Proceedings of Machine Learning Research vol 120:1–10, 2020 2nd Annual Conference on Learning for Dynamics and Control

Learning for Safety-Critical Control with Control Barrier Functions

Andrew J. Taylor AJTAYLOR@CALTECH.EDU
Caltech, Pasadena, CA, USA

Andrew Singletary ASINGLET@CALTECH.EDU
Caltech, Pasadena, CA, USA

Yisong Yue YYUE@CALTECH.EDU
Caltech, Pasadena, CA, USA

Aaron D. Ames AMES@CALTECH.EDU
Caltech, Pasadena, CA, USA

Editors: A. Bayen, A. Jadbabaie, G. J. Pappas, P. Parrilo, B. Recht, C. Tomlin, M.Zeilinger

Abstract

Modern nonlinear control theory seeks to endow systems with properties of stability and safety, and have been deployed successfully in multiple domains. Despite this success, model uncertainty remains a significant challenge in synthesizing safe controllers, leading to degradation in the properties provided by the controllers. This paper develops a machine learning framework utilizing Control Barrier Functions (CBFs) to reduce model uncertainty as it impact the safe behavior of a system. This approach iteratively collects data and updates a controller, ultimately achieving safe behavior. We validate this method in simulation and experimentally on a Segway platform.

Keywords: feedback control, barrier functions, supervised learning, safety, robotics

1. Introduction

Safety is of significant importance in many modern control applications, and as the complexity of systems increases, it is necessary to rigorously encode safety properties during the controller design process. Autonomous driving, industrial robotics, and aerospace vehicles are examples of rapidly growing applications where safe control synthesis is critical. In practice, the models used to design such controllers are imperfect, with model uncertainty arising due to parametric error and unmodeled dynamics. This uncertainty can lead to unsafe or dangerous behavior, and thus it is critical to quantify safety in the face of uncertainty.

In this work we consider a data-driven, machine learning approach to reduce model uncertainty and improve the safe behavior of a system. Learning-based approaches have already shown great promise for controlling systems with uncertain models (Schaal and Atkeson (2010); Kober et al. (2013); Khansari-Zadeh and Billard (2014); Cheng et al. (2019); Taylor et al. (2019b); Shi et al. (2019)). We look to achieve safety defined in terms of *set invariance* (Blanchini (1999); Ames et al. (2019)), which is an area of active research at the intersection of machine learning and control theory (Berkenkamp et al. (2016); Fisac et al. (2018)). Additional details on related work are provided in the extended version of this paper (Taylor et al. (2019c)).

We will leverage Control Barrier Functions (CBFs) for synthesizing safe controllers (Ames et al. (2014, 2017)) that achieve set invariance. CBFs have become increasingly popular for enforcing safety (Nguyen and Sreenath (2016); Wang et al. (2018)), but their safety guarantees require an

© 2020 A.J. Taylor, A. Singletary, Y. Yue & A.D. Ames.

accurate model of the system, and thus model uncertainty can lead to degradation of these guarantees (Kolathaya and Ames (2018)). Robust CBF formulations can ensure safety in the face of model uncertainty (Gurriet et al. (2018); Xu et al. (2015)), but may be overly restrictive of the system's behavior. Adaptive CBF methods can provide safety with arbitrarily large parametric error (Taylor and Ames (2019)), but only over a restricted class of model uncertainty.

One challenge in utilizing learning for improving controllers is collecting training data that accurately reflects the desired learning environment. Exhaustive data collection suffers from the *curse of dimensionality* and scales poorly with the size of the system's state and input. Additionally, the pre-collection of data upfront can lead to bad performance as the learning-augmented controller may enter new states not present in the pre-collected training data. We will utilize episodic learning approaches such as Dataset Aggregation (DAgger) (Ross et al. (2010)) to address these challenges efficiently, and lead to an iteratively improved controller. Furthermore, we build upon recent work utilizing learning in the context of Control Lyapunov Functions (CLFs) (Taylor et al. (2019b)) to construct an approach for learning model uncertainty. Unlike this preceding work, our method learns model uncertainty impacting safety rather than stability, and provides experimental demonstration.

Our contribution: The contribution of this work is a novel approach for learning model uncertainty directly as it impacts the evolution of a Control Barrier Function and synthesizing a new controller for enforcing safety. Previous work has focused on learning an invariant set under a given controller, or learning unmodeled dynamics in the full-order state dynamics and modifying a controller. This approach learns the effect of unmodeled dynamics and parametric error on the system's behavior episodically, and integrates with optimization based methods for enforcing safety from nonlinear control theory. We demonstrate the functionality of this learning approach for a Segway system both in simulation and, for the first time, experimentally on a hardware platform.

