

# End-to-End Imitation Learning with Safety Guarantees using Control Barrier Functions

Ryan K. Cosner, Yisong Yue, Aaron D. Ames

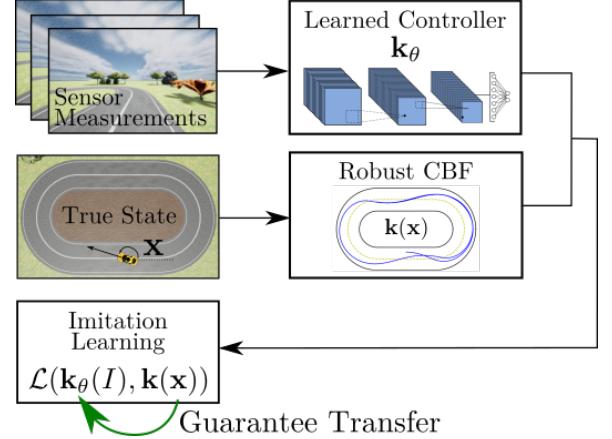
**Abstract**—Imitation learning (IL) is a learning paradigm which can be used to synthesize controllers for complex systems that mimic behavior demonstrated by an expert (user or control algorithm). Despite its popularity, IL methods generally lack guarantees of safety, which limits their utility for complex safety-critical systems. In this work we consider safety, formulated as set-invariance, and the associated formal guarantees endowed by Control Barrier Functions (CBFs). We develop conditions under which robustly-safe expert controllers, utilizing CBFs, can be used to learn end-to-end controllers (which we refer to as *CBF-Compliant* controllers) that have safety guarantees. These guarantees are presented from the perspective of input-to-state safety (ISSf) which considers safety in the context of disturbances, wherein it is shown that IL using robustly safe expert demonstrations results in ISSf with the disturbance directly related to properties of the learning problem. We demonstrate these safety guarantees in simulated vision-based end-to-end control of an inverted pendulum and a car driving on a track.

## I. INTRODUCTION

The use of learning in conjunction with control has become an increasingly popular area of study—especially in the context of autonomous and robotic systems where system properties, e.g., stability and safety, must generalize to the real world. Of particular interest in this paper, Imitation Learning (IL) trains a policy to mimic behavior demonstrated by an expert [1]. IL is an appealing paradigm that has shown significant success in video-games [2], humanoid robotics [3], and autonomous vehicles [4], [5]. The goal of this paper is to extend the theoretic underpinnings of safety to IL applications.

Safety, framed as the forward-invariance of a *safe set*, has become a dominant definition in modern control theory. Several methods have been introduced which provide such guarantees of safety including model predictive control (MPC) [6], optimal reachability-based methods [7], and control barrier functions (CBFs) [8], [9], [10]. Control Barrier Function methods use a Lyapunov-like condition to guarantee safety and present advantages over other methods in that they provide guarantees for general continuous-time nonlinear systems and can be implemented efficiently as convex optimization programs. The robustness of CBF-based safety methods have also been studied in the context of state uncertainty [11], dynamics uncertainty [12], and reduced-order models [13].

With the goal deploying machine learning to real-world systems where safety is critical, there has been a large body of recent work exploring machine learning’s connection to



**Fig. 1.** The robust control barrier function-based controller  $k$  is used as the expert controller in an imitation learning problem to generate the end-to-end controller  $k_\theta$ . By ensuring that the expert controller  $k$  has robust safety guarantees, we ensure that  $k_\theta$  is also guaranteed to keep the system safe.

safety-critical control. These approaches have largely focused on the forward-invariance notion of safety considered by the control community. In particular, in this setting learning is often applied when there are uncertainty associated with the dynamics, i.e., there is uncertain residual dynamics. This can be included in MPC-based controllers and used to learn constraints and cost functions [14]. Learning has also been used in the context of optimal control to ensure safety in the context of reinforcement learning [15]. Finally, one can then learn this residual with a view toward ensuring safety in the context of CBFs, either on the full-order dynamics [16], [17] or in the projection of the uncertainty to the CBF [18], [19].

Specifically in the context of IL, safety of learned controllers has been studied from several perspectives. In the simplest setting, the controller can be disengaged whenever it is deemed unsafe by a human operator [5]. By defining safety as the deviation from the expert controller, several methods such as SafeDAgger [20] and EnsembleDAgger [21] can achieve this form of safety by returning control of the system to the expert whenever an uncertainty threshold is reached. This perspective on safety is different from set-invariance as it may be possible to leave a safe set with only small deviations from the expert controller. Alternatively, forward-invariance for systems with IL-based controllers for discrete-time systems has been studied in the context Lyapunov stability [22] and MPC [23]. Success of IL-based controllers learned using CBF-based demonstrations has been shown in simulation without guarantees [24]. Yet theoretical connections between IL and CBF-based safety guarantees for have not been established.

This paper presents an end-to-end IL method that endows the learned controller with formal guarantees of safety, as illustrated in Fig. 1. The key idea is that if the expert controller

This research is supported by BP and AeroVironment.

R.K. Cosner and A.D. Ames are with the Department of Mechanical and Civil Engineering and Y. Yue is with the Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, CA, 91125, USA, {rkcosner, yyue, ames}@caltech.edu. Y. Yue is also affiliated with Argo AI, Pittsburgh, PA.

has robust CBF-based safety guarantees and the proper sampling scheme is employed then it is possible to obtain safety guarantees for the IL-synthesized controllers, we call these learned controllers **CBF-compliant**. These results are framed in the context of robust safety, and specifically input-to-state safety (ISSf), wherein we established connections between the learning methods, sampling density, network smoothness, and learning accuracy. This manifests itself in the main result of the paper, which gives a formal guarantee of safety (in an ISSf context) where properties of the learning problem directly affect the corresponding robust safe set. This result is demonstrated in simulation on two examples: an inverted pendulum and a vehicle driving on a circular track. In both cases, safe vision-based control is demonstrated using our end-to-end safe imitation learning methodology, where end-to-end refers to direct perception-to-control.

The structure of the paper is as follows. In Section II we setup the problem by giving an overview of imitation learning and safety-critical control with CBFs. The notion of robust safety (via ISSf) is introduced in Section III along with a discussion of continuity properties of the safe set—these concepts will allow us to quantify how uncertainty in the imitation learning process will be represented in terms of formal guarantees. In section IV we define CBF-compliance and present the main theorem connecting the properties of the learned controller to robust safety. Finally in Section V we demonstrate these safety guarantees in simulation.

## II. PROBLEM SETUP

Throughout the rest of this work we consider nonlinear systems with a control-affine structure:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}, \quad (1)$$

where  $\mathbf{x} \in \mathbb{R}^n$  and  $\mathbf{u} \in \mathbb{R}^m$  are the states and inputs respectively and also where the drift dynamics  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$  and input matrix  $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$  are locally Lipschitz continuous on  $\mathbb{R}^n$ . Given a continuously differentiable state-feedback controller  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  that has access to full state information, the closed loop dynamics are given by:

$$\dot{\mathbf{x}} = \mathbf{f}_{cl}(\mathbf{x}) \triangleq \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{k}(\mathbf{x}). \quad (2)$$

The assumption of continuous differentiability of  $\mathbf{f}$ ,  $\mathbf{g}$ , and  $\mathbf{k}$  implies that  $\mathbf{f}_{cl}$  is locally Lipschitz continuous. Thus for any initial condition  $\mathbf{x}_0 \triangleq \mathbf{x}(0) \in \mathbb{R}^n$  there exists a time interval  $I(\mathbf{x}_0) = [0, t_{\max}]$  such that  $\mathbf{x}(t)$  is the unique solution to (2) on  $I(\mathbf{x}_0)$  [25]. In the remainder of this work we assume that the closed loop system (2) is *forward complete*, i.e.  $T_{\max} = \infty$ .

