Data-Driven Abstraction and Model Invalidation for Unknown Systems With Bounded Jacobians

Zeyuan Jin, *Graduate Student Member, IEEE*, Mohammad Khajenejad, *Member, IEEE*, and Sze Zheng Yong, *Member, IEEE*

***Abstract*—In this letter, we consider data-driven abstraction and model invalidation problems for unknown nonlinear discrete-time dynamical systems with bounded Jacobians, where only prior noisy sampled data of the systems, instead of mathematical models, are available. First, we introduce a novel non-parametric learning approach to over-approximate the unknown model/dynamics with upper and lower functions, i.e., to find model abstractions, under the assumption of known bounded Jacobians. Notably, the resulting data-driven models can be mathematically proven to be equal to or more accurate than componentwise Lipschitz continuity-based methods. Further, we show that the resulting data-driven model can be used to determine its (in)compatibility with a newly observed length-T output trajectory, i.e., to (in)validate models, using a tractable feasible check. Finally, we propose a method to estimate the Jacobian bounds if they are not known or given.**

***Index Terms*—Nonlinear systems identification, model validation.**

I. Introduction

MATHEMATICAL models of systems/processes are often assumed to be known or available during system design and analysis in most cyber-physical systems applications. However, since noise/disturbances and model uncertainties are inevitable in the real world, it is often impossible to compute the precise model of the complex dynamics. Even when the models are known, abstraction approaches have been widely developed to approximate the original complex dynamics of nonlinear, uncertain or hybrid systems, with simpler dynamics. Hence, the development of methods for obtaining accurate model abstractions from noisy sampled data will be beneficial for control design and analysis.

Literature Review: The model (in)validation problem, i.e., to determine whether a finite sequence of experimental input-output data can be generated by an admissible model set [1], is crucial in many engineering applications such as model identification and fault diagnosis [2], [3]. Similarly, when mathematical models are available, *abstraction* approaches have been widely studied for linear systems [4], nonlinear systems [5], [6], uncertain affine and nonlinear systems [3], and discrete-time hybrid systems [7] for the purpose of finding simpler dynamics/models that retain most properties of interest with the original system to reduce computational complexity. However, these approaches are not applicable when accurate mathematical models are unavailable.

On the other hand, in order to *abstract*, i.e., to over-approximate unknown dynamics when (bounded) set-valued uncertainties are considered, data-driven approaches have been leveraged extensively over the last few years to take advantage of observed/sampled input-output data for finding *a set of known systems* that retain most properties of interest with the unknown system dynamics [8], [9]. In this context, data-driven approaches [10], [11] have been proposed to obtain finite-state/symbolic abstractions. On the other hand, the work in [12] proposes a recursive algorithm to compute upper and lower bounding functions without state discretization for univariate Lipschitz continuous unknown dynamics, while the study in [13], [14] considered the extension to multivariate functions. This technique was extended to Hölder continuous unknown dynamics in [15]. In [16], the authors further considered componentwise Hölder continuous functions where the contribution of each input to each output of the function is independently counted.

Contributions: In this letter, we consider a novel continuity assumption in the form of bounded Jacobian matrices that can be mathematically proven to lead to equivalent or tighter data-driven abstractions when compared to abstractions based on (componentwise) Lipschitz continuity assumptions. Then, leveraging this approach to over-approximate/abstract unknown dynamic systems from observed/sampled data, we consider the data-driven model invalidation problem, where we determine if a newly observed noisy length-T output trajectory can be generated by the learned data-driven model/abstraction. Note that while the proposed bounded Jacobian-based abstraction models may sometimes be more conservative than those

Manuscript received 21 March 2022; revised 21 May 2022; accepted 7 June 2022. Date of publication 22 June 2022; date of current version 1 July 2022. This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) under Grant D18AP00073, and in part by the National Science Foundation (NSF) under Grant CNS-1943545. Recommended by Senior Editor G. Cherubini. *(Corresponding author: Sze Zheng Yong.)*

Zeyuan Jin is with the School for Engineering of Matter, Transport and Energy, Arizona State University, Tempe, AZ 85281 USA (e-mail: zjin43@asu.edu).

Mohammad Khajenejad was with the School for Engineering of Matter, Transport and Energy, Arizona State University, Tempe, AZ 85281 USA. He is now with the Department of Mechanical and Aerospace Engineering, University of California, San Diego, CA 92122 USA (e-mail: mkhajenejad@ucsd.edu).

Sze Zheng Yong was with the School for Engineering of Matter, Transport and Energy, Arizona State University, Tempe, AZ 85281 USA. He is now with the Department of Mechanical and Industrial Engineering, Northeastern University, Boston, MA 02115 USA (e-mail: s.yong@northeastern.edu).

Digital Object Identifier 10.1109/LCSYS.2022.3185410

based on componentwise Hölder continuity, the advantage of this approach is that we obtain a data-driven model invalidation algorithm that can be posed as a mixed-integer linear feasibility check problem, instead of an intractable mixed-integer nonlinear optimization problem. Further, when the Jacobian bounds are unavailable, we design some *offline* mixed-integer linear programs (MILP) to learn/estimate the Jacobian bounds.

II. Background

Notation: $\mathbb{R}^n, \mathbb{R}^+, \mathbb{IR}^n, \mathbb{IR}^{n\times m}$ denote the n-dimensional Euclidean space and the sets of positive real numbers, n-dimensional real intervals and n by m interval matrices. For $v, w \in \mathbb{R}^n$, $v^{(i)}$ is the i-th element of v, $\|v\|_p \triangleq (\sum_{i=1}^n |v^{(i)}|^p)^{1/p}, 1 \leq p \leq \infty$, in particular, $\|v\|_\infty \triangleq \max_{1\leq i \leq n} v^{(i)}$. $v \leq w$ is a component-wise inequality. For $M \in \mathbb{R}^{p\times q}$, $M^\oplus \triangleq \max(M, 0_{p\times q})$, $M^\ominus \triangleq M^\oplus - M$, $|M| \triangleq M^\oplus + M^\ominus$ and $M^{(i,j)}$ denotes the (i, j)'th entry of M.

Next, we introduce some useful results that will be used throughout this letter.