Our paper is organized as follows. Section 2 provides a review of Control Barrier Functions and safety-critical control. In Section 3 we state assumptions on our model uncertainty, and define our learning approach for capturing the impact of this uncertainty on the CBF dynamics. Next, Section 4 details issues with standard supervised learning in a sequential prediction task, and provides an episodic framework that utilizes our particular learning approach. Lastly, Section 5 provides both simulation and experimental results on a Segway platform.

2. Control Barrier Functions

In this section we provide a review of Control Barrier Functions (CBFs) and how they can be utilized to ensure the safety of a system. This will motivate a particular episodic learning approach that achieves safety in the presence of uncertainty in modeled system dynamics.

Consider a dynamic system with state $\mathbf{x} \in \mathbb{R}^n$ and control input $\mathbf{u} \in \mathbb{R}^m$. We assume that the dynamics of the system are continuous in time and affine in the control input, such that:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u} \qquad \xrightarrow{\mathbf{u}=\mathbf{k}(\mathbf{x})} \qquad \dot{\mathbf{x}} = \mathbf{f}_{\text{cl}}(\mathbf{x}) = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{k}(\mathbf{x}) \tag{1}$$

where $\mathbf{k} : \mathbb{R}^n \to \mathbb{R}^m$ is a locally Lipschitz continuous controller that yields closed-loop dynamics. Here $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$, $\mathbf{g} : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are assumed to be locally Lipschitz continuous and, therefore, $\mathbf{f}_{\text{cl}} : \mathbb{R}^n \to \mathbb{R}^n$ is locally Lipschitz continuous. Under this assumption, for any initial state $\mathbf{x}_0 \in \mathbb{R}^n$, there exists a time interval of existence, $I(\mathbf{x}_0) = [0, \tau_{max})$, such that there is a unique solution, $\mathbf{x} : I(\mathbf{x}_0) \to \mathbb{R}^n$, satisfying (1) with $\mathbf{x}(0) = \mathbf{x}_0$ (Perko (2013)).

The notion of safety is formalized by specifying a *safe set* that the state of the system must remain in to be considered safe. In particular, consider a set $\mathcal{S} \subset \mathbb{R}^n$ defined as the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$, with 0 a regular value ($h(\mathbf{x}) = 0 \implies \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \neq \mathbf{0}$), yielding:

$$\mathcal{S} \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\}. \tag{2}$$

We refer to $\mathcal{S}$ as the *safe set*, and note that $\partial\mathcal{S} \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) = 0\}$ and $\text{int}(\mathcal{S}) \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) > 0\}$. This motivates the following definitions of forward invariance and safety:

Definition 1 (*Invariance & Safety*) *A set $\mathcal{S}$ is forward invariant if for every $\mathbf{x}_0 \in \mathcal{S}$, $\mathbf{x}(t) \in \mathcal{S}$ for all $t \in I(\mathbf{x}_0)$. The system (1) is safe with respect to the set $\mathcal{S}$ if the set $\mathcal{S}$ is forward invariant.*

This definition implies that a system remains safe under a particular control policy only if all solutions to (1) that begin in the safe set remain in the safe set.

Control Barrier Functions serve as a synthesis tool for attaining the forward invariance, and thus safety, of a set (Ames et al. (2014); Xu et al. (2015)). Before defining CBFs, we note that a continuous function $\alpha : (-b, a) \to \mathbb{R}$ is an *extended class* $\mathcal{K}$ function ($\alpha \in \mathcal{K}_e$) if $\alpha(0) = 0$ and α is strictly monotonically increasing. If $a, b = \infty$ and $\lim_{r\to\infty} \alpha(r) = \infty$ and $\lim_{r\to-\infty} \alpha(r) = -\infty$, then α is said to be an *extended class* $\mathcal{K}_\infty$ function ($\alpha \in \mathcal{K}_{\infty,e}$). This enables the following definition of Control Barrier Functions:

