x_0}(x)$ returns α. Notice that for every $x \in \mathcal{X}$, we have $P^{x_0}(x) \in [0, \alpha]$, and the construction of this function relies solely on data from the trajectory of the system $\mathcal{S}$ starting from x_0. This property is fundamental to our data-driven safety certification approach in Section 4. The function P^{x_0} exhibits several important properties, which we present below.

Theorem 1 (Properties of Dissipation Function) *Consider a monotone system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ and let $x_0 \in \mathcal{X}$ and P^{x_0} be the dissipation function defined in (4). Then, the following statements hold:*

1. *P^{x_0} is monotone on $\mathcal{X}$,*
2. *P^{x_0} is non-increasing along evolution of $\mathcal{S}$, i.e., $P^{x_0}(f(x)) \leq P^{x_0}(x)$, for every $x \in \mathcal{X}$,*
3. *for every $c \in [0, \alpha]$, the c-sublevel set*

$$P^{x_0}_{\leq c} := \{x \in \mathcal{X} \mid P^{x_0}(x) \leq c\}$$

is a forward invariance set for $\mathcal{S}$.

Proof: See Section 6.1 in the Appendix. ■

Remark 1 *The following remarks are in order.*

1. *(A data-driven approach) Theorem 1, part (3), establishes that the sublevel sets of the dissipation function P^{x_0} are forward invariant for the monotone system $\mathcal{S}$. Notably, computing P^{x_0}, as defined in (4), requires only the trajectory of $\mathcal{S}$ starting from x_0, i.e., $\mathbf{f}^t(x_0)$. This key property makes dissipation functions particularly suitable for data-driven safety verification.*
2. *(Comparison with literature) For continuous-time systems, similar functions have been used in the proof of [13, Theorem 3.2] to construct max-separable Lyapunov functions on compact sets of globally exponentially stable monotone systems. In contrast, our dissipation function P^{x_0} is more general, as it can be defined for any monotone system over its entire state space.*
3. *The Dissipation Function in (4) is defined due to its properties in Theorem 1. However, other functions also satisfy such properties, for example, scaling $Q^{x_0}(x,t)$ by a $\gamma > 0$ or $\exp(-t)$ if $\leq f^{(t)}(x_0)$.*

The following example provides intuition about the dissipation function for a simple system.

Example 1 *Consider the system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ with $\mathcal{X} = [0,5] \times [0,5]$, singleton initial condition $\mathcal{X}_0 = \{(4.25, 5.0)\}$ and assume the state transition map f is*

$$\begin{aligned} f_1(x) &= 0.9x_1 + 0.2, \\ f_2(x) &= 0.9x_2 + 0.2. \end{aligned} \tag{5}$$

Note that the trajectories of (5) are given by $\mathbf{f}^{(t)}(x_0) = 0.9^t x_0 + 0.2\frac{1-0.9^t}{1-0.9}\mathbb{1}_2$ and that $x^* = (2,2)$ is globally asymptotically stable fixed point of the system. From the trajectory, we define functions $\tilde{t}_j :]2,5] \to \mathbb{R}$, for $j \in [1;2]$, as follows:

$$\begin{aligned} \tilde{t}_j(x_j) &= \log_{0.9}\left(\frac{0.1x_j - 0.2}{0.1x_{0j} - 0.2}\right), \\ Q^{x_0}(x,t) &= \begin{cases} 0 & x \leq (2,2), \\ \frac{1}{t+1} & 0 \leq t \leq \tilde{t}_j, \quad j \in [1;2], \\ \alpha & \text{otherwise}, \end{cases} \\ P^{x_0}(x) &= \inf_{t \geq 0}\{Q^{x_0}(x,t)\}. \end{aligned}$$

Fig. 1 shows the sublevel sets of the dissipation function P^{x_0}. Note that for all $x \leq (2,2)$, we have $P^{x_0}(x) = 0$. Moreover, note that $P^{x_0}(]4.25, 5] \times [0,5]) = \alpha$ since $\tilde{t}_1 < 0$.

In the following section, we provide a way to leverage these functions to verify safety of unknown discrete-time systems.

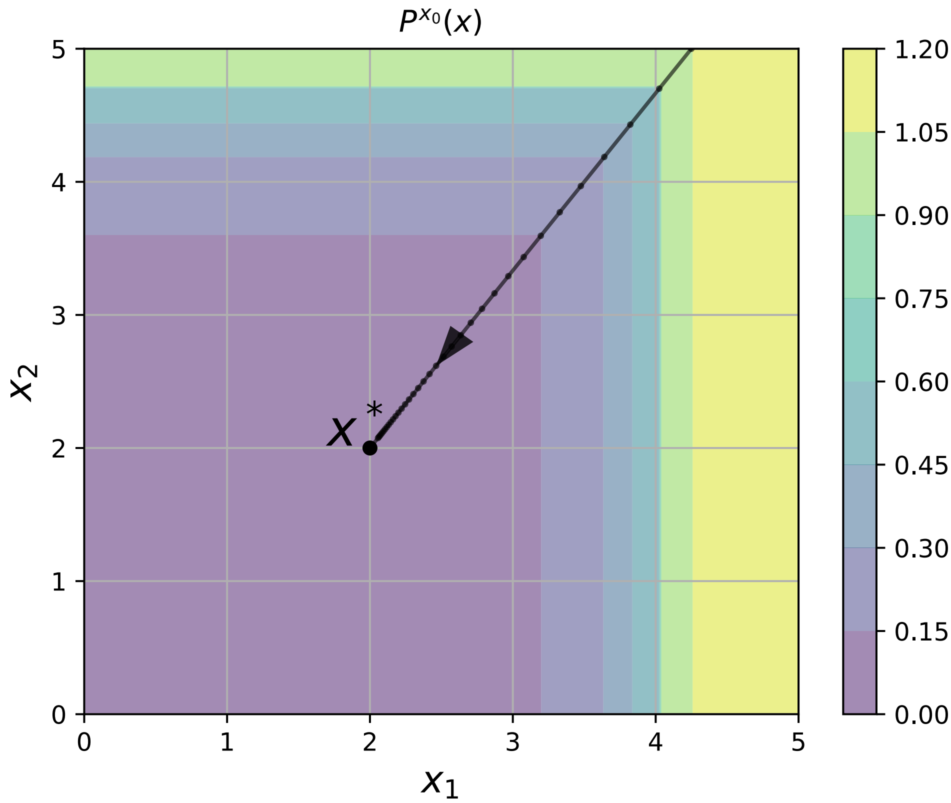

Figure 1: Sublevel sets of P^{x_0} for system in (5) with $x_0 = (4.25, 5.0)$ and $\alpha = 1.1$. The black points on the black line shows the trajectory $\mathbf{f}^{(t)}(x_0)$.