Proposition 1 [17, eq. (2.53)]: Let $a \triangleq [\underline{a}, \overline{a}], b \triangleq [\underline{b}, \overline{b}] \in \mathbb{IR}$ be real intervals. Then,

$$a \odot b \triangleq \{xy | x \in a, y \in b\} = [\underline{h}(\overline{a}, \underline{a}, \overline{b}, \underline{b}) \leq ab \leq \overline{h}(\overline{a}, \underline{a}, \overline{b}, \underline{b})],$$

where $\underline{h}(\cdot)$ and $\overline{h}(\cdot)$ are defined as following:

$$\underline{h}(\overline{a}, \underline{a}, \overline{b}, \underline{b}) \triangleq \min\{\overline{a}\overline{b}, \overline{a}\underline{b}, \underline{a}\overline{b}, \underline{a}\underline{b}\}, \tag{1a}$$
$$\overline{h}(\overline{a}, \underline{a}, \overline{b}, \underline{b}) \triangleq \max\{\overline{a}\overline{b}, \overline{a}\underline{b}, \underline{a}\overline{b}, \underline{a}\underline{b}\}. \tag{1b}$$

Corollary 1: If the sign of a is known, the $\underline{h}(\cdot)$ and $\overline{h}(\cdot)$ in Proposition 1 can be simplified as:

$$\underline{h}(\overline{a}, \underline{a}, \overline{b}, \underline{b}) \triangleq \begin{cases} \min\{\overline{a}\underline{b}, \underline{a}\underline{b}\}, & \overline{a} \geq \underline{a} \geq 0, \\ \min\{\overline{a}\underline{b}, \underline{a}\overline{b}\}, & \overline{a} \geq 0 \geq \underline{a}, \\ \min\{\overline{a}\overline{b}, \underline{a}\overline{b}\}, & 0 \geq \overline{a} \geq \underline{a}, \end{cases} \tag{2a}$$
$$\overline{h}(\overline{a}, \underline{a}, \overline{b}, \underline{b}) \triangleq \begin{cases} \max\{\overline{a}\overline{b}, \underline{a}\overline{b}\}, & \overline{a} \geq \underline{a} \geq 0, \\ \max\{\overline{a}\overline{b}, \underline{a}\underline{b}\}, & \overline{a} \geq 0 \geq \underline{a}, \\ \max\{\overline{a}\underline{b}, \underline{a}\underline{b}\}, & 0 \geq \overline{a} \geq \underline{a}. \end{cases} \tag{2b}$$

Proof: First, since we only consider proper intervals, we have $\underline{b} \leq \overline{b}$. Thus, when $\overline{a} \geq \underline{a} \geq 0$, then $\overline{a}\underline{b} \leq \overline{a}\overline{b}$ and $\underline{a}\underline{b} \leq \underline{a}\overline{b}$, thus, $\min\{\overline{a}\underline{b}, \overline{a}\overline{b}\} = \overline{a}\underline{b}$ and $\max\{\overline{a}\underline{b}, \overline{a}\overline{b}\} = \overline{a}\overline{b}$. On the other hand, when $0 \geq \overline{a} \geq \underline{a}$, then $\overline{a}\underline{b} \geq \overline{a}\overline{b}$ and $\underline{a}\underline{b} \geq \underline{a}\overline{b}$, thus, $\min\{\overline{a}\underline{b}, \overline{a}\overline{b}\} = \overline{a}\overline{b}$ and $\max\{\overline{a}\underline{b}, \overline{a}\overline{b}\} = \overline{a}\underline{b}$. The second case with $\overline{a} \geq 0 \geq \underline{a}$ can be similarly obtained. ■

Proposition 2: Let $A \triangleq [\underline{A}, \overline{A}] \in \mathbb{IR}^{m\times n}$ and $b \triangleq [\underline{b}, \overline{b}] \in \mathbb{IR}^n$ be an interval matrix and an interval vector, respectively. Then, the following inequalities hold:

$$\underline{g}(\overline{A}, \underline{A}, \overline{b}, \underline{b}) \leq Ab \leq \overline{g}(\overline{A}, \underline{A}, \overline{b}, \underline{b}),$$

where $\underline{g}(\cdot)$ and $\overline{g}(\cdot)$ are obtained via either of the following:

i) For $i \in \{1, \ldots, m\}$,

$$\underline{g}^{(i)}(\overline{A}, \underline{A}, \overline{b}, \underline{b}) \triangleq \sum_{j=1}^n \underline{h}(\overline{A}^{(i,j)}, \underline{A}^{(i,j)}, \overline{b}^{(j)}, \underline{b}^{(j)}), \tag{3a}$$
$$\overline{g}^{(i)}(\overline{A}, \underline{A}, \overline{b}, \underline{b}) \triangleq \sum_{j=1}^n \overline{h}(\overline{A}^{(i,j)}, \underline{A}^{(i,j)}, \overline{b}^{(j)}, \underline{b}^{(j)}), \tag{3b}$$

with $\underline{h}(\cdot)$ and $\overline{h}(\cdot)$ given in (1).

ii)
$$\underline{g}(\overline{A}, \underline{A}, \overline{b}, \underline{b}) \triangleq \underline{A}^\oplus \underline{b}^\oplus - \overline{A}^\oplus \underline{b}^\ominus - \underline{A}^\ominus \overline{b}^\oplus + \overline{A}^\ominus \overline{b}^\ominus, \tag{4a}$$
$$\overline{g}(\overline{A}, \underline{A}, \overline{b}, \underline{b}) \triangleq \overline{A}^\oplus \overline{b}^\oplus - \underline{A}^\oplus \overline{b}^\ominus - \overline{A}^\ominus \underline{b}^\oplus + \underline{A}^\ominus \underline{b}^\ominus. \tag{4b}$$

Proof: The results in (3) can be obtained by applying Proposition 1 element-wise to bound Ab from below and above and the bounds in (4) are derived in [18, Proposition 1]. ■

A. Modeling Framework

We consider a noisy discrete-time nonlinear auto-regressive dynamic system model $\mathcal{G}$ of (known) order n_y (with time step $k \geq 0$):

$$y_{k+1} = f(s_k) + w_k, \tag{5}$$
$$\tilde{y}_k = y_k + v_k, \tag{6}$$

where $y_k \triangleq \left[y_k^{(1)} \; y_k^{(2)} \cdots y_k^{(m)}\right]^\top \in \mathcal{Y} \triangleq [\mathcal{Y}_l, \mathcal{Y}_u] \subset \mathbb{IR}^m$, $s_k \triangleq [y_k^{(1)} \cdots y_k^{(m)} \cdots y_{k-n_y+1}^{(1)} \cdots y_{k-n_y+1}^{(m)}]^\top \in \mathcal{S} \triangleq \mathcal{Y}^{n_y} \subset \mathbb{IR}^n$ with $n = mn_y$ and $f(\cdot) \triangleq [f^{(1)}(\cdot) \cdots f^{(i)}(\cdot) \cdots f^{(m)}(\cdot)]^\top$ with $f^{(i)}(\cdot) : \mathbb{R}^n \to \mathbb{R}$ for all $i \in \{1 \ldots m\}$, as well as process and measurement noise signals $w_k \in \mathcal{W}, v_k \in \mathcal{V}$ that are bounded, i.e., $\mathcal{W} \triangleq \{w_k \mid |w_k^{(i)}| \leq \varepsilon_w^{(i)}, \forall i \in \{1, 2, \ldots, m\}\}, \mathcal{V} \triangleq \{v_k \mid |v_k^{(i)}| \leq \varepsilon_v^{(i)}, \forall i \in \{1, 2, \ldots, m\}\}$, with $\varepsilon_w^{(i)}, \varepsilon_v^{(i)} > 0$. Functions $f^i(\cdot)$ are unknown but a noisy sampled data set $\mathcal{D} = \bigcup_{\ell=1}^N \mathcal{D}_\ell$ is available, consisting of N trajectories each of length T_ℓ represented by $\mathcal{D}_\ell = \{\tilde{y}_{j,\ell} | j = 0, \ldots, T_\ell - 1\}$, where $\tilde{y}_{k,\ell}$ are noise corrupted measurements of $y_{k,\ell} \in \mathcal{Y}$ according to (6). We further define noise corrupted input $\tilde{s}_{j,\ell} \triangleq [(\tilde{y}_{j,\ell})^\top \cdots (\tilde{y}_{j-n_y+1,\ell})^\top]^\top$ and $\varepsilon_s^{(i)}$ as the upper bound of $|s_{j,\ell}^{(im)} - \tilde{s}_{j,\ell}^{(im)}|$ with $\varepsilon_s^{(i)} = \varepsilon_v^{(i)}$ (i.e., $\varepsilon_s = \varepsilon_v$) for all $i \in \{1, \ldots, m\}$, $j \in \{n_y, \ldots, T_\ell - 1\}$ and $\ell \in \{1, \ldots, N\}$. Then, a concatenated data set $\overline{\mathcal{D}}_\ell \triangleq \{(\tilde{s}_{j,\ell}, \tilde{y}_{j+1,\ell}) | j = n_y, \ldots, T_\ell - 1\}$ can be constructed from $\mathcal{D}_\ell$ for simplicity and similarly, $\overline{\mathcal{D}} = \bigcup_{\ell=1}^N \overline{\mathcal{D}}_\ell$.