In the remainder of this section we provide preliminaries on imitation learning and control barrier functions.

### A. Imitation Learning

Imitation learning (IL) is a common learning framework in which a mapping between observations and actions is trained using expert demonstrations. Common methodologies in IL include behavioral cloning (a form of supervised learning), DAgger [2] which repeatedly collects additional state-action pairs, and inverse reinforcement learning (IRL) [26] which

learns a cost function such that the action or action sequence with minimal cost agrees with the expert demonstrations. We will present our method in the context of behavioral cloning due to its simplicity, but note that our method is not specific to this form of IL and can be generalized to provide safety guarantees for DAgger and IRL.

For end-to-end IL we model sensor measurements as:

$$\mathbf{y} = \mathbf{c}(\mathbf{x}) \quad (3)$$

where  $\mathbf{y} \in \mathbb{R}^k$  represents the system observations and  $\mathbf{c} : \mathbb{R}^n \rightarrow \mathbb{R}^k$  represents the system's sensors which we assume to be locally Lipschitz as in [11]. In the context of computer vision  $\mathbf{y}$  may be a vector representation of image data and  $\mathbf{c}$  may be the camera sensor which maps from state to image.

In order to train an end-to-end controller, we collect an exogenous dataset of observation-input pairs using the expert controller  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ :

$$\mathcal{D} = \{D_i\}_{i=1}^N, \quad D_i = (\mathbf{c}(\mathbf{x}_i), \mathbf{k}(\mathbf{x}_i)) \in \mathbb{R}^k \times \mathbb{R}^m \quad (4)$$

for  $N \in \mathcal{N}$  samples <sup>1</sup>. Given a nonlinear function class  $\mathcal{H} : \mathbb{R}^k \rightarrow \mathbb{R}^m$  and a loss function  $\mathcal{L} : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$ , the learning problem can be expressed as optimizing the parameters  $\theta$  of the function  $\mathbf{k}_\theta \in \mathcal{H}$  via empirical risk minimization:

$$\min_{\mathbf{k}_\theta \in \mathcal{H}} \frac{1}{N} \sum_{i=1}^N \mathcal{L}(\mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_i)), \mathbf{k}(\mathbf{x}_i)). \quad (5)$$

**Remark 1.** Behavioral cloning as in (5) suffers from compounding errors affecting the resulting trajectories [2]. However, since our goal is to transfer safety guarantees from the expert controller to the learned controller rather than to exactly mimic the expert behavior, we show that forward-invariance can be achieved despite compounding errors if the expert controller enforces robust forward-invariance as in Def. 6.

### B. Safety and Control Barrier Functions

In this section we define safety as the forward-invariance of a *safe set*  $\mathcal{C} \subset \mathbb{R}^n$  and discuss control barrier functions (CBFs) as a tool for achieving safety for nonlinear systems.

**Definition 1** (Forward Invariance and Safety). A set  $\mathcal{C} \subseteq \mathbb{R}^n$  is forward invariant if for every  $\mathbf{x}_0 \in \mathcal{C}$  the solution to (2) satisfies  $\mathbf{x}(t) \in \mathcal{C}$  for all  $t \geq 0$ . The system (2) is *safe* with respect to the set  $\mathcal{C}$  if  $\mathcal{C}$  is forward invariant.

Consider the set  $\mathcal{C}$  defined as the 0-superlevel set of some continuously differentiable function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$ :

$$\mathcal{C} \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\} \quad (6)$$

$$\text{Int}(\mathcal{C}) \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) > 0\} \quad (7)$$

$$\partial\mathcal{C} \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) = 0\} \quad (8)$$

These functions  $h$  are used in the following definition of CBFs:

**Definition 2** (Control Barrier Function (CBF), [8]). Let  $\mathcal{C} \subseteq \mathbb{R}^n$  be the 0-superlevel set of a continuously differentiable function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$  with 0 a regular value<sup>2</sup>. The function

<sup>1</sup>As in [5], we assume that the expert controller has access to the true state.

<sup>2</sup>0 is a regular value of  $h : \mathbb{R}^n \rightarrow \mathbb{R}$  if  $h(\mathbf{x}) = 0 \implies \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \neq 0$

$h$  is a *Control Barrier Function (CBF)* for (1) on  $\mathcal{C}$  if there exists a locally Lipschitz extended class  $\mathcal{K}$  infinity function<sup>3</sup>  $\alpha \in \mathcal{K}_\infty^e$  such that for all  $\mathbf{x} \in \mathcal{C}$ :

$$\sup_{\mathbf{u} \in \mathbb{R}^m} \underbrace{\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \mathbf{f}(\mathbf{x}) + \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \mathbf{g}(\mathbf{x})}_{L_f h(\mathbf{x})} \mathbf{u} \geq -\alpha(h(\mathbf{x})). \quad (9)$$

This safety condition (9) constrains the time derivative of  $h$ , preventing it from being negative if  $h(\mathbf{x}(0)) = 0$  and thus rendering the 0-superlevel set  $\mathcal{C}$  forward invariant.

This mathematical guarantee of safety is formalized as:

**Theorem 1** ([8]). Given a set  $\mathcal{C} \subseteq \mathbb{R}^n$  defined as the 0-superlevel set of a continuously differentiable function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$ , if  $h$  is a CBF for (1) on  $\mathcal{C}$ , then any locally Lipschitz continuous controller  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ , such that  $\mathbf{k}(\mathbf{x}) = \mathbf{u}$  satisfies (9), renders system (2) safe with respect to  $\mathcal{C}$ .

Beyond verifying the safety of existing closed-loop systems, CBFs can also be used as a tool for generating safe control inputs. One such controller that satisfies (9) is the CBF-QP:

$$\begin{aligned} \mathbf{k}_{\text{cbf-qp}} &= \underset{\mathbf{u} \in \mathbb{R}^m}{\operatorname{argmin}} \quad \frac{1}{2} \|\mathbf{u} - \mathbf{k}_{\text{nom}}(\hat{\mathbf{x}}_\theta)\|^2 \\ &\text{s.t. } L_f h(\hat{\mathbf{x}}_\theta) + L_g h(\hat{\mathbf{x}}_\theta) \geq -\alpha(h(\hat{\mathbf{x}}_\theta)). \end{aligned} \quad (\text{CBF-QP})$$

The controller  $\mathbf{k}_{\text{cbf-qp}} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  filters a nominal and potentially unsafe controller  $\mathbf{k}_{\text{nom}} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  and returns an input which satisfies the safety constraint (9).

### III. ROBUST SAFETY

In this section we discuss robust safety, which will allow us to capture the uncertainties generated by IL. In particular, we begin by giving a summary of input-to-state safety (ISSf) followed by a discussion of the continuity properties of level sets of  $h$  which, again, will allow us to reason about uncertainty as it is reflected in the safe set  $\mathcal{C}$  given by  $h$ .