Definition 2 (*Control Barrier Function (CBF), Ames et al. (2017)*) *Let $\mathcal{S} \subset \mathbb{R}^n$ be the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ with 0 a regular value. h is a Control Barrier Function (CBF) for (1) on $\mathcal{S}$ if there exists $\alpha \in \mathcal{K}_{\infty,e}$ such that for all $\mathbf{x} \in \mathcal{S}$:*

$$\sup_{\mathbf{u}\in\mathbb{R}^m} \dot{h}(\mathbf{x}, \mathbf{u}) = \sup_{\mathbf{u}\in\mathbb{R}^m} \left[\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \left(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}\right)\right] > -\alpha(h(\mathbf{x})). \tag{3}$$

Given a CBF h for (1) and a corresponding $\alpha \in \mathcal{K}_{\infty,e}$, we can consider the point-wise set of all control values that satisfy (3):

$$K_{\text{cbf}}(\mathbf{x}) = \left\{\mathbf{u} \in \mathbb{R}^m \;\middle|\; \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \left(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}\right) \geq -\alpha(h(\mathbf{x}))\right\},$$

One of the main results of (Ames et al. (2014); Xu et al. (2015)) is that the existence of a CBF for $\mathcal{S}$ implies the system (1) can be rendered safe with respect to $\mathcal{S}$:

Theorem 1 *Given a set $\mathcal{S} \subset \mathbb{R}^n$ defined as the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ with 0 a regular value, if h is a CBF for (1) on $\mathcal{S}$, then any Lipshitz continuous controller $\mathbf{k} : \mathbb{R}^n \to \mathbb{R}^m$, such that $\mathbf{k}(\mathbf{x}) \in K_{\text{cbf}}(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{S}$, renders the system (1) safe with respect to the set $\mathcal{S}$.*

This result motivates the construction of a point-wise optimal controller seeking to minimize a cost associated with the choice of input. To this end, we consider the *safety-critical control* formulation (Gurriet et al. (2018)) that seeks to filter a hand-designed but potentially unsafe Lipschitz continuous controller, $\mathbf{k}_d : \mathbb{R}^n \to \mathbb{R}^m$, to find the nearest safe action:

$$\begin{aligned} \mathbf{k}(\mathbf{x}) = \underset{\mathbf{u}\in\mathbb{R}^m}{\text{argmin}} \quad & \frac{1}{2}\|\mathbf{u} - \mathbf{k}_d(\mathbf{x})\|_2^2 \\ \text{s.t.} \quad & \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x}) \left(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}\right) \geq -\alpha(h(\mathbf{x})) \end{aligned} \tag{CBF-QP}$$

The validity of h as a CBF ensures the feasibility of this optimization problem, and the resulting controller is Lipschitz continuous (Morris et al. (2015); Jankovic (2018)). As the dynamics model (1) appears in the constraint enforcing safety, the guarantees of safe control synthesis endowed by this controller require strong assumptions on the accuracy of the dynamics model.

3. Uncertainty Models & Learning

This section provides structural assumptions on the model uncertainty present in our system, and defines our dynamics learning problem in the context of safety. In practice, uncertainty in the dynamics of the system exists due to parametric error and unmodeled dynamics, such that the functions $\mathbf{f}$ and $\mathbf{g}$ are not precisely known. Rather, control synthesis is done with a nominal model that estimates the true dynamics of the system:

$$\widehat{\dot{\mathbf{x}}} = \widehat{\mathbf{f}}(\mathbf{x}) + \widehat{\mathbf{g}}(\mathbf{x})\mathbf{u} \tag{4}$$

where $\widehat{\mathbf{f}} : \mathbb{R}^n \to \mathbb{R}^n$ and $\widehat{\mathbf{g}} : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are Lipschitz continuous. By adding and subtracting this expression to (1), we see that the system evolves under the differential equation given by:

$$\dot{\mathbf{x}} = \widehat{\mathbf{f}}(\mathbf{x}) + \widehat{\mathbf{g}}(\mathbf{x})\mathbf{u} + \underbrace{\mathbf{f}(\mathbf{x}) - \widehat{\mathbf{f}}(\mathbf{x})}_{\mathbf{b}(\mathbf{x})} + \underbrace{\left(\mathbf{g}(\mathbf{x}) - \widehat{\mathbf{g}}(\mathbf{x})\right)}_{\mathbf{A}(\mathbf{x})}\mathbf{u} \tag{5}$$

where $\mathbf{b} : \mathbb{R}^n \to \mathbb{R}^n$ and $\mathbf{A} : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are the unmodeled dynamics. This uncertainty in the dynamics additionally manifests in the time derivative of CBFs designed for the system:

$$\dot{h}(\mathbf{x},\mathbf{u}) = \underbrace{\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})(\widehat{\mathbf{f}}(\mathbf{x}) + \widehat{\mathbf{g}}(\mathbf{x})\mathbf{u})}_{\widehat{\dot{h}}(\mathbf{x},\mathbf{u})} + \underbrace{\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{b}(\mathbf{x})}_{b(\mathbf{x})} + \underbrace{\frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\mathbf{A}(\mathbf{x})}_{\mathbf{a}(\mathbf{x})^\top}\mathbf{u} \tag{6}$$

where $b : \mathbb{R}^n \to \mathbb{R}$ and $\mathbf{a} : \mathbb{R}^n \to \mathbb{R}^m$. The presence of uncertainty in the CBF time derivative makes it impossible to verify if a given control input satisfies the inequality in (3), and can lead to unsafe behavior. One critical assumption we make on the uncertainty in the dynamics is as follows:

Assumption 1 *If a function h is a valid CBF for the nominal model of the system (4), then it is a valid CBF for the uncertain dynamic system (5). Mathematically this assumption appears as:*

$$\sup_{\mathbf{u}\in\mathbb{R}^m} \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\left(\widehat{\mathbf{f}}(\mathbf{x}) + \widehat{\mathbf{g}}(\mathbf{x})\mathbf{u}\right) > -\alpha(h(\mathbf{x})) \implies \sup_{\mathbf{u}\in\mathbb{R}^m} \frac{\partial h}{\partial \mathbf{x}}(\mathbf{x})\left(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}\right) > -\alpha(h(\mathbf{x}))$$

Intuitively this assumption requires that a set within the state space that can be kept safe for the nominal model of the system can also be kept safe for the uncertain system. Technically this assumption amounts to an assumption that the relative degree of h for the model is the same as the relative degree of h for the true system. This assumption on the true system naturally follows if the true system retains the same actuation capability as the model, with more technical details provided in Sastry (1999). While this assumption may enable a robust CBF approach for certifying safety Gurriet et al. (2018), this method may be overly conservative. Instead, we take a data-driven approach similar to (Taylor et al. (2019b)) to learn uncertainty as it appears in the time derivative of the CBF, $\dot{h}$, given in (6).

To motivate our learning framework, consider a simple approach for learning $\mathbf{a}$ and b via supervised regression (Györfi et al. (2006)): an experiment is conducted using a nominal controller to collect data and learn functions that approximate $\mathbf{a}$ and b via supervised regression. More concretely, an experiment consists of sampling an initial state for the system from an initial state distribution $\mathcal{X}_0$, and allowing the system to evolve for a finite time interval under a given state-feedback controller (using a sample-hold implementation). This yields a discrete time state and input history, allowing the computation of a discrete time history of the CBF h. This time history is numerically differentiated to compute an approximate time history of the true value of the time derivative $\dot{h}$, yielding a data-set of input-output pairs:

$$\mathfrak{D} = \{((\mathbf{x}_i, \mathbf{u}_i), \dot{h}_i)\}_{i=1}^{N} \subseteq (\mathbb{R}^n \times \mathbb{R}^m) \times \mathbb{R}. \tag{7}$$

Consider two nonlinear function classes $\mathcal{H}_{\mathbf{a}} : \mathbb{R}^n \to \mathbb{R}^m$ and $\mathcal{H}_b : \mathbb{R}^n \to \mathbb{R}$. For $\widehat{\mathbf{a}} \in \mathcal{H}_a$ and $\widehat{b} \in \mathcal{H}_b$, we define the estimator:

$$\widehat{\dot{S}}(\mathbf{x}, \mathbf{u}) = \widehat{\dot{h}}(\mathbf{x}, \mathbf{u}) + \widehat{\mathbf{a}}(\mathbf{x})^\top \mathbf{u} + \widehat{b}(\mathbf{x}) \tag{8}$$

and let $\mathcal{H}$ be the class of all such estimators mapping $\mathbb{R}^n \times \mathbb{R}^m$ to $\mathbb{R}$. Specifying a loss function $\mathcal{L} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, the supervised regression task is to find an estimator in $\mathcal{H}$ via empirical risk minimization (ERM):

$$\inf_{\substack{\widehat{\mathbf{a}} \in \mathcal{H}_a \\ \widehat{b} \in \mathcal{H}_b}} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(\widehat{\dot{S}}(\mathbf{x}_i, \mathbf{u}_i), \dot{h}_i\right) \tag{9}$$

This experimental framework can be utilized in both simulation (learning estimates of the CBF dynamics when model information is withheld from the state-feedback controller), or directly on hardware platforms (when a given model does not accurately describe the physical system).