Further, we assume the following about the function f:

Assumption 1: The vector-valued function $f(\cdot)$ is continuous in its domain and the Jacobian matrix $\nabla f(\cdot)$ of f is bounded, i.e., there exists two finite-valued matrices J_u and J_l such that $J_l \leq \nabla f(x) \leq J_u$ for all x in the domain of f.

B. Abstraction/Over-Approximation

The abstraction/over-approximation procedure aims to find a pair of functions $\underline{f}$ and $\overline{f}$ to over-approximate the unknown function f in the system dynamics in (5), i.e., $\underline{f}$ and $\overline{f}$ satisfy:

$$\underline{f}(s_k) \leq f(s_k) \leq \overline{f}(s_k), \;\; \forall s_k \in \mathcal{S},$$

and consequently, for all $s_k \in \mathcal{S}, w_k \in \mathcal{W}$, the system dynamics in (5) is over-approximated by an abstraction model $\mathcal{H} \triangleq \{\overline{f}, \underline{f}\}$ satisfying:

$$\underline{f}(s_k) - \varepsilon_w \leq y_{k+1} = f(s_k) + w_k \leq \overline{f}(s_k) + \varepsilon_w. \tag{7}$$

C. Length-T Behavior

Next, to introduce the model invalidation problem, we utilize the definition in [2] of the length-T behavior of the original unknown model $\mathcal{G}$ and the abstracted model $\mathcal{H}$ based on the prior sampled data $\mathcal{D}$, as follows.

Definition 1 (Length-T Behaviors of Original and Abstracted Models $\mathcal{G}$ and $\mathcal{H}$): The length-T behaviors of the (unknown) original model $\mathcal{G}$ and the abstracted model $\mathcal{H}$ are the sets of all length-T output trajectories compatible with $\mathcal{G}$ and $\mathcal{H}$, respectively, given by the sets

$$\mathcal{B}^T(\mathcal{G}) := \{\{\tilde{y}_k\}_{k=0}^{T-1} \mid \exists y_k \in \mathcal{Y}, w_k \in \mathcal{W}, v_k \in \mathcal{V}, \text{ for } k \in \mathbb{Z}_{T-1}^0, \quad \text{s.t. (5)–(6) hold}\}. \tag{8}$$

$$\mathcal{B}^T(\mathcal{H}) := \{\{\tilde{y}_k\}_{k=0}^{T-1} \mid \exists y_k \in \mathcal{Y}, w_k \in \mathcal{W}, v_k \in \mathcal{V}, \text{ for } k \in \mathbb{Z}_{T-1}^0, \quad \text{s.t. (6)–(7) hold}\}. \tag{9}$$

Furthermore, by definition, $\mathcal{B}^T(\mathcal{G}) \subseteq \mathcal{B}^T(\mathcal{H})$.

III. Problem Statement

Equipped with definitions from the previous section, we now state the data-driven model invalidation problem.

Problem 1 (Model Invalidation for $\mathcal{G}$): Given a new noisy output trajectory $\{\tilde{y}_k^n\}_{k=0}^{T-1}$, an unknown target model $\mathcal{G}$, for which only prior sampled data $\mathcal{D}_\mathcal{G}$ is available, and an integer T, determine whether the trajectory belongs to the target model, i.e., to determine if $\{\tilde{y}_k^n\}_{k=0}^{T-1} \in \mathcal{B}^T(\mathcal{G})$ holds.

However, solving Problem 1 is non-trivial since the exact dynamics of original model $\mathcal{G}$ is not available. Thus, we propose to solve an auxiliary problem that provides a sufficient solution to Problem 1. Specifically, we propose a two-step process, where the first step obtains an abstraction model of the unknown dynamics $\mathcal{G}$ from prior sampled data $\mathcal{D}_\mathcal{G}$:

Problem 2 (Data-Driven Abstraction): For a set of N sampling data points $\mathcal{D}_\mathcal{G}$, find a pair of upper and lower functions $\overline{f}$ and $\underline{f}$ (i.e., $\mathcal{H} \triangleq \{\overline{f}, \underline{f}\}$) such that:

$$\underline{f}(s) \le f(s) \le \overline{f}(s), \quad \forall s \in \mathcal{S}, \tag{10}$$

where $f(\cdot) : \mathcal{S} \subset \mathbb{R}^n \to \mathbb{R}^m$ is the unknown function in the system dynamics (5), and correspondingly determine $\mathcal{B}^T(\mathcal{H})$.

Then, the second step seeks to solve the following model invalidation problem for the abstracted models.

Problem 3 (Data-Driven Model Invalidation for $\mathcal{H}$): Given a new noisy input-output trajectory $\{\tilde{y}_k^n\}_{k=0}^{T-1}$, an abstraction model $\mathcal{H}$ of target model $\mathcal{G}$ and an integer T, determine whether the trajectory belongs to the target model. That is, to determine if $\{\tilde{y}_k^n\}_{k=0}^{T-1} \in \mathcal{B}^T(\mathcal{H})$ holds.

Leveraging the property that $\mathcal{B}^T(\mathcal{G}) \subseteq \mathcal{B}^T(\mathcal{H})$, it is trivial to show that finding the solution to Problem 1.2 and 1.3 is sufficient to solve Problem 1. Further, Problem 1.2 is interesting in its own right and useful for many control problems.

IV. Main Approach

In this section, we first present an approach to obtain a data-driven abstraction for the unknown system (5), which solves Problem 1.2. Then, we propose an optimization-based approach to invalidate the resulting abstraction model when given new noisy output trajectories, which provides a solution to Problem 1.3. Together, the two algorithms we propose provide a sufficient solution to Problem 1.

