Data-Driven Safety Filters

HAMILTON-JACOBI REACHABILITY, CONTROL BARRIER FUNCTIONS, AND PREDICTIVE METHODS FOR UNCERTAIN SYSTEMS

KIM P. WABERSICH, ANDREW J. TAYLOR, JASON J. CHOI, KOUSHIL SREENATH, CLAIRE J. TOMLIN, AARON D. AMES, and MELANIE N. ZEILINGER

Summary

Some of the most challenging problems in control typically consist of minimizing an objective function under safety constraints and physical limitations. These often conflicting requirements render classical stabilization-based control design tricky, and even modern learning-based alternatives rarely provide strict safety guarantees 'out-of-the-box'. Safety filters address this limitation through a modular approach to safety. The first part of this article formalizes an ideal safety filter to enhance any controller with safety guarantees and provides a tutorial-style exposition of invariance-based methods using Hamilton-Jacobi reachability, control barrier functions, and predictive control related techniques. While the first part assumes perfect knowledge of the system dynamics, the second part bridges the gap toward real-world applications through data-driven model corrections. To this end, deterministic-, robust-, and probabilistic model learning techniques are outlined, and a selection of mini-tutorials for learning-based safety filters is provided. The article concludes with recent applications to demonstrate the capability of various safety filter formulations when combined with stabilizing controllers, learning-based controllers, and even humans.

Digital Object Identifier 10.1109/MCS.2022.000000
Date of current version: XXXXXX

Today's control engineering problems exhibit an unprecedented complexity, with examples including the reliable integration of renewable energy sources into power grids [1], safe collaboration between humans and robotic systems [2], and dependable control of medical devices [3] offering personalized treatment [4]. In addition to compliance with safety criteria, the corresponding control objective is often multi-faceted. It ranges from relatively simple stabilization tasks to unknown objective functions, which are, for example, only accessible through demonstrations from interactions between robots and humans [5]. Classical control engineering methods are, however, often based on stability criteria with respect to set points and reference trajectories and can therefore be challenging to apply in such unstructured tasks under consideration of potentially conflicting safety specifications [6, Section 3 & 6]. While control experts and, increasingly, learning-based control methods tackle these challenges from various angles, missing safety certificates often prohibit the widespread application of innovative designs outside research environments. As described in "Summary," this article provides an introduction to safety filters together with advanced data-driven enhancements as a flexible framework to overcome these limitations.

To illustrate the fundamental challenges in guaranteeing safety for dynamic systems, consider a vehicle driving on a road as depicted in Figure 1. Depending on a specific vehicle state, including current position, current velocity, and relative heading to the road, taking a particular control action can either maintain a safe system state or put the car at risk. Thereby, the difficulty arising in safety critical

dynamical systems is that unsafe control actions will only cause constraint violations at some point in the future. For example, if the steering angle does not correspond to the road's curvature for a fraction of a second, it might be unavoidable that the car goes off track, as depicted by the red trajectory in Figure 1. Safety filters detect such unsafe input control signals before constraint violations occur and *minimally* modify them to ensure safety, as illustrated by the green trajectory in Figure 1.

Three primary research directions have evolved over the past decades to tackle such safety-critical control problems, providing the core mechanisms of safety filters: Invariance- and reachability-based methods [7]–[10], control barrier functions [11], [12], and predictive control techniques [13], [14]. Although they all address the same fundamental problem of ensuring safety, they have developed relatively independently in their respective research areas. In recent years, however, joint research efforts have demonstrated tremendous potential by combining the core competencies of each field, enabling high-performance safety-critical applications and promising perspectives for future research [15]–[20].

Despite the differences and connections between the fields, all methods rely on a mathematical model that describes the evolution of the dynamic system in order to ensure safety at all times. The derivation, identification, and verification of these high-fidelity system models are among the most time-consuming tasks in the design phase of safety-critical controllers [21]. To reduce this effort, the increasing availability of low-cost sensing and connectivity capabilities and growing computational resources have triggered research efforts across all fields toward the safe use of data-driven models [22]–[24].