3.3 Trajectory-based Barrier Certificates

In this section, given multiple initial conditions $x_{0,1}, \ldots, x_{0,N} \in \mathcal{X}$, we utilize the dissipation functions $P^{x_{0,k}}$ for all $k \in [1, N]$ as a family of basis functions to construct barrier certificates for safety verification of discrete-time monotone systems. We begin with the following definition.

Definition 4 (Dissipation Barrier Function) *Consider a monotone system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ with an unsafe set $\mathcal{X}_u \subseteq \mathcal{X}$. Suppose $x_{0,1}, \ldots, x_{0,N} \in \mathcal{X}$ and $P^{x_{0,1}}, \ldots, P^{x_{0,N}}$ are their associated dissipation functions as defined in* (4)*. Then, a differentiable function $\mathbb{V} : \mathbb{R}^N \to \mathbb{R}$ is called a Dissipation Barrier Function (DBF) for $\mathcal{S}$ if it satisfies*

$$\mathbb{V}(\overline{P}^{x_{0,k}}(x)) \leq 0, \qquad \textit{for all } x \in \mathcal{X}_0, \tag{6a}$$

$$\mathbb{V}(\overline{P}^{x_{0,k}}(x)) > 0, \qquad \textit{for all } x \in \mathcal{X}_u, \tag{6b}$$

$$D\mathbb{V}(\overline{P}^{x_{0,k}}(x)) \geq \mathbb{0}_N, \qquad \textit{for all } x \in \mathcal{X}, \tag{6c}$$

where $\overline{P}^{x_{0,k}}(x) = (P^{x_{0,1}}(x), \ldots, P^{x_{0,N}}(x))$.

Conditions (6a) and (6b) for the dissipation barrier function $\mathbb{V} : \mathbb{R}^N \to \mathbb{R}$ guarantee a distinct separation between the initial set $\mathcal{X}_0$ and the unsafe set $\mathcal{X}_u$. On the other hand, condition (6c) enforces the *monotonicity* of $\mathbb{V}$ with respect to each of the N dissipation functions. This condition enables us to leverage the properties of the dissipation functions P^{x_0}, as shown in the following result.

Theorem 2 (Dissipation Barrier Certificates) *Consider a monotone system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ with an unsafe set $\mathcal{X}_u \subseteq \mathcal{X}$. Suppose $x_{0,1}, \ldots, x_{0,N} \in \mathcal{X}$ and there exists a dissipation barrier function $\mathbb{V}$ satisfying conditions* (6)*. Then, $\mathcal{S}$ is safe with respect to unsafe set $\mathcal{X}_u$.*

Proof: We show that $\mathbb{B} : \mathcal{X} \to \mathbb{R}$ defined by

$$\mathbb{B}(x) = \mathbb{V}(\overline{P}^{x_{0,k}}(x)), \quad \text{for all } x \in \mathcal{X},$$

is a barrier certificate for $\mathcal{S}$. It can be easily checked that conditions (6a) and (6b) are directly equivalent to (3a) and (3b), respectively. Now, we show that (6c) is a sufficient condition to obtain (3c). Let $x \in \mathcal{X}$ be such that $\mathbb{B}(x) = \mathbb{V}(\overline{P}^{x_{0,k}}(x)) \leq 0$. By Theorem 1 part 2, for every $k \in [1; N]$, we have $P^{x_{0,k}}(f(x)) \leq P^{x_{0,k}}(x)$. Now, we define $u : [0, 1] \to \mathbb{R}^N$ by

$$u(\theta) = \Pi_{k=1}^{N}(\theta P^{x_{0,k}}(x) + (1 - \theta)P^{x_{0,k}}(f(x))).$$

Using condition (6c), for every $\theta \in [0, 1]$, we have $D\mathbb{V}(u(\theta)) \geq \mathbb{0}_n$. Therefore,

$$\mathbb{V}(\overline{P}^{x_{0,k}}(x)) = \mathbb{V}(\overline{P}^{x_{0,k}}(f(x))) + \int_0^1 \sum_{k=1}^{N} [D\mathbb{V}(u(\theta))]_k (P^{x_{0,k}}(x) - P^{x_{0,k}}(f(x)))d\theta. \quad (7)$$

Notice that $P^{x_{0,k}}(x) \geq P^{x_{0,k}}(f(x))$, for every $k \in [1; N]$ and $D\mathbb{V}(u(\theta)) \geq \mathbb{0}_N$, for every $\theta \in [0, 1]$. As a result, inequality (7) implies that

$$\mathbb{B}(f(x)) = \mathbb{V}(\overline{P}^{x_{0,k}}(f(x))) \leq \mathbb{V}(\overline{P}^{x_{0,k}}(x)) \leq 0.$$

This means that condition (3c) holds and the dissipation barrier function in (6) is a barrier certificate.

■

4 Data-Driven Barrier Certificates

Theorem 2 provides a systematic approach for searching for DBFs in monotone discrete-time dynamical systems. However, applying this result to derive DBFs for an unknown monotone system with few collected trajectory data presents two key challenges: (i) computing the dissipation function P^{x_0} requires knowledge of infinitely long trajectories, whereas in practice, collected data from the system's trajectory is of finite length (see Assumption 1). (ii) many off-the-shelf methods, such as SMT solvers, are ineffective for verifying conditions (6) because the dissipation function P^{x_0} lacks a closed-form expression and is computed in a data-driven manner. To overcome these challenges, we propose a data-driven approach that leverages truncated trajectories of the system to compute DBFs. We start with the following assumptions.