4. Episodic Learning Framework

In this section we discuss the consequences of using conventional supervised learning in this context, and formulate an episodic learning framework to overcome these consequences. A critical assumption of supervised learning is independently and identically distributed (i.i.d) training data. The data collected via the preceding experimental framework violates this assumption, as each data point (state and input pair) depends on the value of preceding data points. One potential consequence of violating this assumption when using standard supervising learning is error cascades (Le et al. (2016)). To overcome the impacts of violating this assumption, we extend this experimental protocol to an episodic learning framework.

In particular, this framework iteratively alternates between running experiments with an intermediate controller (or roll-outs in reinforcement learning (Kober et al. (2013))) to collect data, and using the newly collected data to synthesize a new controller. Thus learning $\mathbf{a}$ and b is integrated with improving the safety of the system. The new controller can be synthesized similarly to the safety-critical controller (CBF-QP), but utilizing the updated estimate of the CBF time derivative given in (8):

$$\begin{aligned} \mathbf{k}'(\mathbf{x}) = \ & \underset{\mathbf{u} \in \mathbb{R}^m}{\operatorname{argmin}} \quad \frac{1}{2}\|\mathbf{u} - \mathbf{k}_d(\mathbf{x})\|_2^2 \\ & \text{s.t.} \quad \widehat{\dot{S}}(\mathbf{x}, \mathbf{u}) \geq -\alpha(h(\mathbf{x})) \end{aligned} \tag{LCBF-QP}$$

This controller effectively finds the nearest control input to the hand-designed controller $\mathbf{k}_d$ that renders the system safe according to the estimator (8). Thus, the actual safety achieved by the system degrades if there is error present in (8).

To mitigate this remaining error, we use the newly synthesized controller to improve our estimator. One method for reducing this error is to run experiments and use conventional supervised regression to update the estimator. As previously noted, conventional supervised regression faces limitations with non-i.i.d data. These limitations can be overcome episodically by utilizing reduction techniques, whereby a sequential prediction problem is reduced to a sequence of supervised learning problems (Ernst et al. (2005); Ross et al. (2011)). In each episode, data is generated using a different controller. The data set is aggregated with all previously collected data and a new ERM problem (9) is solved over the full data set. Our approach draws inspiration from the DAgger algorithm (Ross et al. (2010)), with a key difference: DAgger is a model-free policy learning algorithm, which trains a policy directly with an optimal policy oracle. Our method does not receive information on the optimal policy nor does it train a policy directly, but learns information about the unmodeled dynamics impacting safety and indirectly determines a policy via (LCBF-QP).

Lastly, as the newly synthesized controller will expose the system to new regions of the state space in which the estimator may not be accurate, the synthesized controller is blended with the nominal controller using heuristically chosen weights, $w_j \in [0, 1]$, that reflect the trust in the learned model. This effectively limits how quickly the system's behavior can change. The episodic framework is summarized in Algorithm 1.

Algorithm 1 Dataset Aggregation for Control Barrier Functions (DaCBarF)

input : CBF h, CBF derivative estimate $\widehat{\dot{h}}_0$, model classes $\mathcal{H}_{\mathbf{a}}$ and $\mathcal{H}_b$, loss function $\mathcal{L}$, set of initial conditions $\mathcal{X}_0$, nominal state-feedback controller $\mathbf{k}_0$, number of experiments T, sequence of trust coefficients $0 \leq w_1 \leq \cdots \leq w_T \leq 1$

output: Dataset D, CBF derivative estimate $\widehat{\dot{h}}_T$, augmented controller $\mathbf{k}_T$

$D = \emptyset$ // Initialize data set

for $j = 1, \ldots, T$ **do**

  $\mathbf{x}_0 \leftarrow$ sample$(\mathcal{X}_0)$ // Sample initial condition

  $D_j \leftarrow$ experiment$(\mathbf{x}_0, \mathbf{k}_{j-1})$ // Execute experiment

  $D \leftarrow D \cup D_j$ // Aggregate dataset

  $\widehat{\mathbf{a}}, \widehat{b} \leftarrow$ ERM$(\mathcal{H}_{\mathbf{a}}, \mathcal{H}_b, \mathcal{L}, D, \widehat{\dot{h}}_0)$ // Fit estimators