A. Data-Driven Abstraction Algorithm

In this section, we assume for the moment that the Jacobian bounds in Assumption 1 are given. We will consider the case when the bounds are unknown in Section IV-C.

Theorem 1: Consider system (5) and its corresponding data set $\overline{\mathcal{D}} = \bigcup_{\ell=1}^N \{(\tilde{s}_{j,\ell}, \tilde{y}_{j+1,\ell}) | j = n_y, \ldots, T_\ell - 1\}$. Suppose Assumption 1 holds. Then, for all $s \in \mathcal{S}$, $\underline{f}(\cdot)$ and $\overline{f}(\cdot)$ are lower and upper abstraction functions for unknown function $f(\cdot)$, i.e., $\forall s \in \mathcal{S}$, $\underline{f}(s) \le f(s) \le \overline{f}(s)$,

$$\overline{f}(s) = \min_{\substack{j\in\{n_y,\ldots,T_\ell-1\},\\ \ell\in\{1,\ldots,N\}}} (\tilde{y}_{j+1,\ell} + \overline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) + \varepsilon_v + \varepsilon_w), \tag{11a}$$

$$\underline{f}(s) = \max_{\substack{j\in\{n_y,\ldots,T_\ell-1\},\\ \ell\in\{1,\ldots,N\}}} (\tilde{y}_{j+1,\ell} + \underline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) - \varepsilon_v - \varepsilon_w), \tag{11b}$$

where $\overline{g}(\cdot)$ and $\underline{g}(\cdot)$ are as defined in (3) or (4) in Proposition 2, J_u and J_l are (known) bounds of Jacobian matrix $\nabla f(\cdot)$ and $\overline{\Delta s}_{j,\ell}$ and $\underline{\Delta s}_{j,\ell}$ are defined as $s - \tilde{s}_{j,l} + \varepsilon_s$ and $s - \tilde{s}_{j,l} - \varepsilon_s$ respectively.

Proof: From the mean value theorem, for any $s \in \mathcal{S}$, we have $f(s) = f(s_{j,\ell}) + \nabla f(s')(s - s_{j,\ell})$ for some $s' \in [\min(s, s_{j,\ell}), \max(s, s_{j,\ell})]$. Since the Jacobian matrix $\nabla f(\cdot)$ is bounded by $[J_l, J_u]$ as assumed in Assumption 1 and $s - s_{j,\ell}$ is within the interval $[s - \tilde{s}_{j,\ell} - \varepsilon_s, s - \tilde{s}_{j,\ell} + \varepsilon_s]$ according to definition of $\tilde{s}_{j,\ell}$, we have $\underline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) \le \nabla f(s')(s - s_{j,\ell}) \le \overline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell})$, by applying Proposition 2. Furthermore, from (5) and (6), we have $f(s_{j,\ell}) = y_{j+1,\ell} - w_{j,\ell} = \tilde{y}_{j+1,\ell} - w_{j,\ell} - v_{j,\ell}$. Therefore, $\tilde{y}_{j+1,\ell} + \underline{g}(\cdot) - w_{j,\ell} - v_{j,\ell} \le f(s) \le \tilde{y}_{j+1,\ell} + \overline{g}(\cdot) + w_{j,\ell} + v_{j,\ell}$. Finally, since these inequalities must hold for *all* the sampled data and *all* the possible values of noise signals to obtain a model abstraction, we obtain (11a) and (11b). ■

Proposition 3: Given two data sets $\mathcal{D}$ and $\mathcal{D}'$, if $\mathcal{D}' \subseteq \mathcal{D}$, then $\mathcal{H}_\mathcal{D}$ is closer to the unknown model $\mathcal{G}$ than $\mathcal{H}_{\mathcal{D}'}$ in the sense that the abstraction model $\mathcal{H}_\mathcal{D}$ is over-approximated by the abstraction model $\mathcal{H}'_\mathcal{D}$.

Proof: Let $\mathcal{K}$ and $\mathcal{K}'$ be the set of indices corresponding to data pairs included in $\overline{\mathcal{D}}$ and $\overline{\mathcal{D}'}$ (constructed from $\mathcal{D}$ and $\mathcal{D}'$) and $\overline{f}(s)$ and $\overline{f'}(s)$, and $\underline{f}(s)$ and $\underline{f}'(s)$ are upper and lower abstraction functions returned by the abstraction models $\mathcal{H}_\mathcal{D}$ and $\mathcal{H}_{\mathcal{D}'}$, respectively. Then, $\mathcal{D}' \subseteq \mathcal{D} \implies \overline{\mathcal{D}'} \subseteq \overline{\mathcal{D}} \implies \mathcal{K}' \subseteq \mathcal{K} \implies \overline{f}(s) = \min_{j,\ell\in\mathcal{K}}(\tilde{y}_{j+1,\ell} + \overline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) + \varepsilon_v + \varepsilon_w) \le \min_{j,\ell\in\mathcal{K}'}(\tilde{y}_{j+1,\ell} + \overline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) + \varepsilon_v + \varepsilon_w) = \overline{f'}(s) \implies \overline{f}(s) \le \overline{f}'(s)$, where the second inequality holds since the two optimization problems have the same objective function, but the constraint set of the latter is a subset of the former. Similarly, we can obtain $\underline{f}(s) \ge \underline{f}'(s)$. According to the definition of abstractions, $\mathcal{H}_{\mathcal{D}'}$ over-approximates $\mathcal{H}_\mathcal{D}$. ■

This proposition implies that by increasing the number of sample points, the abstraction model will become equally or more accurate, which in turn implies that downsampling strategies, e.g., in [14], can be beneficial to improve computational time at the expense of slightly less accurate models.

Next, in preparation for comparing our result with (componentwise) Lipschitz interpolation methods [14], [16], we recap the previous approaches in the following proposition.

Proposition 4 [(Componentwise) Lipschitz Interpolation [14], [16]]: Consider system (5) and its corresponding data set $\overline{\mathcal{D}} = \bigcup_{\ell=1}^N \{(\tilde{s}_{j,\ell}, \tilde{y}_{j+1,\ell}) | j = n_y, \ldots, T_\ell - 1\}$. Suppose $f(\cdot)$ is (componentwise) Lipschitz continuous. Then, $\underline{f}_L^{(i)}(\cdot)$, $\overline{f}_L^{(i)}(\cdot)$ and $\underline{f}_{CL}^{(i)}(\cdot)$, $\overline{f}_{CL}^{(i)}(\cdot)$ for all $i \in \{1, \ldots, m\}$ are lower and upper Lipschitz and componentwise Lipschitz abstraction functions for the unknown function $f^{(i)}(\cdot)$, i.e., $\forall s \in \mathcal{S}$, $\underline{f}_L^{(i)}(s) \le f^{(i)}(s) \le \overline{f}_L^{(i)}(s)$ and $\underline{f}_{CL}^{(i)}(s) \le f^{(i)}(s) \le \overline{f}_{CL}^{(i)}(s)$, respectively, where for $* \in \{L, CL\}$,