#### A. Input-to-state Safety

We begin by discussing robust safety and the Input-to-State Safety (ISSf) property [12] which extends CBF-based set-invariance to systems with disturbances. To do this we consider the system (1) and introduce a matched, bounded, and potentially time-varying disturbance  $\mathbf{d} : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^m$ ,

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})(\mathbf{k}(\mathbf{x}) + \mathbf{d}(t)). \quad (10)$$

Due to the disturbance, the safety guarantees established in Theorem 1 no longer hold. To analyze the effect of the disturbance on safety we instead consider the expanded safe set  $\mathcal{C}_\delta \supseteq \mathcal{C}$  for some  $\delta \geq 0$ :

$$\mathcal{C}_\delta \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq -\gamma(\delta)\} \quad (11)$$

$$\text{Int}(\mathcal{C}_\delta) \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) > -\gamma(\delta)\} \quad (12)$$

$$\partial\mathcal{C}_\delta \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) = -\gamma(\delta)\}. \quad (13)$$

<sup>3</sup>A continuous function  $\alpha : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  is *class  $\mathcal{K}_\infty$*  ( $\alpha \in \mathcal{K}_\infty$ ) if  $\alpha(0) = 0$ ,  $\alpha$  is strictly monotonically increasing, and  $\lim_{c \rightarrow \infty} \alpha(c) = \infty$ . We say that a continuous function  $\alpha : \mathbb{R} \rightarrow \mathbb{R}$  is *extended class  $\mathcal{K}_\infty^e$*  ( $\alpha \in \mathcal{K}_\infty^e$ ) if  $\alpha(0) = 0$ ,  $\alpha$  is strictly monotonically increasing,  $\lim_{c \rightarrow \infty} \alpha(c) = \infty$  and  $\lim_{c \rightarrow -\infty} \alpha(c) = -\infty$ .

where  $\gamma \in \mathcal{K}_\infty$ . We note that the original safe set is recovered when  $\delta = 0$ , i.e.  $\mathcal{C}_0 = \mathcal{C}$ .

We can now define Input-to-State-Safety (ISSf) as the forward invariance of the set  $\mathcal{C}_\delta$  in the presence of disturbances.

**Definition 3** (Input-to-State Safety (ISSf) [27]). Let  $\mathcal{C} \subset \mathbb{R}^n$  be the 0-superlevel set of a continuously differentiable function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$ . The system (10) is *input-to-state safe* (ISSf) if there exists  $\gamma \in \mathcal{K}_\infty$  and  $\delta \geq 0$  such that  $\forall \mathbf{d}$  where  $\|\mathbf{d}\|_\infty \leq \delta$ , the set  $\mathcal{C}_\delta$  defined by (11) is forward invariant. In this case we refer to the set  $\mathcal{C}$  as an *input-to-state safe set* (ISSf set).

ISSf is the safety analog to the more common Input-to-State Stability (ISS) property of Lyapunov stable systems [28].

#### B. Continuity Properties

Before proving the safety-transfer that occurs between the expert and learned controllers in (5), we must first establish the upper semi-continuity (USC) of the level sets of  $h$  which will allow us to reason about the expanded safe set  $\mathcal{C}_\delta$ .

To discuss the non-zero level sets, we define the  $c$ -level set of a function  $h$  using the point-to-set map  $h^{-1} : \mathbb{R} \rightsquigarrow \mathcal{P}(\mathbb{R}^n)$ ,

$$h^{-1}(c) = \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) = c\} \quad (14)$$

where  $c \in \mathbb{R}$  and  $\mathcal{P}(\mathbb{R}^n)$  denotes the power set of  $\mathbb{R}^n$ .

One definition used to describe the continuity properties of point-to-set maps is u.s.c.:

**Definition 4** (Upper Semi-Continuity (u.s.c.) [29]). A set valued function map  $h^{-1} : \mathbb{R} \rightsquigarrow \mathcal{P}(\mathbb{R}^n)$  is *upper semi-continuous* at  $c \in \mathbb{R}$  if and only if for any  $\epsilon > 0$ , there exists  $\eta > 0$  such that  $c' \in B_\eta(c) \implies h^{-1}(c') \subset h^{-1}(c) \oplus B_\epsilon$ .

where  $B_\eta$  is the Euclidean ball of radius  $\eta$ .

It was established in [30, Proposition 6] that, under common assumptions for CBFs, the level sets  $h^{-1}$  are u.s.c.

**Proposition 1** ([30]). Let  $h^{-1} : \mathbb{R} \rightsquigarrow \mathcal{P}(\mathbb{R}^n)$  be the point-to-set map (14) representing the  $c$ -level set of some continuously differentiable function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$ . If 0 is a regular value of  $h$  and  $\Lambda := \{\mathbf{x} \in \mathbb{R}^n \mid -\delta \leq h(\mathbf{x}) \leq \delta\}$  is compact for all  $\delta \geq 0$ , then  $h^{-1}$  is upper semi-continuous at 0.

As noted in the statement of the proposition, this result follows from [30]. We include a proof here for completeness.

*Proof.* We first claim that  $h^{-1}$  is a continuous point-to-set map on some open set  $\mathbf{U}$  containing 0. Let  $C = \{c \in \mathbb{R} \mid h(\mathbf{x}) = c, \mathbf{x} \in \mathbb{R}^n\}$  and  $\mathcal{C} = \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\}$ . Define two point-to-set maps  $h^+, h^- : C \rightsquigarrow \mathbb{R}^n$  according to:

$$h^+(c) = \{\mathbf{x} \in \mathbb{R}^n \mid -h(\mathbf{x}) + c \leq 0\} \quad (15)$$

$$h^-(c) = \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) + c \leq 0\} \quad (16)$$

and observe that  $h^{-1}(c) = h^+(c) \cap h^-(c)$  for all  $c \in C$ . Because 0 is assumed to be a regular value of  $h$  and  $\partial\mathcal{C}$  is nonempty,  $h^+(c)$  and  $h^-(c)$  are nonempty for all  $c$  in some neighborhood of 0 of  $V$ . The functions  $-h(\mathbf{x}) + c$  and  $h(\mathbf{x}) - c$  are both continuous on  $C \times \mathbb{R}^n$ , so  $h^+$  and  $h^-$  are closed on  $V$  [31, Thm. 10], and so is  $h^{-1}$  due to [32, 6.1 Thm. 5]. Since  $h^{-1}(c)$  is assumed to map into the compact set  $\Lambda$  for

$c \in [-\delta, \delta]$ ,  $h^{-1}(c)$  is upper semi-continuous on  $V \cap [-\delta, \delta]$  [32, 6.1 Corollary to Thm. 7 ].  $\square$

This proposition relates the regularity of  $h$  to the upper semi-continuity of its level sets. In essence, the regularity of  $h$  at 0 ensures that small changes in the value defining the level set have small effects on the level sets themselves.

#### IV. MAIN RESULT

In this section we present our main result relating the supervised training of end-to-end controllers to input-to-state safety. To render the following closed loop system safe:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{k}_\theta(\mathbf{c}(\mathbf{x})). \quad (17)$$

To set up the behavioral cloning problem (5) we require an expert controller that provides robustness to matched disturbances and state uncertainty. For this purpose we choose the Tunable Robust Optimization Program TR-OP controller [33].