Assumption 2 *For a monotone discrete-time dynamical system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$, the state set $\mathcal{X}$ is compact.*

Assumption 3 *Each collected trajectory of the system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ in Assumption 1 converges to a point in $\mathcal{X}$.*

Suppose that the trajectory $\mathbf{f}^{(t)}(x_0)$ of $\mathcal{S}$ converges to a point in $\mathcal{X}$. Then for every $T \geq 0$, there exists $\varepsilon_T \in \mathbb{R}_{>0}$ such that

$$f^{(T)}(x_0) - \varepsilon_T \mathbb{1}_n \leq f^{(t)}(x_0) \leq f^{(T)}(x_0) + \varepsilon_T \mathbb{1}_n, \quad (8)$$

for every $t \in [T; \infty[$. The condition in (8) allows us to address the loss of information due to using T-truncated trajectories in Assumption 1. To show this effect, we introduce T-truncated dissipation function $P_T^{x_0} : \mathcal{X} \to \mathbb{R}_{\geq 0}$ based on $\mathbf{f}_T^t(x_0)$ as follows:

$$Q_T^{x_0}(x, t) = \begin{cases} \frac{1}{t+1} & \text{if } x - \varepsilon_T \mathbb{1}_n \leq f^{(t)}(x_0) \\ \alpha & \text{otherwise.} \end{cases} \quad (9a)$$

$$P_T^{x_0}(x) = \inf_{t \in [0;T]} \{Q_T^{x_0}(x, t)\}, \quad (9b)$$

where ε_T is as defined in (8) and $\alpha \in \mathbb{R}_{>1}$. Notice that the function $P_T^{x_0}$ allows us to use the T-truncated trajectories $\mathbf{f}_T^{(t)}(x_{0,k})$, for $k \in [1; N]$, under Assumption 1. In the following result, we show how this function can be used to provide upper and lower bounds for the dissipation function P^{x_0}.

Theorem 3 (Truncated Dissipation Function) *Consider a monotone discrete-time dynamical system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$. Let $x_0 \in \mathcal{X}$ be such that $\mathbf{f}^{(t)}(x_0)$ converges to a point in $\mathcal{X}$. For every $T \geq 0$, the following statements hold:*

1. $P_T^{x_0}$ *is a monotone function,*

2. *for every* $x \in \mathcal{X}$*,*

$$P_T^{x_0}(x) - \tfrac{1}{T+1} \leq P^{x_0}(x) \leq P_T^{x_0}(x + \varepsilon_T \mathbb{1}_n),$$

where ε_T *is as defined in* (8)*.*

Proof: See Section 6.2 in the Appendix. ■

Theorem 3 provides *error bounds* between the data-driven T-truncated dissipation function $P_T^{x_0}$ and the dissipation function P^{x_0}. Furthermore, note that these bounds become tighter as T increases.

To develop a data-driven approach for synthesizing the DBFs satisfying condition (6), we formulate the search for $\mathbb{V}$ as an optimization problem. First, we fix a parametrized template $\widehat{\mathbb{V}} : \mathcal{P} \times \mathbb{R}^N \to \mathbb{R}$ where $\mathcal{P}$ is the parameter set, e.g., the set of the coefficients of a polynomial of degree d in $\mathbb{R}^N$. Then, we propose the following optimization problem:

$$\textbf{OP} := \min_{p \in \mathcal{P}} \mathsf{Loss}(p, x) \text{ subject to (6)},$$

where $x \in \mathcal{X}$, Loss is a desired loss function, e.g., $||p||_1$, and we replace $\mathbb{V}$ with $\widehat{\mathbb{V}}$ in (6). Notice that solving **OP** is challenging because the dissipation functions $P^{x_{0,k}}$ in condition (6) are computed in a data-driven manner. To address this, we propose using Sampling-Partition Optimization Programming (SpOP). Most existing Scenario Optimization Programming (SOP) approaches (see [4] and references therein) rely on finite randomly-drawn samples of a system's evolution and find a valid solution on the whole state space $\mathcal{X}$ assuming the Lipschitz continuity of the constraints. However, these approaches are not applicable in our case because the dissipation function P^{x_0} is not Lipschitz continuous. Hence, we leverage the concept of the *inclusion function* to design a novel SpOP for finding DBFs of monotone discrete-time systems. We begin by introducing the notion of hyper-rectangular partitioning.

Definition 5 (Hyper-rectangular partition) *Let* $\mathcal{I}$ *be a finite set of indices. Then, a family of hyper-rectangles* $\{[\underline{x}_i, \overline{x}_i]\}_{i \in \mathcal{I}}$*, is called a* hyper-rectangular partition *of compact set* $\mathcal{X}$ *if, for every* $x \in \mathcal{X}$*, there exists* $i \in \mathcal{I}$ *such that* $x \in [\underline{x}_i, \overline{x}_i]$ *and for every* $i, j \in \mathcal{I}$*, such that* $i \neq j$*, we have* $[\underline{x}_i, \overline{x}_i] \cap [\underline{x}_j, \overline{x}_j] = \emptyset$*.*

Consider a monotone discrete-time dynamical system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ satisfying Assumption 2 with a hyper-rectangular partition $\{[\underline{x}_i, \overline{x}_i]\}_{i \in \mathcal{I}}$ for $\mathcal{X}$ with $\mathcal{I}_0$ and $\mathcal{I}_u$ minimal subsets of $\mathcal{I}$ such that

$$\mathcal{X}_0 \subseteq \bigcup_{i \in \mathcal{I}_0} [\underline{x}_i, \overline{x}_i], \qquad \mathcal{X}_u \subseteq \bigcup_{i \in \mathcal{I}_u} [\underline{x}_i, \overline{x}_i]. \tag{10}$$