  $\widehat{\dot{h}}_j \leftarrow \widehat{\dot{h}}_0 + \widehat{\mathbf{a}}^\top \mathbf{u} + \widehat{b}$ // Update derivative estimator

  $\mathbf{k}_j \leftarrow (1 - w_j) \cdot \mathbf{k}_0 + w_j \cdot$ augment$(\widehat{\dot{h}}_j)$ // Update controller

end

5. Simulation & Experimental Results

In this section we apply the episodic learning algorithm developed in Sections 3 and 4 to the Segway platform. In particular, we consider a 4-dimensional asymmetrical planar Segway model as in Gurriet et al. (2018), seen in Figure 1 and 2. The state of the system consists of the horizontal

position, x, velocity, $\dot{x}$, pitch angle, θ, and pitch angle rate $\dot{\theta}$. Control input to a motor in the system's wheel is specified as a percentage of the systems battery voltage, which leads to a torque. This model permits an affine control system as per (1). Stabilization of this system to an equilibrium configuration is done using a hand-tuned proportional-derivative (PD) controller, $\mathbf{k}_d$.

Safety of the Segway system is often encoded as limitations on how far and how fast the Segway is allowed to tip from an equilibrium configuration. This appears mathematically as a Control Barrier Function on the pitch angle and pitch angle rate,

$$h(\theta, \dot{\theta}) = \frac{1}{2}\left(\theta_{max}^2 - (\theta - \theta_e)^2 - c\dot{\theta}^2\right), \tag{10}$$

where θ_{max} is a limit on how far the Segway can tip, θ_e is an equilibrium angle, and the coefficient c scales the importance of $\dot{\theta}$ in defining safety. As h is a CBF, the controller (CBF-QP) can be used to ensure the safety of the system.

To evaluate the impact of model uncertainty on the CBF's ability to ensure the safety of the system, we perturb the mass and electrical parameters of the Segway model by up to 15%, but withhold this uncertain model from the controller (CBF-QP). Figure 1 shows that the model-based controller is unable to achieve safety, with the state of the system leaving the 0-superlevel set of h (left), and the value of h falling below 0 (right).

We deploy our episodic learning framework on the uncertain Segway system to overcome this model uncertainty and achieve safety. In particular, we conduct a sequence of 10 experiments with varying initial conditions reflecting disturbances to the Segway. Data from each experiment is aggregated into one dataset that is used to train regression models. For this system we use neural-networks with one hidden layer of 200 nodes for the estimators $\widehat{\mathbf{a}}$ and $\widehat{b}$. These networks are trained to minimize mean absolute error using stochastic gradient descent (SGD)[1]. After each episode the controller (LCBF-QP) is blended with the original controller (CBF-QP) using weights that grow linearly across the episodes. Figure 1 shows the behavior of the system when the controller (LCBF-QP) is deployed after episodic learning. We see that the state of the system is kept within the 0-superlevel set of h (left), with the value of h being kept above 0 (right). This indicates that the learning-augmented controller is able to keep the system safe even with model uncertainty.

To further demonstrate the ability of this learning approach, we deployed it on the physical Segway system seen in Figure 2. The objective of the Segway was specified as tracking a desired horizontal velocity profile while satisfying a barrier function on the pitch angle rate specified as:

$$h(\dot{\theta}) = \frac{1}{2}(1 - c\dot{\theta}^2). \tag{11}$$

The desired velocity profile led to a violation of the barrier function when only enforcing the safety constraint using model information. A series of 3 episodes were conducted tracking this velocity profile without introducing learned information. The estimators $\widehat{\mathbf{a}}$ and $\widehat{b}$ were implemented as neural networks with two hidden layers of 50 nodes and trained on this data after which they were incorporated into the controller using $w_j = 1$. This high weighting was used to see noticeable changes in behavior of the Segway after learning. The improvement in safety using learning can be seen in Figure 2.