**Definition 5** (Tunable Robust Optimization Program (TR-OP) Controller ).

$$\begin{aligned} \mathbf{k}_T(\hat{\mathbf{x}}) &= \underset{\mathbf{v} \in \mathbb{R}^m}{\operatorname{argmin}} \|\mathbf{v} - \mathbf{k}_{\text{nom}}(\hat{\mathbf{x}})\|^2 && \text{(TR-OP)} \\ \text{s.t. } L_{\mathbf{f}} h(\hat{\mathbf{x}}) + L_{\mathbf{g}} h(\hat{\mathbf{x}}) \mathbf{v} \\ &\quad - \varphi \|L_{\mathbf{g}} h(\hat{\mathbf{x}})\|^2 - a - b \|\mathbf{v}\| \geq -\alpha(h(\hat{\mathbf{x}})). \end{aligned}$$

with parameters  $\varphi, a, b \in \mathbb{R}_{\geq 0}$  and  $\alpha \in \mathcal{K}_\infty^e$ .

$\varphi, a$ , and  $b$  effect the robustness of the closed loop system with respect to matched disturbances and state uncertainty.

**Definition 6** (CBF-Compliance). The learned controller  $\mathbf{k}_\theta : \mathbb{R}^k \rightarrow \mathbb{R}^m$  is a *CBF-Compliant* for some  $h : \mathbb{R}^n \rightarrow \mathbb{R}$  with measurement function  $\mathbf{c} : \mathbb{R}^n \rightarrow \mathbb{R}^k$  if

$$\min_{\mathbf{x} \in \mathcal{D}} \|\mathbf{x}_1 - \mathbf{x}\| \leq r_1, \quad (18)$$

$$\|\mathbf{k}_T(\mathbf{x}_2) - \mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_2))\| \leq M_e, \quad (19)$$

$$\|\mathbf{k}_\theta(\mathbf{x}_3) - \mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_4))\| \leq \mathfrak{L}_{\mathbf{k}_\theta \circ \mathbf{c}} \|\mathbf{x}_3 - \mathbf{x}_4\|, \quad (20)$$

for all  $\mathbf{x}_1 \in \partial \mathcal{C}$ ,  $(\mathbf{x}_2, \mathbf{k}_T(\mathbf{x}_2)) \in \mathcal{D}$ , and  $\mathbf{x}_3, \mathbf{x}_4 \in \partial \mathcal{C} \oplus \overline{B}_{r_2}$  where  $r_1, r_2 \in \mathbb{R}_{>0}$ ,  $\mathfrak{L}_{\mathbf{k}_\theta}, M_e \in \mathbb{R}_{\geq 0}$ , and  $\mathbf{k}_T : \mathbb{R}^n \rightarrow \mathbb{R}^m$  is a TR-OP controller for the function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$  with parameters  $\varphi, a, b \in \mathbb{R}_{\geq 0}$  and  $\alpha \in \mathcal{K}_\infty^e$ .

Note that  $\partial \mathcal{C} \oplus \overline{B}_{r_2}$  indicates the Minkowski sum of  $\partial \mathcal{C} \oplus \overline{B}_{r_2}$ .

**Remark 2.** Intuitively, any trajectory  $\mathbf{x}(t)$  that may leave an expanded safe set  $\mathcal{C}_\delta \supset \mathcal{C}$  must pass through the boundary of an expanded set  $\partial \mathcal{C}_\delta$ . Forward invariance of  $\mathcal{C}_\delta$  and upper semi-continuity of  $h$  ensure that the trajectory remains sufficiently close to a point in  $\partial \mathcal{C}$  for which we have expert data, thus preventing the cascading failure typical of behavioral cloning.

Next, we use the u.s.c. of  $h$  to relate the existence of a CBF-compliant controller to the ISSf of system (17).

**Theorem 2.** Let  $\mathcal{C} \subset \mathbb{R}^n$  be the 0-superlevel set of a some function  $h : \mathbb{R}^n \rightarrow \mathbb{R}$  which satisfies the Proposition 1.

There exist  $\underline{\varphi}, \underline{a}, \underline{b} \in \mathbb{R}_{\geq 0}$  such that if

- $\mathbf{k}_\theta : \mathbb{R}^k \rightarrow \mathbb{R}^m$  is a CBF-compliant controller on  $\mathcal{C}$  for parameters  $\varphi \geq \underline{\varphi}, a \geq \underline{a}, b \geq \underline{b}, \alpha \in \mathcal{K}_\infty^e$  with constants  $r_1, r_2 > 0$

- and  $L_{\mathbf{f}} h, L_{\mathbf{g}} h, \|L_{\mathbf{g}} h\|^2$ , and  $\alpha \circ h$  be Lipschitz continuous on  $\partial \mathcal{C} \oplus \overline{B}_{r_2}$ ,

then the closed loop system (17) is ISSf with respect to  $\mathcal{C}$  and safe with respect to

$$\mathcal{C}_\delta = \left\{ \mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq \alpha^{-1} \left( \frac{-1}{2\varphi} (\mathfrak{L}_{\mathbf{k}_\theta} r_3 + M_e)^2 \right) \right\}. \quad (21)$$

where  $r_3 \triangleq r_2 + r_1$ .

*Proof.* Consider some state  $\mathbf{x}_3 \in \partial \mathcal{C} \oplus B_{r_2}$ . Given  $\mathbf{x}_3$  there must exist some  $\mathbf{x}_2 \in \partial \mathcal{C}$  such that  $\|\mathbf{x}_2 - \mathbf{x}_3\| \leq r_2$ . Additionally, since  $\mathbf{k}_\theta$  is a CBF-compliant controller there must be some  $\mathbf{x}_1 \in \mathcal{D}$  such that  $\|\mathbf{x}_1 - \mathbf{x}_2\| \leq r_1$  and so  $\|\mathbf{x}_1 - \mathbf{x}_3\| \leq r_3$  by the triangle inequality.

The function  $h$  satisfies Proposition 1 by assumption so by the definition of upper semi-continuity it follows that:

$$\exists \eta > 0 \text{ s.t. } c \in B_\eta \implies h^{-1}(c) \subset h^{-1}(0) \oplus B_{r_2}. \quad (22)$$

Thus we can choose  $\underline{\varphi} > 0$  large enough such that for any  $\varphi \geq \underline{\varphi}$  we have that  $\partial \mathcal{C}_\delta \subset \partial \mathcal{C} \oplus B_{r_2}$  for  $\mathcal{C}_\delta$  as in (21). Next we choose the remaining parameter bounds to be:

$$a \geq \underline{a} = r_3 (\mathfrak{L}_{L_{\mathbf{f}} h} + \mathfrak{L}_{\alpha \circ h} + \mathfrak{L}_{\varphi \|L_{\mathbf{g}} h\|^2}), \quad (23)$$

$$b \geq \underline{b} = r_3 \mathfrak{L}_{L_{\mathbf{g}} h}. \quad (24)$$

where  $\mathfrak{L}$  represents the Lipschitz constant of the subscripted function on  $\partial \mathcal{C} \oplus B_{r_2}$ .