Suppose that $\mathsf{F} = (\underline{\mathsf{F}}, \overline{\mathsf{F}})$ and $\partial\mathsf{F} = (\underline{\partial\mathsf{F}}, \overline{\partial\mathsf{F}})$ are the parametrized inclusion functions for $\widehat{\mathbb{V}}$ and $D\widehat{\mathbb{V}}$, respectively. We consider the data-driven conditions

$$\overline{\mathsf{F}}(\underline{u}_i, \overline{u}_i) \leq 0, \qquad \forall i \in \mathcal{I}_0, \tag{11a}$$

$$\underline{\mathsf{F}}(\underline{u}_i, \overline{u}_i) > 0, \qquad \forall i \in \mathcal{I}_u, \tag{11b}$$

$$\underline{\partial\mathsf{F}}(\underline{u}_i, \overline{u}_i) \geq 0, \qquad \forall i \in \mathcal{I}, \tag{11c}$$

where $\underline{u}_i = \prod_{k=1}^N \left(P_T^{x_{0,k}}(\underline{x}_i) - \frac{1}{T+1}\right)$ and $\overline{u}_i = \prod_{k=1}^N \left(P_T^{x_{0,k}}(\overline{x}_i + \varepsilon_T \mathbb{1}_n)\right)$, for every $i \in \mathcal{I}$. Now, we can state the following optimization problem:

$$\textbf{SpOP} := \min_{p \in \mathcal{P}} \mathsf{Loss}(p, \underline{x}_i, \overline{x}_i) \quad \text{subject to (11)}, \tag{12}$$

where $\underline{x}_i, \overline{x}_i$ are the *finitely many* extreme points of the hyper-rectangular partition $\{[\underline{x}_i, \overline{x}_i]\}_{i \in \mathcal{I}}$. The next theorem establishes that, for a monotone discrete-time system $\mathcal{S}$, searching for a DBF of the form $\widehat{\mathbb{V}} : \mathcal{P} \times \mathbb{R}^N \to \mathbb{R}$ can be reduced to solving **SpOP** in (12).

Theorem 4 (Data-driven Dissipation Barrier Certificates) *Consider a discrete-time monotone system* $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ *with an unknown state transition map* f *satisfying Assumptions 1, 2, and 3. Let* $\mathcal{X}_u \subseteq \mathcal{X}$ *be an unsafe set and* $\{[\underline{x}_i, \overline{x}_i]\}_{i \in \mathcal{I}}$ *be a hyper-rectangular partitioning of* $\mathcal{X}$ *satisfying* (10)*. Suppose that there exists* $\widehat{\mathbb{V}}$ *such that the inclusion functions for* $\widehat{\mathbb{V}}$ *and* $D\widehat{\mathbb{V}}$*, denoted by* F *and* $\partial\mathsf{F}$*, satisfy the **SpOP** in* (12)*. Then* $\mathcal{S}$ *is safe with respect to the unsafe set* $\mathcal{X}_u$*.*

Proof: See Section 6.3 in the Appendix. ■

Remark 2 *The following remarks are in order.*

1. *(Truncated trajectories) Theorem 4 employs the* ***SpOP*** *with constraints in* (11) *to ensure the safety of* $\mathcal{S}$*. Consequently, it requires only the collected* T*-truncated trajectories* $\mathbf{f}_T^{(t)}(x_{0,1}),\ldots,\mathbf{f}_T^{(t)}(x_{0,N})$ *of* $\mathcal{S}$*, as stated in Assumption 1,*

2. *(Inclusion function computation) In Theorem 4, the computation of the inclusion functions* F *and* $\partial\mathsf{F}$ *can be aided with interval analysis libraries such as [17]. In the next subsection, we present a systematic approach for finding inclusion functions for polynomial maps,*

3. *(Granularity of partitioning) From the proof of Theorem 4, it follows that the size of each hyper-rectangular cell* $[\underline{x}_i,\overline{x}_i]$ *determines the under- and over-approximation of* $\widehat{\mathbb{V}}(\overline{P}^{x_{0,k}}(x))$*. Consequently, if no solution exists for* ***SpOP*** *in* (12) *with a given hyper-rectangular partition, a solution may still be found by refining the partition.*

4.1 Polynomial Dissipation Barrier Inclusion Functions

In this subsection, we consider a polynomial template of degree less than or equal to d for $\widehat{\mathbb{V}}$ and demonstrate how to construct inclusion functions for $\widehat{\mathbb{V}}$ and $D\widehat{\mathbb{V}}$. We consider $\widehat{\mathbb{V}}:\mathcal{P}\times\mathbb{R}^N\to\mathbb{R}$, where $\mathcal{P}$ represents the set of coefficients of a polynomial of order less than or equal to d, given by $\mathcal{P}=\mathbb{R}^m$ with $m:=\frac{(N+d)!}{N!d!}$. For every $p\in\mathcal{P}$, we first consider the decomposition $p=p_1^+-p_2^+$, where $p_1^+,p_2^+\in\mathbb{R}^m_{\geq 0}$. This will lead to the following decomposition:

$$\widehat{\mathbb{V}}(p,u):=\widehat{\mathbb{V}}(p_1^+,u)-\widehat{\mathbb{V}}(p_2^+,u),$$

Moreover, the polynomials $\widehat{\mathbb{V}}(p_1^+,u)$ and $\widehat{\mathbb{V}}(p_2^+,u)$ are monotone functions on $\mathbb{R}^N_{\geq 0}$. Thus, for every $u\in[\underline{u},\overline{u}]$,

$$\widehat{\mathbb{V}}(p_1^+,\underline{u})-\widehat{\mathbb{V}}(p_2^+,\overline{u})\leq\widehat{\mathbb{V}}(p,u)\leq\widehat{\mathbb{V}}(p_1^+,\overline{u})-\widehat{\mathbb{V}}(p_2^+,\underline{u}).$$

Hence, we can obtain the inclusion functions:

$$\begin{aligned}\overline{\mathsf{F}}(\underline{u},\overline{u})&=\widehat{\mathbb{V}}(p_1^+,\overline{u})-\widehat{\mathbb{V}}(p_2^+,\underline{u}),\\ \underline{\mathsf{F}}(\underline{u},\overline{u})&=\widehat{\mathbb{V}}(p_1^+,\underline{u})-\widehat{\mathbb{V}}(p_2^+,\overline{u}),\\ \underline{\partial\mathsf{F}}(\underline{u},\overline{u})&=D_{\underline{u}}\widehat{\mathbb{V}}(p_1^+,\underline{u})-D_{\overline{u}}\widehat{\mathbb{V}}(p_2^+,\overline{u}).\end{aligned}\tag{13}$$

Using these inclusion functions, we can set the **SpOP** in (12) as a linear program and use any available optimization solver to search for parameters $p_1^+,p_2^+\in\mathbb{R}^m_{\geq 0}$. We use this approach in the next case study section.