1. The models were implemented and trained in Python using the Keras learning package.

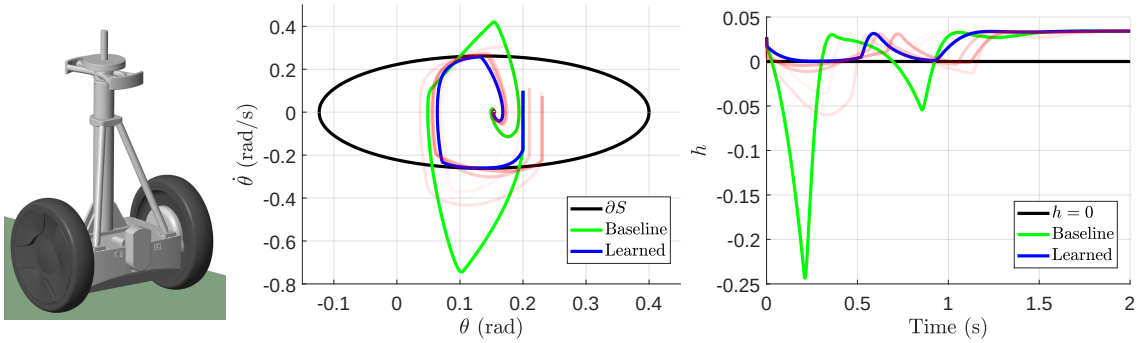

Figure 1. Simulation results using the model-based CBF controller (CBF-QP) (green) and learning-augmented controller (LCBF-QP) (blue). Episodic data appears as red traces. (Left) Segway dynamics simulation model. (Center) The phase portrait showing the evolution of the state of the system leaves the safe set (black ellipse) with the model-based controller, while it remains within using the learning-augmented controller. (Right) The value of h drops below zero using the model-based controller, while it remains above zero using the learning-augmented controller. Video can be found at https://vimeo.com/user106627792/review/380798349/649d4c391e

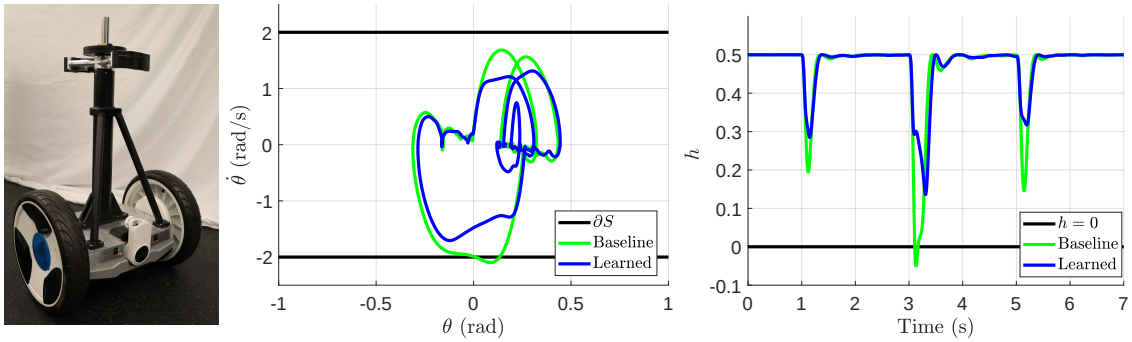

Figure 2. Experimental results using the model-based CBF controller (CBF-QP) (green) and learning-augmented controller (LCBF-QP) (blue). (Left) Custom robotic Ninebot Segway platform. (Center) The phase portrait showing the evolution of the state of the system leaves the safe set (black strip) with the model-based controller, while it remains within using the learning-augmented controller. (Right) The value of h drops below zero (at $t = 3$) using the model-based controller, while it remains above zero using the learning-augmented controller. Video can be found at https://vimeo.com/user106627792/review/380798276/f37b003db3

6. Conclusions & Future Work

We presented an episodic learning framework that integrates with Control Barrier Functions, a model-based nonlinear control method for ensuring safety. Our method is able to learn unmodeled dynamics and parametric error as they directly impact the safety of the system and incorporate this information into an optimization based controller. We demonstrate the ability of this method to improve the safety of a Segway system in simulation and experimentally on hardware.

Future work will seek to investigate the impact of residual error on safe behavior through the analysis established in Taylor et al. (2019a). Furthermore, this work will be used in the development of a safe exploration framework that actively collects data relevant to both the CLF and CBF learning problems. Lastly, this work will be further demonstrated in more complex hardware applications to assess the validity of the assumptions made and demonstrate the overall capability.

References

A. Ames, J. Grizzle, and P. Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *Conference on Decision & Control (CDC)*, pages 6271–6278. IEEE, 2014.

A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada. Control barrier function based quadratic programs for safety critical systems. *Transactions on Automatic Control*, 62(8):3861–3876, 2017.