$$\overline{f}_*^{(i)}(s) = \min_{\substack{j\in\{n_y,\ldots,T_\ell-1\},\\ \ell\in\{1,\ldots,N\}}} (\tilde{y}_{j+1,\ell}^{(i)} + \varphi_*^{(i)}(s^{(i)}, \tilde{s}_{j,\ell}^{(i)}) + \varepsilon_*^{(i)}, \tag{12a}$$

$$\underline{f}_*^{(i)}(s) = \max_{\substack{j\in\{n_y,\ldots,T_\ell-1\},\\ \ell\in\{1,\ldots,N\}}} (\tilde{y}_{j+1,\ell}^{(i)} - \varphi_*^{(i)}(s^{(i)}, \tilde{s}_{j,\ell}^{(i)}) - \varepsilon_*^{(i)}, \quad (12b)$$

with $p = \{1, \infty\}$, $\varphi_L^{(i)}(s, \tilde{s}_{j,\ell}) \triangleq L_p^{(i)}\|s - \tilde{s}_{j,\ell}\|_p$, $\varepsilon_L^{(i)} \triangleq \varepsilon_w + \varepsilon_v + L_p^{(i)}\|\varepsilon_s\|_p$, $\varphi_{CL}^{(i)}(s^{(i)}, \tilde{s}_{j,\ell}^{(i)}) \triangleq \sum_{k=1}^n L_c^{(i,k)}|s^{(k)} - \tilde{s}_{j,\ell}^{(k)}|$ and $\varepsilon_{LC}^{(i)} \triangleq \varepsilon_w + \varepsilon_v + \sum_{k=1}^n L_c^{(i,k)}|\varepsilon_s^{(k)}|$, where $L_p^{(i)} > 0$ is the functionwise scalar Lipschitz constant and $L_c^{(i,k)} > 0$ are the functionwise, componentwise Lipschitz constants.

Then, we show that (componentwise) Lipschitz abstractions are special cases of our proposed Jacobian abstractions and that the Jacobian abstractions enhance methods based on (componentwise) Lipschitz abstractions.

Lemma 1: Suppose $\varepsilon_s = 0$. If the Jacobian matrix bounds are $J_u = -J_l = L_c$, then $\underline{f}, \overline{f}$ in Theorem 1 are equal to the componentwise Lipschitz abstraction functions $\underline{f}_{CL}, \overline{f}_{CL}$. Further, if $J_u = -J_l = L_1\mathbf{1}_{m\times n}$, then $\underline{f}, \overline{f}$ in Theorem 1 are equal to the Lipschitz abstraction functions $\underline{f}_L, \overline{f}_L$ when applied using the 1-norm.

Proof: For arbitrary $i \in \{1, \ldots, m\}$, $j \in \{n_y, \ldots, T_\ell - 1\}$ and $\ell \in \{1, \ldots, N\}$,

$$\overline{g}^{(i)}(\cdot) = \sum_{k=1}^n \max\{J_u^{(i,k)}\overline{\Delta s}_{j,\ell}^{(k)}, J_u^{(i,k)}\underline{\Delta s}_{j,\ell}^{(k)}, J_l^{(i,k)}\overline{\Delta s}_{j,\ell}^{(k)}, J_l^{(i,k)}\underline{\Delta s}_{j,\ell}^{(k)}\}. \quad (13)$$

Then, with $\varepsilon_s = 0$ and $J_u = -J_l = L_c$, we have $\underline{\Delta s}_{j,\ell}^{(k)} = \overline{\Delta s}_{j,\ell}^{(k)}$ and (13) reduces to $\varphi_{CL}^{(i)}$ in Proposition 4. Similarly, $\underline{g}^{(i)}(\cdot)$ reduces to $-\varphi_{CL}^{(i)}$; thus, it is equivalent to the componentwise Lipschitz case. The equivalence to the Lipschitz case can be obtained by combining the above and the result from [16, Lemma. 2]. ■

Theorem 2: The Jacobian abstraction function using Theorem 1 (with (3) in Proposition 2) is always tighter than or equal to the (componentwise) Lipschitz abstraction functions in Proposition 4, i.e.,

$$\underline{f}_L(s) \le \underline{f}_{CL}(s) \le \underline{f}(s) \le f(s) \le \overline{f}(s) \le \overline{f}_{CL}(s) \le \underline{f}_L(s).$$

Moreover, if $|J_l^{(i,k)}| < L_c^{(i,k)}$ or $|J_u^{(i,k)}| < L_c^{(i,k)}$ for any $i \in \{1, \ldots, m\}$, $k \in \{1, \ldots, n\}$, the Jacobian abstraction is always tighter than (componentwise) Lipschitz abstraction unless $|\overline{\Delta s}_{j,\ell}^{(k)}| = |\underline{\Delta s}_{j,\ell}^{(k)}| = 0$ (a measure zero set).

Proof: For arbitrary $i \in \{1, \ldots, m\}$, $j \in \{n_y, \ldots, T_\ell - 1\}$ and $\ell \in \{1, \ldots, N\}$, from (13),

$$\overline{g}^{(i)}(\cdot) \le \sum_{k=1}^n \max\{|J_u^{(i,k)}||\overline{\Delta s}_{j,\ell}^{(k)}|, |J_u^{(i,k)}||\underline{\Delta s}_{j,\ell}^{(k)}|, |J_l^{(i,k)}||\overline{\Delta s}_{j,\ell}^{(k)}|, |J_l^{(i,k)}||\underline{\Delta s}_{j,\ell}^{(k)}|\}. \quad (14)$$

Since $|J_l^{(i,k)}| \le L_c^{(i,k)}$ and $|J_u^{(i,k)}| \le L_c^{(i,k)}$ based on the definition s of the Jacobian matrix and componentwise Lipschitz constants, then $\overline{g}^{(i)}(J_u^{(i)}, J_l^{(i)}, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) \le \sum_{k=1}^n \max\{L_c^{(i,k)}|\overline{\Delta s}_{j,\ell}^{(k)}|, L_c^{(i,k)}|\underline{\Delta s}_{j,\ell}^{(k)}|\}$. Next, by applying triangular inequality, we obtain the fact that both $L_c^{(i,k)}|s^{(k)} - \tilde{s}_{j,l}^{(k)} + \varepsilon_s^{(k)}|$ and $L_c^{(i,k)}|s^{(k)} - \tilde{s}_{j,l}^{(k)} - \varepsilon_s^{(k)}|$ are smaller than $L_c^{(i,k)}|s^{(k)} - \tilde{s}_{j,l}^{(k)}| + L_c^{(i,k)}|\varepsilon_s^{(k)}|$. Therefore, $\overline{g}^{(i)}(J_u^{(i)}, J_l^{(i)}, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) \le \sum_{k=1}^n (L_c^{(i,k)}|s^{(k)} - \tilde{s}_{j,l}^{(k)}| + L_c^{(i,k)}|\varepsilon_s^{(k)}|)$. Finally, by adding $\tilde{y}_{j+1,\ell}^{(i)} + \varepsilon_v + \varepsilon_w$ to both sides of inequality and leveraging the fact that the inequality holds for all $i \in \{1, \ldots, m\}$, $j \in \{n_y, \ldots, T_\ell - 1\}$ and $\ell \in \{1, \ldots, N\}$, we can conclude that $\overline{f}(s) \le \overline{f}_{CL}(s)$. Similarly, we can obtain $\underline{f}(s) \ge \underline{f}_{CL}(s)$. Moreover, by [16, Corollary 2], $\underline{f}_L(s) \le \underline{f}_{CL}(s) \le f(s) \le \overline{f}_{CL}(s) \le \overline{f}_L(s)$. Further, we can deduce that Jacobian abstractions are generally tighter than (componentwise) Lipschitz abstractions from (14). ■