5 Numerical Simulation

In this section, we synthesize polynomial DBFs for two systems with different properties. The first one is a linear system with stable point, and the second one is a nonlinear system with two stable points and one saddle point.

Example 2 *Consider a system* $\mathcal{S}=(\mathcal{X},\mathcal{X}_0,f)$ *with* $\mathcal{X}=[0,5]\times[0,5]$*,* $\mathcal{X}_0=[0,2]\times[3,5]$*, and the state transition map* f *given by*

$$\begin{aligned}f_1(x)&=(1-2\tau)x_1+\tau x_2,\\ f_2(x)&=\tau x_1+(1-2\tau)x_2.\end{aligned}\tag{14}$$

We define the unsafe set by $\mathcal{X}_u=[4,5]\times[0,3]$*. It can be easily checked that* $\mathcal{S}$ *is monotone for* $\tau\leq 1/2$*, and the origin is the globally asymptotically stable point of* $\mathcal{S}$*. Suppose that the state transition map* f *is unknown, but we have measurements of three trajectories of* $\mathcal{S}$*, namely, we have the data* $\mathbf{f}_T^{(t)}(x_{0,k})$*, for* $k\in[1;3]$ *with* $x_{0,1}=(2.5,6.4)$*,* $x_{0,2}=(5.6,6.5)$*,* $x_{0,3}=(6.0,0.5)$ *and* $T=1300$*. We construct a uniform hyper-rectangular partition of* $\mathcal{X}$ *such that* $\|\overline{x}_i-\underline{x}_i\|_\infty=0.2$*. We construct the dissipation functions* $P_T^{x_{0,k}}$*, for* $k\in[1;3]$*, and use Theorem 4 with the inclusion functions in* (13) *to obtain a linear dissipation barrier function as shown in Fig. 2.*

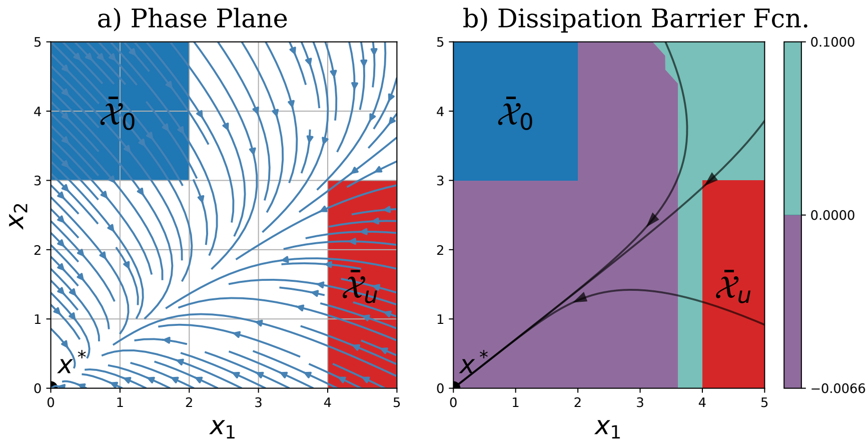

Figure 2: a) Phase plane of (14). b) 0-Sublevel sets of a DBF for Example 2 with $\tau = 0.01$. The black lines illustrate the 3 trajectories used for the synthesis. Note that the 0-sublevel set contains $\mathcal{X}_0$ and does not contain $\mathcal{X}_u$ as required by Theorem 4.

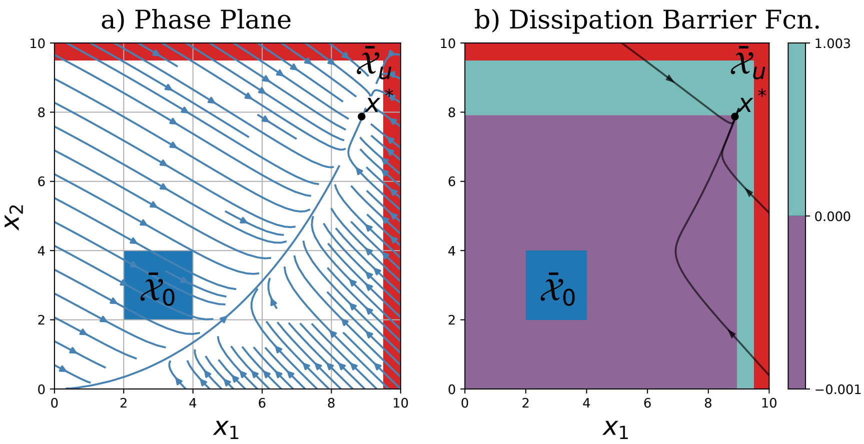

Figure 3: a) Phase plane for (15). b) Sublevel sets of a DBF for Example 3 with $\tau = 0.001$. The black lines illustrate the 3 trajectories used for the synthesis. Note that the 0-sublevel set contain $\mathcal{X}_0$ and does not contain $\mathcal{X}_u$ as required by Theorem 4.

Example 3 *Consider a system $\mathcal{S} = (\mathcal{X}, \mathcal{X}_0, f)$ with $\mathcal{X} = [0, 10] \times [0, 10]$, $\mathcal{X}_0 = [2, 4] \times [2, 4]$, and the state transition map f given by*

$$\begin{aligned} f_1(x) &= x_1 + \tau(20x_2 - x_1^2 - x_1x_2 - x_1), \\ f_2(x) &= x_2 + \tau(x_1^2 - 10x_2). \end{aligned} \tag{15}$$

We define the unsafe set by $\mathcal{X}_u = [9.5, 10] \times [0.0, 10.0] \cup [0.0, 10.0] \times [9.5, 10.0]$. This system is monotone if entries of the Jacobian of the state transition map (16) *are positive, namely, $Df(x) \geq \mathbb{0}$. This is fulfilled if $x_1 \leq 20$ and $\tau \leq 1/61$.*

$$Df(x) = \begin{bmatrix} 1 - \tau(2x_1 + x_2 + 1) & \tau(20 - x_1) \\ 2\tau x_1 & 1 - 10\tau \end{bmatrix}. \tag{16}$$