Using  $\mathbf{k}_\theta$  we can bound the time derivative of the CBF at  $\mathbf{x}_3 \in \mathcal{C} \oplus B_{r_2}$  as:

$$\frac{d}{dt} h(\mathbf{x}_3, \mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_3))) \quad (25)$$

$$= L_{\mathbf{g}} h(\mathbf{x}_1) \mathbf{k}_T(\mathbf{x}_1) + \frac{d}{dt} h(\mathbf{x}_3, \mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_3))) - L_{\mathbf{g}} h(\mathbf{x}_1) \mathbf{k}_T(\mathbf{x}_1) \quad (26)$$

$$\geq L_{\mathbf{f}} h(\mathbf{x}_3) - L_{\mathbf{f}} h(\mathbf{x}_1) + r_3 \mathfrak{L}_{L_{\mathbf{f}} h} \\ + \alpha(h(\mathbf{x}_3)) - \alpha(h(\mathbf{x}_1)) + r_3 \mathfrak{L}_{\alpha \circ h} \\ + \varphi \|L_{\mathbf{g}} h(\mathbf{x}_1)\|^2 - \varphi \|L_{\mathbf{g}} h(\mathbf{x}_3)\|^2 + r_3 \mathfrak{L}_{\varphi \|L_{\mathbf{g}} h\|^2} \\ + L_{\mathbf{g}} h(\mathbf{x}_3) \mathbf{k}_T(\mathbf{x}_1) - L_{\mathbf{g}} h(\mathbf{x}_1) \mathbf{k}_T(\mathbf{x}_1) + r_3 \mathfrak{L}_{L_{\mathbf{g}} h} \|\mathbf{k}_T(\mathbf{x}_1)\| \\ - \alpha(h(\mathbf{x}_3)) + \varphi \|L_{\mathbf{g}} h(\mathbf{x}_3)\|^2 \\ + L_{\mathbf{g}} h(\mathbf{x}_3) (\mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_3)) - \mathbf{k}_T(\mathbf{x}_1))$$

We can now lower bound the first four lines of (26) using Lipschitz constants. For example,

$$L_{\mathbf{f}} h(\mathbf{x}_3) - L_{\mathbf{f}} h(\mathbf{x}_1) + r_3 \mathfrak{L}_{L_{\mathbf{f}} h} \quad (27)$$

$$\geq -\|L_{\mathbf{f}} h(\mathbf{x}_3) - L_{\mathbf{f}} h(\mathbf{x}_1)\| + r_3 \mathfrak{L}_{L_{\mathbf{f}} h} \quad (28)$$

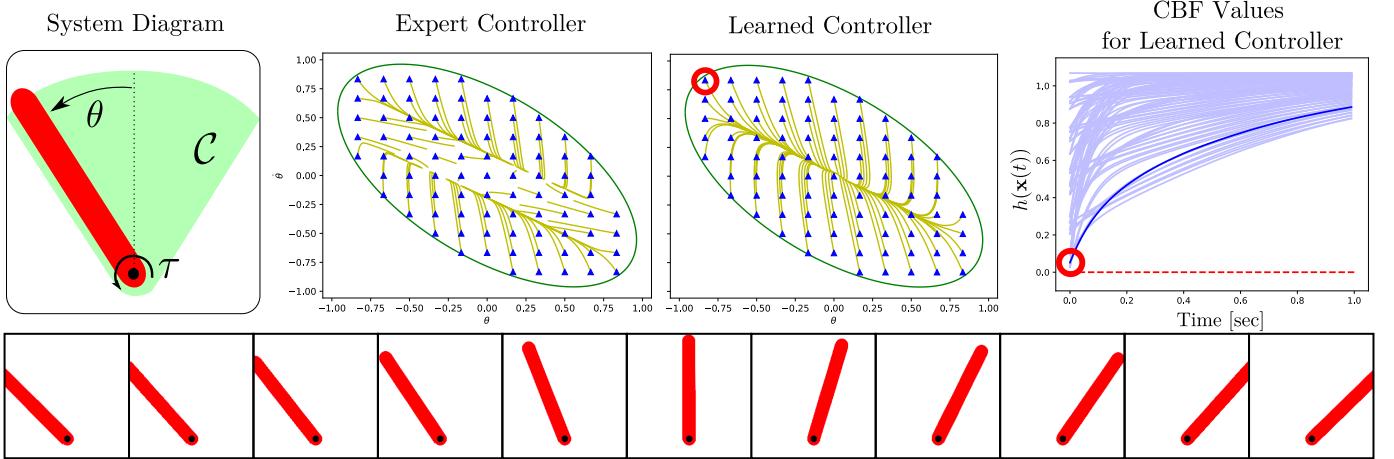
$$\geq \mathfrak{L}_{L_{\mathbf{f}} h} (r_3 - \|\mathbf{x}_1 - \mathbf{x}_2 + \mathbf{x}_2 - \mathbf{x}_3\|) \quad (29)$$

$$\geq \mathfrak{L}_{L_{\mathbf{f}} h} (r_3 - \|\mathbf{x}_1 - \mathbf{x}_2\| - \|\mathbf{x}_2 - \mathbf{x}_3\|) \geq 0 \quad (30)$$

since  $\|\mathbf{x}_1 - \mathbf{x}_2\| \leq r_1$  and  $\|\mathbf{x}_2 - \mathbf{x}_3\| \leq r_2$ .

Applying these Lipschitz-based bounds for  $L_{\mathbf{f}} h, L_{\mathbf{g}} h, \alpha \circ h$ , and  $\varphi \|L_{\mathbf{g}} h\|^2$  yields:

$$\begin{aligned} \frac{d}{dt} h(\mathbf{x}_3, \mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_3))) &\geq -\alpha(h(\mathbf{x}_3)) + \varphi \|L_{\mathbf{g}} h(\mathbf{x}_3)\|^2 \\ &\quad + L_{\mathbf{g}} h(\mathbf{x}_3) (\mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_3)) - \mathbf{k}_T(\mathbf{x}_1)) \end{aligned} \quad (31)$$



**Fig. 2.** Results for the inverted pendulum. **Left:** A diagram of the system where the green represents the safe set  $\mathcal{C}$ . **Center Left:** One second long trajectories generated by the expert controller  $\mathbf{k}_T$  are shown in yellow and plotted for several initial conditions represented as blue triangles. in The green ellipse marks the boundary  $\partial\mathcal{C}$ . **Center Right:** One second long trajectories generated by the learned controller  $\mathbf{k}_\theta$  are shown in yellow and plotted for several initial conditions represented as blue triangles. in The green ellipse marks the boundary  $\partial\mathcal{C}$ . **Right:** CBF values  $h(\mathbf{x}(t))$  achieved by the learned controller. Note all are greater than zero indicating safety of the system. The darker blue trajectory begins at the initial condition marked by the red circle in the plot of the learned controller. **Bottom:** Images spanning the safe set  $\mathcal{C}$  that are used by  $\mathbf{k}_\theta$  for end-to-end control of the system.