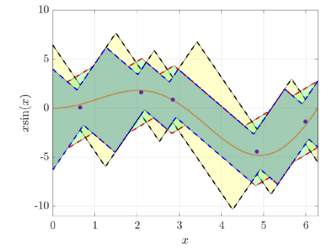


Fig. 1. Comparison of abstractions of $y = s\sin(s)$ with $\varepsilon_s = \varepsilon_v = 0.4$. The dark green region enclosed by the blue dashed lines is obtained by Theorem 1 with (3) in Proposition 2 and the light green region within the red dashed lines is obtained by Theorem 1 with (4), with $[J_l, J_u] = [-3.7, 6.28]$, while the yellow region within the black dashed lines is obtained by the (componentwise) Lipschitz interpolation method in [14], [16] with $L = 6.28$.

Figure 1 illustrates the results of our proposed abstraction method for the function $y = s\sin(s)$ when compared to the (componentwise) Lipschitz interpolation method in Proposition 4. From the figure, we observe that the abstraction is the tightest when adopting (3) from Proposition 2 in Theorem 1. However, when using (4) from Proposition 2 in Theorem 1, the abstraction model is less accurate in the regions close to the sampled points than the other two methods.

B. Data-Driven Model Invalidation Algorithm

Next, we propose an optimization-based model invalidation approach to determine if the data-driven abstraction obtained in the previous section is incompatible with a newly observed length-T output trajectory, i.e., we propose a model invalidation algorithm for an abstracted model $\mathcal{H}$ as follows.

Theorem 3: Given an abstracted model $\mathcal{H}$, a newly observed length-T output sequence $\{\tilde{y}_k^n\}_{k=0}^{k=T-1}$ invalidates model $\mathcal{H}$, if the following MILP problem is infeasible:

$$\begin{aligned}
&\text{Find } y_k, w_k, v_k\ \forall k \in \mathbb{Z}_{T-1}^0\\
&\text{subject to } \forall k \in \mathbb{Z}_{T-1}^{n_y}, \forall \ell \in \mathbb{Z}_N^1, \forall(\tilde{s}_{j,\ell}, \tilde{y}_{j+1,\ell}) \in \overline{\mathcal{D}}_\ell:\\
&y_{k+1} \le \tilde{y}_{j+1,\ell} + \overline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell,k}, \underline{\Delta s}_{j,\ell,k}) + w_k, \quad (15a)\\
&y_{k+1} \ge \tilde{y}_{j+1,\ell} + \underline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell,k}, \underline{\Delta s}_{j,\ell,k}) + w_k, \quad (15b)\\
&\forall k \in \mathbb{Z}_{T-1}^0: \tilde{y}_k^n = y_k + v_k,\ \underline{y}_k \le y_k \le \overline{y}_k, \quad (15c)\\
&-\varepsilon_w \mathbb{1}_m \le w_k \le \varepsilon_w \mathbb{1}_m, -\varepsilon_v \mathbb{1}_m \le v_k \le \varepsilon_v \mathbb{1}_m, \quad (15d)
\end{aligned}$$

where $\overline{\mathcal{D}_\ell} = \{(\tilde{s}_{j,\ell}, \tilde{y}_{j+1,\ell}) | j = n_y, \ldots, T_\ell - 1\}$ is a trajectory of $\overline{\mathcal{D}}$, $\overline{\mathcal{D}} = \bigcup_{\ell=1}^N \overline{\mathcal{D}}_\ell$ is the given sampled data set from which we obtain a data-driven abstraction $\mathcal{H}$ with $\underline{y}_k$ and $\overline{y}_k$ as given bounds of y_k and $s_k = [y_k, \ldots, y_{k-n_y+1}]^T$. Moreover, $\overline{\Delta s}_{j,\ell,k}$ and $\underline{\Delta s}_{j,\ell,k}$ are defined as $s_k - \tilde{s}_{j,l} + \varepsilon_s$ and $s_k - \tilde{s}_{j,l} - \varepsilon_s$

respectively, while the definition of $\overline{g}(\cdot)$ and $\underline{g}(\cdot)$ can be found in Proposition 2.

Proof: By the definition of the model invalidation problem for the abstracted model $\mathcal{H}$, we know that the abstraction is invalidated if the following problem is infeasible:

$$\begin{aligned} &\text{Find} \quad y_k, w_k, v_k \ \forall k \in \mathbb{Z}_{T-1}^0 && \text{(MI)}\\ &\text{subject to} \quad \forall k \in \mathbb{Z}_{T-1}^{n_y} : y_{k+1} \le \overline{f}(s_k) + w_k, && \text{(16a)}\\ &\quad y_{k+1} \ge \underline{f}(s_k) + w_k, && \text{(16b)}\\ &\quad \forall k \in \mathbb{Z}_{T-1}^{0} : {}^{n}_{k} = y_k + v_k, \ \underline{y}_k \le y_k \le \overline{y}_k, && \text{(16c)}\\ &\quad -\varepsilon_w \mathbb{1}_m \le w_k \le \varepsilon_w \mathbb{1}_m, -\varepsilon_v \mathbb{1}_m \le v_k \le \varepsilon_v \mathbb{1}_m. && \text{(16d)} \end{aligned}$$

Since the upper bound of the abstraction is given by (11a), constraint (16a) is equivalent to (15a). Similarly, by (11b), (15b) is equivalent to (16b). Thus, two optimization problems are equivalent. If the above optimization problem is infeasible, it means that the output sequence $\{\tilde{y}_k^n\}_{k=0}^{k=T-1}$ cannot be consistent with the length-T behavior of $\mathcal{H}$, i.e., $\{\tilde{y}_k^n\}_{k=0}^{k=T-1} \notin \mathcal{B}^T(\mathcal{H})$, hence the model is invalidated. ■

Note that the min and max operators in functions $\overline{g}(\cdot)$ and $\underline{g}(\cdot)$ are encoded as integer constraints in the optimization problem, which may generate a large number of integer constraints that lead to intractability. Hence, to help reduce the computation complexity for large problems, the use of (4) in Proposition 2 is preferable to (3) for the functions $\overline{g}(\cdot)$ and $\underline{g}(\cdot)$ in Theorem 3. Although the abstraction model may be less tight, the number of integer constraints can be reduced. Another method to reduce computation cost is to rely on Corollary 1, where we only need to compare two instead of four values to find the min and max for each multiplication since the Jacobian bounds J_u and J_l are known.