Suppose that the state transition map f is unknown, but we have measurements of three trajectories of $\mathcal{S}$, namely, we have the data $\mathbf{f}_T^{(t)}(x_{0,k})$, for $k \in [1; 3]$ with $x_{0,1} = (5.0, 10.1)$, $x_{0,2} = (10.4, 0.0)$, $x_{0,3} = (10.1, 5.0)$ and $T = 2000$. We construct a uniform hyper-rectangular partition of $\mathcal{X}$ such that $\|\overline{x}_i - \underline{x}_i\|_\infty = 0.05$. We construct the dissipation functions $P_T^{x_{0,k}}$, for $k \in [1; 3]$ and use Theorem 4 with the inclusion functions in (13) *to obtain a linear dissipation barrier function as shown in Fig. 3.*

6 Conclusion

In this paper, we addressed the two main drawbacks of data-driven safety verification approaches by leveraging the properties of monotone systems. First, we constructed a family of "basis" functions (dissipation functions) provided N trajectories of the system. Then, we analyzed the properties of the dissipation functions and used them for the synthesis of *dissipation barrier functions (DBF)*. We showed that under certain conditions, this DBF acts as a so-called barrier certificate. We cast the search of the DBF as an SpOP and provided guarantees for the correctness of the certificate over the whole state set. Finally, we illustrated our approach by computing polynomial DBFs for two systems. In future work, we will address the conservative nature of monotone DBF, providing new types of dissipation function, and we will study time-varying dynamical systems.

References

[1] A. Makdesi, A. Girard, and L. Fribourg, "Data-driven models of monotone systems," *IEEE Transactions on Automatic Control*, vol. 69, no. 8, pp. 5294–5309, 2023.

[2] P. Wieland and F. Allgöwer, "Constructive safety using control barrier functions," *IFAC Proceedings Volumes*, vol. 40, no. 12, pp. 462–467, 2007.

[3] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *Hybrid Systems: Computation and Control*, R. Alur and G. J. Pappas, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 477–492.

[4] A. Nejati, A. Lavaei, P. Jagtap, S. Soudjani, and M. Zamani, "Formal verification of unknown discrete-and continuous-time systems: A data-driven approach," *IEEE Transactions on Automatic Control*, vol. 68, no. 5, pp. 3011–3024, 2023.

[5] R. Isermann and M. Münchhof, *Identification of dynamic systems: an introduction with applications*. Springer, 2011, vol. 85.

[6] A. Nejati, B. Zhong, M. Caccamo, and M. Zamani, "Data-driven controller synthesis of unknown nonlinear polynomial systems via control barrier certificates," in *Learning for Dynamics and Control Conference*. PMLR, 2022, pp. 763–776.

[7] C. Folkestad, Y. Chen, A. D. Ames, and J. W. Burdick, "Data-driven safety-critical control: Synthesizing control barrier functions with Koopman operators," *IEEE Control Systems Letters*, vol. 5, no. 6, pp. 2012–2017, 2021.

[8] D. Ajeleye and M. Zamani, "Data-driven controller synthesis via co-büchi barrier certificates with formal guarantees," *IEEE Control Systems Letters*, 2024.

[9] C. Dawson, S. Gao, and C. Fan, "Safe control with learned certificates: A survey of neural Lyapunov, barrier, and contraction methods for robotics and control," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 1749–1767, 2023.

[10] A. Robey, H. Hu, L. Lindemann, H. Zhang, D. V. Dimarogonas, S. Tu, and N. Matni, "Learning control barrier functions from expert demonstrations," in *IEEE Conference on Decision and Control (CDC)*, 2020, pp. 3717–3724.

[11] H. L. Smith, *Monotone dynamical systems: An Introduction to the Theory of Competitive and Cooperative Systems*. American Mathematical Soc., 1995, no. 41.

[12] Y. Wang and J. Jiang, "The general properties of discrete-time competitive dynamical systems," *Journal of Differential Equations*, vol. 176, no. 2, pp. 470–493, 2001.

[13] G. Dirr, H. Ito, A. Rantzer, and B. Rüffer, "Separable Lyapunov functions for monotone systems: Constructions and limitations," *Discrete Contin. Dyn. Syst. Ser. B*, vol. 20, no. 8, pp. 2497–2526, 2015.

[14] A. Sootla, "Construction of max-separable lyapunov functions for monotone systems using the Koopman operator," in *IEEE Conference on Decision and Control (CDC)*, 2016, pp. 6512–6517.

[15] H. R. Feyzmahdavian, B. Besselink, and M. Johansson, "Stability analysis of monotone systems via max-separable Lyapunov functions," *IEEE Transactions on Automatic Control*, vol. 63, no. 3, pp. 643–656, 2017.

[16] M. Abate and S. Coogan, "Robustly forward invariant sets for mixed-monotone systems," *IEEE Transactions on Automatic Control*, vol. 67, no. 9, pp. 4947–4954, 2022.

[17] A. Harapanahalli, S. Jafarpour, and S. Coogan, "Forward invariance in neural network controlled systems," *IEEE Control Systems Letters*, vol. 7, pp. 3962–3967, 2023.

[18] ——, "immrax: A parallelizable and differentiable toolbox for interval analysis and mixed monotone reachability in JAX," *IFAC-PapersOnLine*, vol. 58, no. 11, pp. 75–80, 2024.

[19] L. Jaulin, M. Kieffer, O. Didrit, E. Walter, L. Jaulin, M. Kieffer, O. Didrit, and É. Walter, *Interval analysis.* Springer, 2001.

[20] K. Chatterjee, A. Goharshady, E. Goharshady, M. Karrabi, and D. Žikelić, "Sound and complete witnesses for template-based verification of LTL properties on polynomial programs," in *International Symposium on Formal Methods.* Springer, 2024, pp. 600–619.