Additionally we can lower bound the final term using properties (19) and (20) of CBF-compliant controller as:

$$\begin{aligned} & L_{\mathbf{g}} h(\mathbf{x}_3)(\mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_3)) - \mathbf{k}_T(\mathbf{x}_1)) \\ & \geq L_{\mathbf{g}} h(\mathbf{x}_3)(\mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_3)) - \mathbf{k}_\theta(\mathbf{x}_1) + \mathbf{k}_\theta(\mathbf{x}_1) - \mathbf{k}_T(\mathbf{x}_1)) \end{aligned} \quad (32)$$

$$\begin{aligned} & \geq -\|L_{\mathbf{g}} h(\mathbf{x}_3)\| \left( \|\mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_3)) - \mathbf{k}_\theta(\mathbf{x}_1)\| \right. \\ & \quad \left. + \|\mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_1)) - \mathbf{k}_T(\mathbf{x}_1)\| \right) \end{aligned} \quad (33)$$

$$\geq -\|L_{\mathbf{g}} h(\mathbf{x}_3)\| (\mathfrak{L}_{\mathbf{k}_\theta \circ \mathbf{c}} r_3 + M_e). \quad (34)$$

Using (34) to lower-bound (31) results in:

$$\begin{aligned} & \frac{d}{dt} h(\mathbf{x}_3, \mathbf{k}_\theta(\mathbf{c}(\mathbf{x}_3))) \\ & \geq -\alpha(h(\mathbf{x}_3)) + \varphi \|L_{\mathbf{g}} h(\mathbf{x}_3)\|^2 \\ & \quad - \|L_{\mathbf{g}} h(\mathbf{x}_3)\| (\mathfrak{L}_{\mathbf{k}_\theta \circ \mathbf{c}} r_3 + M_e), \end{aligned} \quad (35)$$

$$\geq -\alpha(h(\mathbf{x}_3)) - \frac{1}{2\varphi} (\mathfrak{L}_{\mathbf{k}_\theta \circ \mathbf{c}} r_3 + M_e)^2, \quad (37)$$

where the final bound is achieved by completing the square and removing the positive term.

To achieve forward invariance of  $\mathcal{C}_\delta$  we note that  $h(\mathbf{x}_3) = \alpha^{-1} \left( -\frac{1}{2\varphi} (\mathfrak{L}_{\mathbf{k}_\theta} r_3 + M_e) \right) \implies \frac{d}{dt} h(\mathbf{x}_3, \mathbf{k}_\theta(\mathbf{x}_3)) \geq 0$ . Since the bound (37) holds for all  $\mathbf{x}_3 \in \partial\mathcal{C} \oplus B_{r_2}$  and  $\varphi$  was chosen such that  $\partial\mathcal{C}_\delta \subset \partial\mathcal{C} \oplus B_{r_2}$  it is true that  $\frac{d}{dt} h(\mathbf{x}_4, \mathbf{k}_\theta(\mathbf{x}_4)) = 0$  for all  $\mathbf{x}_4 \in \partial\mathcal{C}_\delta$ . Thus by Nagumo's theorem [34] the set  $\mathcal{C}_\delta$  is forward invariant and  $\mathcal{C}$  is ISSf.  $\square$

To the best of our knowledge, this is the first result to establish a direct relationship between the safety of a system and the parameters of the imitation learning problem used to develop its controller. We recognize that finding and using the exact values of the Lipschitz constants may be impractical, but note that due to their conservatism the CBF-compliant controller may be capable of achieving safety with far smaller values as we demonstrate in simulation in Section V.

The learned controller  $\mathbf{k}_\theta$  developed in Theorem 2 has mathematical guarantees of safety, but may result in behaviors

significantly different than the expert controller in the interior of the safe set  $\text{Int}(\mathcal{C})$  when the system is far from the boundary  $\partial\mathcal{C}$ . Therefore we present a corollary which may result in improved behavioral cloning on the interior of  $\mathcal{C}$  due to increased sampling. The safety guarantees of this corollary follow immediately from Theorem 2.

**Corollary 1.** Let the dataset  $\mathcal{D}$  satisfy the inequality

$$\min_{\mathbf{x} \in \mathcal{D}} \|\mathbf{x}_1 - \mathbf{x}\| \leq r_1, \quad \forall \mathbf{x}_1 \in \mathcal{C} \quad (38)$$

in place of (18) for some  $r_1 > 0$ . Let the remaining assumptions of Theorem 2 hold, then the closed loop system (17) is ISSf with respect to  $\mathcal{C}$  and safe with respect to  $\mathcal{C}_\delta$  (11).

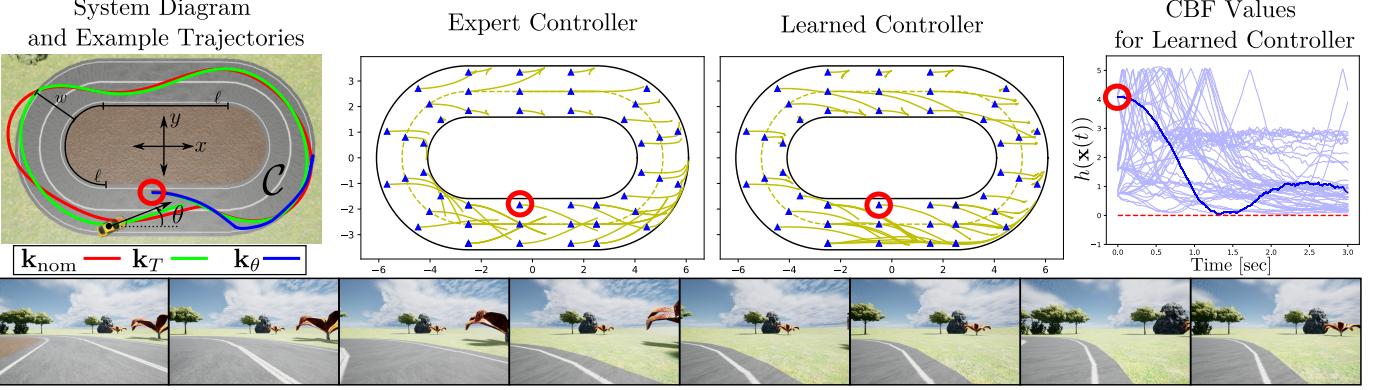
*Proof.*  $\mathcal{C}$  is a closed set so  $\partial\mathcal{C} \subseteq \mathcal{C}$ , thus (38)  $\implies$  (18) and the conditions of Theorem 2 are met.  $\square$

## V. SIMULATION EXPERIMENTS

In this section we discuss the simulation results for safe vision-based end-to-end control of an inverted pendulum and a simplified car using CBF-compliant controller. In both cases the convolutional neural network used for end-to-end learning was MobileNetV2 [35] with an additional full-connected layer added to output control inputs of the proper dimension from the network. The full network has approximately 3.4 million parameters. Training was performed using the ADAM optimizer, an  $\ell_2$  loss with  $\ell_2$  weight decay, and batched training. The frequency of the observations was chosen to be 100 Hz and 60 Hz for the pendulum and car, respectively. The simulations were conducted using zero-order-hold control inputs of the same frequency with no latency. Additional information such as values of constants and image formatting can be found in our code [36].

### A. Inverted Pendulum

We first consider an inverted pendulum system with the states  $\mathbf{x} = [\theta \ \dot{\theta}]^\top$  with torque inputs  $\tau \in \mathbb{R}$  as shown in



**Fig. 3.** Results for the car. **Left:** three 15 second long trajectories are shown starting from the same initial condition, the nominal controller generates the unsafe trajectory in red, the TR-OP controller generates the safe green trajectory, and the learned controller generates the safe blue trajectory. **Center Left:** 3 second long trajectories starting at several initial conditions are shown for the expert controller  $k_T$ . The red triangles represent initial conditions and are all within the safe set. The yellow lines represents the trajectories of the car beginning at a red triangle. **Center Right:** 3 second long trajectories for starting at several initial conditions shown for the learned controller  $k_\theta$ . **Right:** CBF values  $\min\{h_1(\mathbf{x}(t)), h_2(\mathbf{x}(t))\}$  achieved by the learned controller. Note all are greater than zero. The darker blue trajectory begins at the initial condition shown by the red circles in each plot. **Bottom:** Images used by  $k_\theta$  for end-to-end control of the system. From left to right, each the first-person view of the system starting at time  $t = 0$  seconds and increasing by 0.25 seconds.