C. Estimation of Jacobian Bounds

In the previous sections, the Jacobian bounds are assumed to be given. However, in case that the bounds are not known, we can estimate the Jacobian bounds from the noisy sampled data set $\overline{\mathcal{D}} = \{(\tilde{s}_j, \tilde{y}_{j+1}) | j = n_y, \ldots, N-1\}$ by solving the following mixed-integer linear program (MILP):

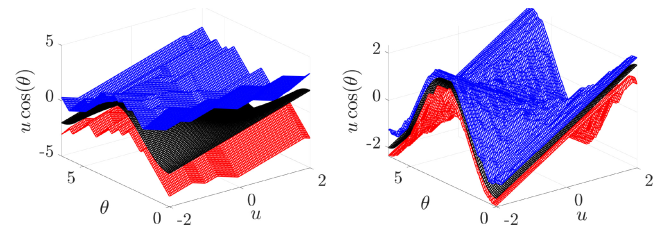
$$\min_{J_u, J_l} \sum_{i=1}^{m} \overline{g}^{(i)}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) - \underline{g}^{(i)}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell})$$

$$\begin{aligned} &\text{subject to} \quad \forall j, \ell \in \{n_y, \ldots, N-1\}, \ j \ne \ell: \\ &\quad \tilde{y}_{j+1} - \tilde{y}_{\ell+1} \le \overline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) + 2\varepsilon_v, && \text{(17a)}\\ &\quad \tilde{y}_{j+1} - \tilde{y}_{\ell+1} \ge \underline{g}(J_u, J_l, \overline{\Delta s}_{j,\ell}, \underline{\Delta s}_{j,\ell}) - 2\varepsilon_v, && \text{(17b)}\\ &\quad J_u \ge J_l, && \text{(17c)} \end{aligned}$$

with $\overline{g}(\cdot)$ and $\underline{g}(\cdot)$ defined as in either (3) or (4) in Proposition 4, $\overline{\Delta s}_{j,\ell} \triangleq \tilde{s}_j - \tilde{s}_\ell + 2\varepsilon_s$ and $\underline{\Delta s}_{j,\ell} \triangleq \tilde{s}_j - \tilde{s}_\ell - 2\varepsilon_s$.

Equations (17a) and (17b) ensure that the Jacobian bounds hold for all sampled points in the data set $\overline{\mathcal{D}}$ where both the input and output data, i.e., $\tilde{s}_j$ and $\tilde{y}_{j+1}$ for all j, are corrupted by bounded noise. The above expression can be simply obtained from the same reasoning as in Theorem 1. However, the optimization problem becomes intractable when the data set is large. In this case, we propose to solve the problem incrementally (using subsets of the data at each iteration similar to the approach in [19]) by adding the following constraint from the previous iteration in (17):

$$J_l \le \hat{J}_l, \ J_u \ge \hat{J}_u, \tag{18}$$



(a) Abstraction with 10 points (b) Abstraction with 800 points

Fig. 2. Illustration of abstractions with different sizes of data set for function $u\cos(\theta)$ using (3). (a) Abstraction with 10 points (b) Abstraction with 800 points.

where $\hat{J}_u$ and $\hat{J}_l$ are previously estimated Jacobian bounds.

Since the accuracy of J_u and J_l are crucial for the results in the previous section, we proceed to find some guarantees that we obtain the right estimate with high probability. To achieve this, we leverage a classical result on *probably approximately correct* (PAC) learning for linear separators.

Proposition 5 (PAC Learning [20]): Let $\epsilon, \delta \in \mathbb{R}^+$. Suppose the number of sampling points, N, satisfies $N \ge \frac{1}{\epsilon} \ln \frac{1}{\delta}$ and the sample points Γ are drawn from a probability distribution $\mathcal{P}$. Then, with a probability greater than $1 - \delta$, a linear separator (a, b) (such that $\forall (x, y) \in \Gamma : x \le ay + b$) has an error $\texttt{err}_{\mathcal{P}}$ of less than ϵ, where the error of the separator (a, b) is defined as $\texttt{err}_{\mathcal{P}}(a, b) = \Pr((x, y) \in \Gamma | x > ay + b)$.

It is easy to verify that our estimate of Jacobian bounds in (17) is a special case of the above linear separator with $b = 0$. Thus, the estimated $\hat{J}_u$ and $\hat{J}_l$ using (17) is guaranteed to be close to the true Jacobian bounds of the original unknown function with high probability if we have sufficient data.

V. Simulation and Discussion

In this section, we compare the effectiveness of our proposed Methods I and II (Theorem 1 using (4) and (3)) with Method III in [14] (Lipschitz interpolation) and Method IV in [16] (componentwise Lipschitz interpolation) for data-driven abstraction and model invalidation. Both data-driven abstraction and model invalidation algorithms are implemented in MATLAB with Yalmip and Gurobi on a 2.2 GHz machine with 16 GB RAM.

A. Data-Driven Abstraction

To demonstrate the data-driven abstraction approach described in Theorem 1 with (3) as $\overline{g}(\cdot)$ and $\underline{g}(\cdot)$, we first consider the nonlinear functions in the Dubins car dynamics, i.e., the $u\cos(\theta)$ and $u\sin(\theta)$ terms. For illustrative purposes, in Figure 2, we show the data-driven abstraction (Method II) results for the function $f(u, \theta) = u\cos(\theta)$ defined on the domain $u \times \theta \in [-2, 2] \times [0, 2\pi]$, where we additionally assume that u is measured with a noise bound of 0.1, and similarly, the bounds of the noise of measuring θ and $f(u, \theta)$ are also assumed to be 0.1. As expected, the resulting abstraction is indeed an over-approximation of the unknown nonlinear function on the defined domains of interest. Moreover, the accuracy of the over-approximation improves with increasing number of data points in $\mathcal{D}$ as proved in Proposition 3.

Next, we consider the Rastrigin's function [21]:

$$f(x) = 10d + \sum_{j=1}^{d} [x_j^2 - 10\cos(2\pi x_j)] \tag{19}$$

TABLE I
APPROXIMATED ABSTRACTION ERRORS FOR VARYING DIMENSIONS

Dimension (d)	Data Size	Method II (Thm. 1 using (3))	Method IV [16]
2	100	9.74×10^4	1.06×10^5
3	1000	6.55×10^6	7.17×10^6
4	10^4	3.94×10^8	4.34×10^8

TABLE II
MAXIMUM TIME STEPS AND AVERAGE SOLVER TIME TAKEN FOR VARIOUS SIZE OF DATA SET IN 20 TRAJECTORIES

	Size of data set, N	160	176	192	208
(i) Method I (Thm. 1 using (4))	t_{max}	15	14	14	14
	Solver Time (s)	0.11	0.14	0.19	0.20
(ii) Method II (Thm. 1 using (3))	t_{max}	15	14	14	13
	Solver Time (s)	0.43	0.92	1.23	1.31
(iii) Method III [14]	t_{max}	16	16	15	13
	Solver Time (s)	0.11	0.15	0.19	0.30
(iv) Method IV [16]	t_{max}	16	15	14	13
	Solver Time (s)	0.10	0.12	0.16	0.19

where $x = [x_1, \ldots, x_d]^T \in \mathbb{R}^d$ with d being the dimension of state x. We also assume that $x_i \in [0, 5.1]$ for all $i \in \{1, \ldots, d\}$. To compare Jacobian and Lipschitz abstractions, we consider the volume between upper and lower functions as the abstraction error. For simplicity, we approximate the abstraction errors as the sum of the distances between the functions on a uniform grid with 200 grid points in each dimension for the results in Table I, where we observe that Jacobian abstractions are better than componentwise Lipschitz interpolations for all cases with varying dimensions.