This article provides a comprehensive introduction to the previously described aspects of recent safety-critical control research. We present an idealized safety filter problem and demonstrate the capabilities of safety filters based on reachability, control barrier function, and predictive control to provide an approximate solution. Once the basic principles are in place, more recently discovered interconnections between the methods are presented to open new perspectives for future research and applications. It is then shown how to enhance the core concepts through data-driven models and how robust and probabilistic uncertainty bounds can be incorporated to ensure safety with high confidence. While we present a selection of successful techniques and state-of-the-art applications, this direction represents a promising dimension worth investigating in the future. The literature introduced throughout this article is summarized in Table 1, which contains relevant historical developments mentioned at several places.

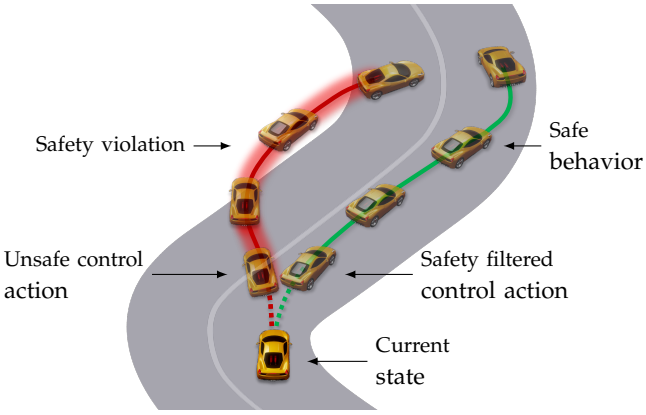


FIGURE 1 Intuitive illustration of safety problems in control using a vehicle. Application of an unsafe control input can result in safety constraint violation at some point in the future. This is depicted by the red trajectory, where the vehicle ends up leaving the track. The goal of this article is to present safety filters, which detect and minimally modify such unsafe inputs to ensure safety for all times.

Outline of the Article

We begin by stating the class of safety-critical nonlinear dynamical systems considered in this article and formalize the desired safety filter module in the form of an optimal control problem. Using this problem formulation, we introduce the fundamental concept of set invariance, followed by techniques for designing and implementing safety filters via Hamilton-Jacobi reachability, control barrier functions, and predictive control methods. The similarities and differences of these three methods are highlighted through a simple illustrative example, and a discussion on recent research efforts integrating aspects of these three methods is provided. In the second part of the article, we consider the challenge of safety-critical control in the context of uncertain nonlinear systems. We discuss how the preceding methods for safety filter design can be modified to incorporate data-driven components, highlighting ongoing research efforts and reviewing examples of state-of-the-art data-driven safety filter applications.

Definitions and Notation

The natural, real, non-negative real, and positive real numbers are denoted as $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}_{\geq 0} = [0, \infty)$, and $\mathbb{R}_{>0} = (0, \infty)$, respectively. The identity matrix of dimension n is denoted as I_n. Given a set $\mathcal{A} \subseteq \mathbb{R}^n$, we denote its interior as $\text{int}(\mathcal{A})$, its boundary by $\partial\mathcal{A}$, and its complement $\mathcal{A}^c = \mathbb{R}^n \setminus \mathcal{A}$. The signed distance function for the set $\mathcal{A}$, $s_{\mathcal{A}} : \mathbb{R}^n \to \mathbb{R}$, is defined as $s_{\mathcal{A}}(x) = \inf_{y \in \mathcal{A}} \|y - x\|$ if $x \in \mathbb{R}^n \setminus \mathcal{A}$ and $s_{\mathcal{A}}(x) = -\inf_{y \in \mathbb{R}^n \setminus \mathcal{A}} \|x - y\|$ for $x \in \mathcal{A}$. Given two sets $\mathcal{A}$ and $\mathcal{B}$, we denote the space of continuous functions, piecewise-continuous functions, and continuously differentiable functions mapping $\mathcal{A}$ to $\mathcal{B}$ by $\mathcal{C}(\mathcal{A}, \mathcal{B})$, $\mathcal{PC}(\mathcal{A}, \mathcal{B})$, and $\mathcal{C}^1(\mathcal{A}, \mathcal{B})$, respectively. A