Fig. 2. The dynamics and observation function of this system are given as:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\theta} \\ \sin \theta \end{bmatrix} + \begin{bmatrix} 0 \\ \tau \end{bmatrix}, \quad \mathbf{y} = \mathbf{c}(\mathbf{x}) = \begin{bmatrix} \text{Img}(\mathbf{x}) \\ \dot{\theta} \end{bmatrix} \quad (39)$$

where  $\text{Img}(\mathbf{x})$  represents the image of the system at state  $\mathbf{x}$  as seen from a camera facing the inverted pendulum. A series of example images can be found in Figure 2. The learned controller is a function of the current image and velocity of the system, so additional fully connected layer was added to incorporate the velocity data into the end-to-end controller.

The safe set for the inverted pendulum is chosen to be:

$$h(\mathbf{x}) = c - \mathbf{x}^\top \mathbf{P} \mathbf{x}, \quad \mathcal{C} = \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\} \quad (40)$$

where  $\mathbf{P} \in \mathbb{R}^{2 \times 2}$  is such that  $\mathbf{x}^\top \mathbf{P} \mathbf{x}$  is a control Lyapunov function derived from the continuous time algebraic Riccati equation using feedback linearization and  $c$  is chosen such that  $\max_{\theta \in \mathcal{C}} |\theta| = \pi/4$ . This safe set is visualized in Fig. 2.

The expert controller is the TR-OP controller with parameters  $\varphi = 2$ ,  $\alpha(c) = c$ , and  $a$  and  $b$  chosen as the Lipschitz constants of  $(L_f h(\mathbf{x}) + \alpha(h(\mathbf{x})) + \varphi \|L_g h(\mathbf{x})\|^2)$  and  $L_g h(\mathbf{x})$  respectively over the compact set  $\partial \mathcal{C}$  multiplied by the minimum sampling distance  $r_1 = 0.01$ . The nominal controller is  $k_{\text{nom}}(\mathbf{x}) = -0.75\theta$  which provides some torque to counteract gravity, but fails to stabilize the pendulum. The boundary of the safe set,  $\partial \mathcal{C}$ , is gridded and sampled uniformly with a minimum distance  $r_1$  to create the training dataset  $\mathcal{D}$ .

## B. Race Car

Next we consider a simplified car given by the unicycle dynamics and observation function:

$$\dot{\mathbf{x}} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \quad \mathbf{y} = \mathbf{c}(\mathbf{x}) = \text{Img}(\mathbf{x}) \quad (41)$$

where the state  $\mathbf{x} = [x \ y \ \theta]^\top$  is the planar position and heading angle and the input  $\mathbf{u} = [v \ \omega]^\top$  is the forward

and angular velocities and  $\text{Img}(\mathbf{x})$  represents the driver's first-person-view from the car at position  $\mathbf{x}$ . A series of example first-person-view images can be seen in Figure 3.

The safe set for the car is chosen to be the 0-superlevel set of the function  $\min\{h_1, h_2\}$  where:

$$h_i(\mathbf{x}) = \delta \hat{\mathbf{n}}^\top \hat{\mathbf{d}}_i + \psi_i \cdot \begin{cases} \rho_i^2 - \left( (x - \frac{\ell}{2})^2 + y^2 \right), & x \geq \frac{\ell}{2} \\ \rho_i^2 - \left( (x + \frac{\ell}{2})^2 + y^2 \right), & x \leq -\frac{\ell}{2} \\ \rho_i^2 - y^2, & \text{else} \end{cases}$$

where  $\rho_1 = (\ell/\pi + w)$ ,  $\psi_1 = 1$ ,  $\rho_2 = \ell/\pi$ , and  $\psi_2 = -1$ . Additionally,  $\delta = 0.1$ ,  $\hat{\mathbf{n}}$  is the unit vector in the car's heading direction,  $\hat{\mathbf{d}}_1$  is the unit vector pointing perpendicularly inward from the outer boundary of the track through the car's position,  $\hat{\mathbf{d}}_2$  is the unit vector point perpendicularly outward from the inner boundary of the track through the car's position,  $\ell$  is the length of the straight portions of the track,  $w$  is the width of the track. An annotated diagram of the track can be found in Figure 3. Given these functions, the safe set  $\mathcal{C} = \{\mathbf{x} \in \mathbb{R}^3 \mid \min\{h_1(\mathbf{x}), h_2(\mathbf{x})\} \geq 0\}$  is a subset of the track with an angle dependence where positions with heading angles pointed towards the center line are considered safer.

The expert controller is the TR-OP controller with the constraint simultaneously enforced for both  $h_1$  and  $h_2$  with parameters  $\varphi = 0.5$ ,  $a = 10^{-2}$ ,  $b = 10^{-4}$ , and  $\alpha(c) = 10c$ .  $\partial \mathcal{C}$  was gridded and sampled uniformly with distance of  $r_1 = 0.1$  to generate  $\mathcal{D}$ . We use Theorem 2 to guide the choice of these constants, but due to the difficulty of estimating the Lipschitz constants and the likely over-conservatism of the resulting controller we choose parameters which are likely much smaller than those required to sufficiently guarantee safety mathematically but we nonetheless succeed in demonstrating safety experimentally.

The nominal controller used in the TR-OP controller is:

$$k_{\text{nom}} = \begin{bmatrix} K_p |r - r_{\text{midline}}| + F \\ K_r (r - r_{\text{midline}}) + K_{\text{dir}} (\hat{\mathbf{n}}^\top \hat{\mathbf{e}}_{\text{midline}}) \end{bmatrix} \quad (42)$$

where  $K_p, F, K_r, K_{\text{dir}} \in \mathbb{R}_{>0}$ ,  $r = \|\mathbf{x}\|$ ,  $r_{\text{midline}}$  from the origin to the midline of the track passing through the

car,  $\hat{\mathbf{e}}_{\text{midline}}$  is the unit vector from the vehicle to the closest point on the midline. This nominal controller is capable of circumnavigating the track, but is unsafe.

### C. Learning and Results

For both the inverted pendulum and the car, the learned controller  $\mathbf{k}_\theta$  is trained until convergence to minimize (5) where the (TR-OP) controller is the expert controller.

The safe set for both systems was gridded with initial conditions and simulated forward for 1 second for the inverted pendulum and 3 seconds for the car. For the car,  $\theta = 0$  was held constant for each initial condition and the planar interior of the track was sampled. The resulting trajectories can be seen in Figures 2 and 3. For each trajectory the TR-OP controller renders the system safe and this safety is transferred to the learned controller despite having different behavior in the interior of the safe set. The smallest value for  $h$  achieved for in inverted pendulum example was 0.028 and the smallest value of  $\min\{h_1(\mathbf{x}), h_2(\mathbf{x})\}$  achieved during simulation for all initial conditions was 0.030, indicating safety of both systems.