B. Application to Model Discrimination of Swarm Intents

We adopt the swarm intent identification example in [14] with three swarm intent models (see [14] for details): the swarm intends to move towards the centroid of the swarm and the sampled data set $\mathcal{D}$ (with size $|\mathcal{D}|$) is collected using this model. Meanwhile, we consider another swarm intent that moves away from the centroid and use this model to generate the new observed trajectories $\{\tilde{y}_k^n\}_{k=0}^{T-1}$ to invalidate the abstraction based on the sampled data set $\mathcal{D}$.

Next, using the data-driven abstractions of the unknown dynamics based on the sampled data $\mathcal{D}$, we compare the model invalidation performances when using Methods I–IV. To compare these methods, we vary the size of the data set, $|\mathcal{D}|$, from 160 to 208, and compare the maximum steps the data-driven model invalidation algorithms take to invalidate the (wrong) model using 20 randomly generated noisy newly observed trajectories from the true model. As shown in Table II, when the data set size is small, the (wrong) model can be invalidated by Method I within fewer time steps than Method III but when the data set size is increased, Method I is not always better. Further, the average solver time of Method I is comparable or shorter than (componentwise) Lipschitz interpolation approaches (Methods III and IV), since we employed (4) when applying Theorem 1 to reduce computational complexity. When employing (3) in Theorem 1 (Method II), our method always takes fewer or equal time steps than (componentwise) Lipschitz interpolation approaches (Methods III and IV), as expected in light of Theorem 2, but at the cost of slightly longer solver times.

VI. CONCLUSION

A data-driven model invalidation approach was proposed in this letter for bounded-error unknown nonlinear discrete-time dynamical systems with bounded Jacobians, where instead of given mathematical models, only prior noisy sampled data of the systems are available. The data-driven model invalidation problem was addressed by leveraging a tractable feasibility check. Further, when the Jacobian bounds are not given or known, we proposed an approach to estimate them. Supported by theory, our simulations showed that the proposed approach generally finds tighter abstractions than (componentwise) Lipschitz abstractions, and the model invalidation problem also takes fewer time steps to invalidate abstraction models obtained using our proposed approach.

REFERENCES

[1] R. S. Smith and J. C. Doyle, "Model validation: A connection between robust control and identification," *IEEE Trans. Autom. Control*, vol. 37, no. 7, pp. 942–952, Jul. 1992.
[2] F. Harirchi, S. Z. Yong, and N. Ozay, "Guaranteed fault detection and isolation for switched affine models," in *Proc. IEEE Conf. Decis. Control*, 2017, pp. 5161–5167.
[3] Z. Jin, Q. Shen, and S. Z. Yong, "Optimization-based approaches for affine abstraction and model discrimination of uncertain nonlinear systems," in *Proc. IEEE Conf. Decis. Control*, 2019, pp. 7976–7981.
[4] A. Girard, "Approximately bisimilar finite abstractions of stable linear systems," in *Proc. ACM Int. Conf. Hybrid Syst. Comput. Control*, 2007, pp. 231–244.
[5] A. Girard and S. Martin, "Synthesis for constrained nonlinear systems using hybridization and robust controller on symplices," *IEEE Trans. Autom. Control*, vol. 57, no. 4, pp. 1046–1051, 2012.
[6] K. Singh, Q. Shen, and S. Z. Yong, "Mesh-based affine abstraction of nonlinear systems with tighter bounds," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 3056–3061.
[7] V. Alimguzhin, F. Mari, I. Melatti, I. Salvo, and E. Tronci, "Linearizing discrete-time hybrid systems," *IEEE Trans. Autom. Control*, vol. 62, no. 10, pp. 5357–5364, Oct. 2017.
[8] M. Milanese and C. Novara, "Set membership identification of nonlinear systems," *Automatica*, vol. 40, no. 6, pp. 957–975, 2004.
[9] M. Canale, L. Fagiano, and M. Signorile, "Nonlinear model predictive control from data: A set membership approach," *Int. J. Robust Nonlinear Control*, vol. 24, no. 1, pp. 123–139, 2014.
[10] S. Sadraddini and C. Belta, "Formal guarantees in data-driven model identification and control synthesis," in *Proc. Int. Conf. Hybrid Syst. Comput. Control*, 2018, pp. 147–156.
[11] A. Devonport, A. Saoud, and M. Arcak, "Symbolic abstractions from data: A PAC learning approach," in *IEEE Conf. Decis. Control (CDC)*, 2021, pp. 599–604.
[12] Z. Zabinsky, R. Smith, and B. Kristinsdottir, "Optimal estimation of univariate black-box Lipschitz functions with upper and lower error bounds," *Comput. Oper. Res.*, vol. 30, no. 10, pp. 1539–1553, 2003.
[13] G. Beliakov, "Interpolation of Lipschitz functions," *J. Comput. Appl. Math.*, vol. 196, no. 1, pp. 20–44, 2006.
[14] Z. Jin, M. Khajenejad, and S. Z. Yong, "Data-driven model invalidation for unknown Lipschitz continuous systems via abstraction," in *Proc. Amer. Control Conf. (ACC)*, 2020, pp. 2975–2980.
[15] J. Calliess, "Conservative decision-making and inference in uncertain dynamical systems," Ph.D. dissertation, Dept. Eng. Sci., Univ. Oxford, Oxford, U.K., 2014.
[16] J. Manzano, D. de la Pena, J.-P. Calliess, and D. Limon, "Componentwise Hölder inference for robust learning-based MPC," *IEEE Trans. Autom. Control*, vol. 66, no. 11, pp. 5577–5583, Nov. 2021.
[17] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, "Interval analysis," in *Applied Interval Analysis*. London, U.K.: Springer, 2001, pp. 11–43.
[18] M. Khajenejad, Z. Jin, and S. Z. Yong, "Resilient interval observer for simultaneous estimation of states, modes, and attack policies," in *Proc. Amer. Control Conf. (ACC)*, 2022, pp. 322–329.
[19] S. Hassaan, M. Khajenejad, S. Jensen, Q. Shen, and S. Z. Yong, "Incremental affine abstraction of nonlinear systems," *IEEE Control Syst. Lett.*, vol. 5, no. 2, pp. 653–658, Apr. 2021.
[20] M. Kearns, U. Vazirani, and U. Vazirani, *An Introduction to Computational Learning Theory*. Cambridge, MA, USA: MIT Press, 1994.
[21] H. Pohlheim. "GEATbx Examples of Objective Functions." 2005. [Online]. Available: http://www.geatbx.com/