APPENDIX

6.1 Proof of Theorem 1

Regarding part 1, consider $x, y \in \mathcal{X}$ such that $x \leq y$. Our goal is to show that $P^{x_0}(x) \leq P^{x_0}(y)$. We consider two cases: (i) there exists a $t \geq 0$ such that $y \leq f^{(t)}(x_0)$, and (ii) there does not exist $t \geq 0$ such that $y \leq f^{(t)}(x_0)$. In case (i), for every $t \in \mathbb{N}$ such that $y \leq f^{(t)}(x_0)$, we have $x \leq y \leq f^{(t)}(x_0)$. This means that

$$\inf\{\tfrac{1}{t+1} \mid x \leq f^{(t)}(x_0)\} \leq \inf\{\tfrac{1}{t+1} \mid y \leq f^{(t)}(x_0)\}.$$

This implies $P^{x_0}(x) \leq P^{x_0}(y)$. In case (ii), using the definition of dissipation function (4), we have $P^{x_0}(y) = \alpha$. This implies $P^{x_0}(x) \leq \alpha = P^{x_0}(y)$.

Regarding part 2, we consider two cases, (i) there exists a $t^* \geq 1$ such that $f(x) \leq f^{(t^*)}(x_0)$, (ii) there does not exist $t^* \geq 1$ such that $f(x) \leq f^{(t^*)}(x_0)$. In case (i), we get

$$P^{x_0}(f(x)) = \inf\left\{\tfrac{1}{t+1} \mid f(x) \leq f^{(t)}(x_0)\right\}.$$

Thus, using monotonicity of f,

$$\begin{aligned} P^{x_0}(f(x)) &= \inf\left\{\tfrac{1}{t+1} \mid f(x) \leq f^{(t)}(x_0), t \geq 1\right\}, \\ &\leq \inf\left\{\tfrac{1}{t+1} \mid x \leq f^{(t-1)}(x_0), t \geq 1\right\}, \end{aligned}$$

or equivalently under a change of coordinates $s := t - 1$,

$$P^{x_0}(f(x)) \leq \inf\left\{\tfrac{1}{s+2} \mid x \leq f^{(s)}(x_0), s \geq 0\right\},$$

since $s + 1 < s + 2$, we have

$$P^{x_0}(f(x)) < \inf\left\{\tfrac{1}{s+1} \mid x \leq f^{(s)}(x_0), s \geq 0\right\} = P^{x_0}(x).$$

In case (ii), we claim that $P^{x_0}(x) = \alpha$. Suppose, for contradiction, that $P^{x_0}(x) < \alpha$. Then, there exists a $t' \geq 0$ such that $x \leq f^{(t)}(x)$ holds. By the monotonicity of f, we obtain $f(x) \leq f^{(t'+1)}(x)$, which contradicts the assumption of this case for $t^* = t'+1$. Therefore, we conclude that $P^{x_0}(x) = \alpha$ and thus $P^{x_0}(f(x)) \leq \alpha = P^{x_0}(x)$.

Regarding part 3, using the result from part 2, we have $P^{x_0}(f(x)) \leq P^{x_0}(x) \leq c$ for all $x \in P^{x_0}_{\leq \tau}$. By induction, it follows that $P^{x_0}(f^{(t)}(x)) \leq c$ for all $t \geq 0$. ■

6.2 Proof of Theorem 3

Regarding part 1, consider $x, y \in \mathcal{X}$ such that $x \leq y$. If $P_T^{x_0}(y) = \alpha$, then $P_T^{x_0}(x) \leq \alpha = P_T^{x_0}(y)$ and the result follows. If $P_T^{x_0}(y) \neq \alpha$, then

$$P_T^{x_0}(y) = \inf_{t \in [0;T]} \{\tfrac{1}{t+1} \mid y - \varepsilon_T \mathbb{1}_n \leq f^{(t)}(x_0)\}.$$

Moreover, $x - \varepsilon_T \mathbb{1}_n \leq y - \varepsilon_T \mathbb{1}_n$. This implies that

$$\begin{aligned} P_T^{x_0}(x) &= \inf_{t \in [0;T]} \{\tfrac{1}{t+1} \mid x - \varepsilon_T \mathbb{1}_n \leq f^{(t)}(x_0)\} \\ &\leq \inf_{t \in [0;T]} \{\tfrac{1}{t+1} \mid y - \varepsilon_T \mathbb{1}_n \leq f^{(t)}(x_0)\} = P_T^{x_0}(y). \end{aligned}$$

This completes the proof of monotonicity of $P_T^{x_0}$.

Regarding part 2, we first show $P^{x_0}(x) \leq P_T^{x_0}(x + \varepsilon \mathbb{1}_n)$, for every $x \in \mathcal{X}$. We consider two cases: (i) there is a time $t \leq T$ such that $x \leq f^{(t)}(x_0)$, and (ii) there is no time $t \leq T$ such that $x \leq f^{(t)}(x_0)$. In case (i), using definition of $P_T^{x_0}$ in (4),

$$\begin{aligned} P^{x_0}(x) &= \inf_{t \geq 0} \left\{\tfrac{1}{t+1} \mid x \leq f^{(t)}(x_0)\right\} \\ &= \inf_{t \geq 0} \left\{\tfrac{1}{t+1} \mid (x + \varepsilon \mathbb{1}_n) - \varepsilon \mathbb{1}_n \leq f^{(t)}(x_0), t \geq 0\right\} \\ &\leq \inf_{t \in [0;T]} \left\{\tfrac{1}{t+1} \mid (x + \varepsilon \mathbb{1}_n) - \varepsilon \mathbb{1}_n \leq f^{(t)}(x_0)\right\} \\ &= P_T^{x_0}(x + \varepsilon \mathbb{1}_n). \end{aligned}$$

where the inequality holds because $[0;T] \subset [0;\infty)$ and the last equality holds using definition of $P_T^{x_0}(x)$ in (9).

Regarding case (ii), since there exists no time $t \leq T$ such that $x \leq f^{(t)}(x_0)$, we have $P_T^{x_0}(x + \varepsilon \mathbb{1}_n) = \alpha$. As a result $P^{x_0}(x) \leq \alpha = P_T^{x_0}(x + \varepsilon \mathbb{1}_n)$.

Now, we show $P_T^{x_0}(x) - \delta \leq P^{x_0}(x)$, for every $x \in \mathcal{X}$. We consider three cases: (i) $\sup\{t \mid x \leq f^{(t)}(x_0)\} \geq T$, (ii) $\sup\{t \mid x \leq f^{(t)}(x_0)\} = t^* < T$, and (iii) $\{t \mid x \leq f^{(t)}(x_0)\} = \emptyset$.