Although the system deviates significantly from the expert trajectories, the learned controller successfully keeps the system inside of the safe set. Although the behaviors differ, we have shown that sampling on the boundary of the safe set is sufficient to render the safe set forward invariant. Additionally, as long as sufficient sampling of the boundary and learning accuracy is maintained for safety, the interior of the safe set can be sampled and alternative IL methods such as DAgger or IRL can be used to improve behavioral mimicry.

### REFERENCES

- [1] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, “Imitation learning: A survey of learning methods,” *ACM Computing Surveys (CSUR)*, vol. 50, no. 2, pp. 1–35, 2017.
- [2] S. Ross, G. Gordon, and D. Bagnell, “A reduction of imitation learning and structured prediction to no-regret online learning,” in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.
- [3] S. Schaal, “Is imitation learning the route to humanoid robots?” *Trends in cognitive sciences*, vol. 3, no. 6, pp. 233–242, 1999.
- [4] F. Codevilla, M. Müller, A. López, V. Koltun, and A. Dosovitskiy, “End-to-end driving via conditional imitation learning,” in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 4693–4700.
- [5] Y. Pan, C.-A. Cheng, K. Saigol, K. Lee, X. Yan, E. Theodorou, and B. Boots, “Agile autonomous driving using end-to-end deep imitation learning,” *arXiv preprint arXiv:1709.07174*, 2017.
- [6] K. P. Wabersich and M. N. Zeilinger, “Linear Model Predictive Safety Certification for Learning-Based Control,” in *2018 IEEE Conference on Decision and Control (CDC)*, Dec. 2018, pp. 7130–7135, iSSN: 2576-2370.
- [7] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin, “Hamilton-jacobi reachability: A brief overview and recent advances,” in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 2242–2253.
- [8] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, “Control Barrier Function Based Quadratic Programs for Safety Critical Systems,” *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, Aug. 2017, arXiv: 1609.06408.
- [9] U. Borrmann, L. Wang, A. D. Ames, and M. Egerstedt, “Control barrier certificates for safe swarm behavior,” *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 68–73, 2015.
- [10] Q. Nguyen and K. Sreenath, “Exponential control barrier functions for enforcing high relative-degree safety-critical constraints,” in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 322–328.
- [11] S. Dean, A. J. Taylor, R. K. Cosner, B. Recht, and A. D. Ames, “Guaranteeing Safety of Learned Perception Modules via Measurement-Robust Control Barrier Functions,” *arXiv:2010.16001 [cs, eess, math, stat]*, Oct. 2020, arXiv: 2010.16001.
- [12] S. Kolathaya and A. D. Ames, “Input-to-state safety with control barrier functions,” *IEEE control systems letters*, vol. 3, no. 1, pp. 108–113, 2018.
- [13] T. G. Molnar, R. K. Cosner, A. W. Singletary, W. Ubellacker, and A. D. Ames, “Model-Free Safety-Critical Control for Robotic Systems,” *arXiv:2109.09047 [cs, eess, math]*, Sept. 2021, arXiv: 2109.09047. [Online]. Available: <http://arxiv.org/abs/2109.09047>
- [14] U. Rosolia, A. Carvalho, and F. Borrelli, “Autonomous racing using learning model predictive control,” in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 5115–5120.
- [15] J. F. Fisac, N. F. Lugovoy, V. Rubies-Royo, S. Ghosh, and C. J. Tomlin, “Bridging hamilton-jacobi safety analysis and reinforcement learning,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8550–8556.
- [16] Y. Emam, P. Glotfelter, Z. Kira, and M. Egerstedt, “Safe model-based reinforcement learning using robust control barrier functions,” *arXiv preprint arXiv:2110.05415*, 2021.
- [17] F. Castañeda, J. J. Choi, B. Zhang, C. J. Tomlin, and K. Sreenath, “Pointwise feasibility of gaussian process-based safety-critical control under model uncertainty,” *arXiv preprint arXiv:2106.07108*, 2021.
- [18] A. J. Taylor, A. Singletary, Y. Yue, and A. D. Ames, “Learning for Safety-Critical Control with Control Barrier Functions,” p. 10.
- [19] N. Csomay-Shanklin, R. K. Cosner, M. Dai, A. J. Taylor, and A. D. Ames, “Episodic learning for safe bipedal locomotion with control barrier functions and projection-to-state safety,” in *Learning for Dynamics and Control*. PMLR, 2021, pp. 1041–1053.
- [20] D. S. Brown, Y. Cui, and S. Niekum, “Risk-aware active inverse reinforcement learning,” in *Conference on Robot Learning*. PMLR, 2018, pp. 362–372.
- [21] J. Zhang and K. Cho, “Query-efficient imitation learning for end-to-end simulated driving,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.
- [22] H. Yin, P. Seiler, M. Jin, and M. Arcak, “Imitation learning with stability and safety guarantees,” *IEEE Control Systems Letters*, vol. 6, pp. 409–414, 2021.
- [23] M. Hertneck, J. Köhler, S. Trimpe, and F. Allgöwer, “Learning an approximate model predictive controller with guarantees,” *IEEE Control Systems Letters*, vol. 2, no. 3, pp. 543–548, 2018.
- [24] S. Yaghoubi, G. Fainekos, and S. Sankaranarayanan, “Training neural network controllers using control barrier functions in the presence of disturbances,” in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2020, pp. 1–6.
- [25] L. Perko, *Differential equations and dynamical systems*. Springer Science & Business Media, 2013, vol. 7.
- [26] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey, et al., “Maximum entropy inverse reinforcement learning,” in *Aaaai*, vol. 8. Chicago, IL, USA, 2008, pp. 1433–1438.
- [27] A. Alan, A. J. Taylor, C. R. He, G. Orosz, and A. D. Ames, “Safe Controller Synthesis With Tunable Input-to-State Safe Control Barrier Functions,” *IEEE Control Systems Letters*, vol. 6, pp. 908–913, 2022, conference Name: IEEE Control Systems Letters.
- [28] E. D. Sontag, “Input to state stability: Basic concepts and results,” in *Nonlinear and optimal control theory*. Springer, 2008, pp. 163–220.
- [29] J.-P. Aubin and H. Frankowska, *Set-valued analysis*. Springer Science & Business Media, 2009.
- [30] R. Konda, A. D. Ames, and S. Coogan, “Characterizing safety: Minimal barrier functions from scalar comparison systems,” *arXiv preprint arXiv:1908.09323*, 2019.
- [31] W. W. Hogan, “Point-to-set maps in mathematical programming,” *SIAM review*, vol. 15, no. 3, pp. 591–603, 1973.
- [32] C. Berge, *Topological Spaces: including a treatment of multi-valued functions, vector spaces, and convexity*. Courier Corporation, 1997.
- [33] R. K. Cosner, M. Tucker, A. J. Taylor, K. Li, T. G. Molnár, W. Ubellacker, A. Alan, G. Orosz, Y. Yue, and A. D. Ames, “Safety-Aware Preference-Based Learning for Safety-Critical Control,” *arXiv:2112.08516 [cs, eess]*, Dec. 2021, arXiv: 2112.08516.
- [34] M. Nagumo, “Über die lage der integralkurven gewöhnlicher differentialgleichungen,” *Proceedings of the Physico-Mathematical Society of Japan. 3rd Series*, vol. 24, pp. 551–559, 1942.
- [35] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.
- [36] Supplementary code, <https://github.com/rkcosner/E2E-IL-CBF.git>.