A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada. Control barrier functions: Theory and applications. In *European Control Conference (ECC)*, pages 3420–3431. IEEE, 2019.

F. Berkenkamp, A. P. Schoellig, and A. Krause. Safe controller optimization for quadrotors with gaussian processes. In *International Conference on Robotics and Automation (ICRA)*, pages 491–496. IEEE, 2016.

F. Blanchini. Set invariance in control. *Automatica*, 35(11):1747–1767, 1999.

R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. In *Conference on Artificial Intelligence*, volume 33, pages 3387–3395. AAAI, 2019.

D. Ernst, P. Geurts, and L. Wehenkel. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6(Apr):503–556, 2005.

J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. Gillula, and C. J. Tomlin. A general safety framework for learning-based control in uncertain robotic systems. *Transactions on Automatic Control*, 2018.

T. Gurriet, A. Singletary, J. Reher, L. Ciarletta, E. Feron, and A. Ames. Towards a framework for realizable safety critical control through active set invariance. In *International Conference on Cyber-Physical Systems (ICCPS)*, pages 98–106. IEEE Press, 2018.

L. Györfi, M. Kohler, A. Krzyzak, and H. Walk. *A distribution-free theory of nonparametric regression*. Springer Science & Business Media, 2006.

M. Jankovic. Robust control barrier functions for constrained stabilization of nonlinear systems. *Automatica*, 96:359–367, 2018.

S. M Khansari-Zadeh and A. Billard. Learning control lyapunov function to ensure stability of dynamical system-based robot reaching motions. *Robotics and Autonomous Systems*, 62(6), 2014.

J. Kober, J. A. Bagnell, and J. Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

S. Kolathaya and A. D. Ames. Input-to-state safety with control barrier functions. *Control Systems Letters*, 3(1):108–113, 2018.

H. M. Le, A. Kang, Y. Yue, and P. Carr. Smooth imitation learning for online sequence prediction. In *International Conference on Machine Learning (ICML)*, pages 680–688. JMLR. org, 2016.

B. J. Morris, M. J. Powell, and A. D. Ames. Continuity and smoothness properties of nonlinear optimization-based feedback controllers. In *Conference on Decision & Control (CDC)*, pages 151–158. IEEE, 2015.

Q. Nguyen and K. Sreenath. Exponential control barrier functions for enforcing high relative-degree safety-critical constraints. In *American Control Conference (ACC)*, pages 322–328. IEEE, 2016.

L. Perko. *Differential equations and dynamical systems*, volume 7. Springer Science & Business Media, 2013.

S. Ross, G. J. Gordon, and J. A. Bagnell. No-regret reductions for imitation learning and structured prediction. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 627–635, 2011.

S. S. Sastry. *Nonlinear Systems: Analysis, Stability and Control*. Springer, NY, 1999.

S. Schaal and C. G. Atkeson. Learning control in robotics. *Robotics & Automation Magazine*, 17 (2):20–29, 2010.

G. Shi, X. Shi, M. O'Connell, R. Yu, K. Azizzadenesheli, A. Anandkumar, Y. Yue, and S. Chung. Neural lander: Stable drone landing control using learned dynamics. In *International Conference on Robotics and Automation (ICRA)*, pages 9784–9790. IEEE, 2019.

A. J. Taylor and A. D. Ames. Adaptive safety with control barrier functions. *arXiv preprint arXiv:1910.00555*, 2019.

A. J. Taylor, V. D. Dorobantu, M. Krishnamoorthy, H. M. Le, Y. Yue, and A. D. Ames. A control lyapunov perspective on episodic learning via projection to state stability. In *Conference on Decision & Control (CDC)*, pages 1448–1455. IEEE, 2019a.

A. J. Taylor, V. D. Dorobantu, H. M. Le, Y. Yue, and A. D. Ames. Episodic learning with control lyapunov functions for uncertain robotic systems. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 6878–6884. IEEE, 2019b.

A. J. Taylor, A. Singletary, Y. Yue, and A. D. Ames. Learning for safety critical control with control barrier functions. *arXiv preprint*, 2019c.

L. Wang, E. A. Theodorou, and M. Egerstedt. Safe learning of quadrotor dynamics using barrier certificates. In *International Conference on Robotics and Automation (ICRA)*, pages 2460–2465. IEEE, 2018.

X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames. Robustness of control barrier functions for safety critical control. *IFAC-PapersOnLine*, 48(27):54–61, 2015.