In case (i), we have $x \leq f^{(t)}(x_0) - f^{(T)}(x_0) + f^{(T)}(x_0)$. Using inequalities (8), this implies that $x \leq \varepsilon_T \mathbb{1}_n + f^{(T)}(x_0)$. Then, by (9) we have $P_T^{x_0}(x) = \frac{1}{T+1}$. Moreover, by (4) we have $0 \leq P^{x_0}(x) \leq \frac{1}{T+1}$; hence, $P_T^{x_0}(x) - \delta \leq P^{x_0}(x_0)$.

Regarding case (ii), using definition (4), we have $P^{x_0}(x) = \frac{1}{t^*+1}$. Therefore $x \leq f^{(t^*)}(x_0)$ and this implies $x - \varepsilon \mathbb{1}_n \leq f^{(t^*)}(x_0)$. Moreover, by (9) we have $P_T^{x_0}(x) \leq \frac{1}{t^*+1}$; hence, $P_T^{x_0}(x) \leq P^{x_0}(x)$, thus, $P_T^{x_0}(x) - \delta \leq P^{x_0}(x)$.

Regarding case (iii), using definition (4), we have $P^{x_0}(x) = P_T^{x_0}(x) = \alpha$. This means that $P_T^{x_0}(x) - \delta \leq \alpha = P^{x_0}(x)$. ■

6.3 Proof of Theorem 4

Suppose that there exists a differentiable function $\mathbb{V} : \mathbb{R}^N \to \mathbb{R}$ such that F and $\partial\mathsf{F}$ are inclusion functions for $\mathbb{V}$ and $D\mathbb{V}$, respectively and they satisfy conditions (11). We show that $\mathbb{V}$ is a dissipation barrier function for $\mathcal{S}$. Let $x \in \mathcal{X}_0$. Then there exists $i \in \mathcal{I}_0$ such that $x \in [\underline{x}_i, \overline{x}_i]$. Using condition (11a), we get that $\overline{\mathsf{F}}(\underline{u}_i, \overline{u}_i) \leq 0$. Using Assumption 3 and Theorem 3 part 2, we have $P_T^{x_{0,k}}(x) - \frac{1}{T+1} \leq P^{x_{0,k}}(x) \leq P_T^{x_{0,k}}(x + \varepsilon_T \mathbb{1}_n)$. On the other hand, using Theorem 3 part 1,

$$\begin{aligned} P_T^{x_{0,k}}(\underline{x}_i) - \tfrac{1}{T+1} &\leq P_T^{x_{0,k}}(x) - \tfrac{1}{T+1} \\ &\leq P^{x_{0,k}}(x) \leq P_T^{x_{0,k}}(x + \varepsilon_T \mathbb{1}_n) \\ &\leq P_T^{x_{0,k}}(\overline{x}_i + \varepsilon_T \mathbb{1}_n). \end{aligned}$$

This implies that $\Pi_{k=1}^N P^{x_{0,k}}(x) \in [\underline{u}_i, \overline{u}_i]$. Therefore, we get $\mathbb{V}(\Pi_{k=1}^N P^{x_{0,k}}(x)) \leq \overline{\mathsf{F}}(\underline{u}_i, \overline{u}_i) \leq 0$. This shows that condition (6a) holds.

Now let $x \in \mathcal{X}_u$. There exists $i \in \mathcal{I}_U$ such that $x \in [\underline{x}_i, \overline{x}_i]$. Now using condition (11b), we get that $\underline{\mathsf{F}}(\underline{u}_i, \overline{u}_i) \geq 0$. Using Assumption 3 and Theorem 3 part 2, we have $P_T^{x_{0,k}}(x) - \frac{1}{T+1} \leq P^{x_{0,k}}(x) \leq P_T^{x_{0,k}}(x + \varepsilon_T \mathbb{1}_n)$. On the other hand, using Theorem 3 part 1,

$$\begin{aligned} P_T^{x_{0,k}}(\underline{x}_i) - \tfrac{1}{T+1} &\leq P_T^{x_{0,k}}(x) - \tfrac{1}{T+1} \\ &\leq P^{x_{0,k}}(x) \leq P_T^{x_{0,k}}(x + \varepsilon_T \mathbb{1}_n) \\ &\leq P_T^{x_{0,k}}(\overline{x}_i + \varepsilon_T \mathbb{1}_n). \end{aligned}$$

This implies that $\Pi_{k=1}^{N} P^{x_{0},k}(x) \in [\underline{u}_i, \overline{u}_i]$. Therefore, we get $\mathbb{V}(\overline{P}^{x_{0},k}(x)) \geq \underline{\mathsf{F}}(\underline{u}_i, \overline{u}_i) \geq 0$. This shows that condition (6b) holds.

Finally, let $x \in \mathcal{X}$ be such that $\mathbb{V}(\Pi_{k=1}^{N} P^{x_0,k}(x)) \leq 0$. Then, there exists $i \in \mathcal{I}$ such that $x \in [\underline{x}_i, \overline{x}_i]$. Using Assumption 3 and Theorem 3 part 2, we have $P_T^{x_0,k}(x) - \frac{1}{T+1} \leq P^{x_0,k}(x) \leq P_T^{x_0,k}(x + \varepsilon_T \mathbb{1}_n)$. On the other hand, using Theorem 3 part 1,

$$\begin{aligned} P_T^{x_0,k}(\underline{x}_i) - \tfrac{1}{T+1} &\leq P_T^{x_0,k}(x) - \tfrac{1}{T+1} \\ &\leq P^{x_0,k}(x) \leq P_T^{x_0,k}(x + \varepsilon_T \mathbb{1}_n) \\ &\leq P_T^{x_0,k}(\overline{x}_i + \varepsilon_T \mathbb{1}_n). \end{aligned}$$

This implies that $\Pi_{k=1}^{N} P^{x_0,k}(x) \in [\underline{u}_i, \overline{u}_i]$. Therefore, $D\mathbb{V}(\Pi_{k=0}^{N} P^{x_0,k}(x)) \geq \underline{\partial \mathsf{F}}(\underline{u}_i, \overline{u}_i) \geq 0$. As a result, condition (6c) holds.