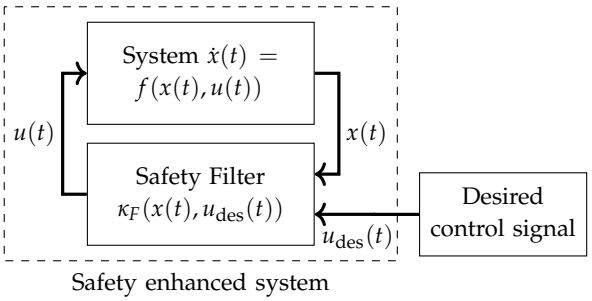


FIGURE 2 Illustration of the safety filter concept. A desired control input $u_{\text{des}}(t)$ is processed by the safety filter to produce a control input signal $u(t) = \kappa_F(x(t), u_{\text{des}}(t))$ that is applied to the system to ensure that safety is maintained at all times.

continuous function $\alpha \in \mathcal{C}([0, a), \mathbb{R})$ for some $a > 0$ is said to be class $\mathcal{K}$ ($\alpha \in \mathcal{K}$) if it is strictly increasing and $\alpha(0) = 0$, and is said to be extended class $\mathcal{K}$ ($\alpha \in \mathcal{K}^e$) if it is a class $\mathcal{K}$ function defined on $(-a, b)$, with $a, b > 0$.

THE SAFETY FILTER PROBLEM WITH KNOWN SYSTEM DYNAMICS

This article considers the construction of safety filtering mechanisms for nonlinear control systems, which can be described by the differential equation

$$\dot{x} = f(x, u), \tag{1}$$

where $x \in \mathbb{R}^{n_x}$ is the system state and $u \in \mathbb{R}^{n_u}$ is the control input. For simplicity, we assume that the function f is continuously differentiable, that is $f \in \mathcal{C}^1(\mathbb{R}^{n_x} \times \mathbb{R}^{n_u}, \mathbb{R}^{n_x})$, and that for any initial condition $x_0 \triangleq x(0) \in \mathbb{R}^{n_x}$ and piecewise-continuous control input signal $u(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathbb{R}^{n_u})$, there exists a unique solution $x(\cdot) \in \mathcal{C}(\mathbb{R}_{\geq 0}, \mathbb{R}^{n_x})$ to (1) for $t \in \mathbb{R}_{\geq 0}$.

Safety for the system (1) is mathematically encoded via a state constraint set $\mathcal{X} \subset \mathbb{R}^{n_x}$ and an input constraint set $\mathcal{U} \subset \mathbb{R}^{n_u}$ that must be respected during the evolution of the system, that is

$$x(t) \in \mathcal{X} \text{ and } u(t) \in \mathcal{U} \text{ for all } t \in \mathbb{R}_{\geq 0}. \tag{2}$$

This article is specifically concerned with ensuring this safety requirement is met when the system is presented with a piecewise-continuous desired control input signal, $u_{\text{des}}(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathbb{R}^{n_u})$, that does not necessarily enforce safety of the system. Such desired input signals are often generated by controllers hand-designed by domain specialists, learning-based controllers that maximize a particular reward signal, or human input to the system.

A safety filter $\kappa_F : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathcal{U}$ (see Figure 2) modifies this desired control input signal to produce an input signal $u(t) = \kappa_F(x(t), u_{\text{des}}(t))$ that ensures the system respects the safety constraint (2), while minimally modifying the desired input signal, that is, minimizing the deviation

$$\int_{t=0}^{\infty} \|\kappa_F(x(t), u_{\text{des}}(t)) - u_{\text{des}}(t)\| \, \mathrm{d}t, \tag{3}$$

with the goal of preserving as much of the desirable behavior achieved by $u_{\text{des}}(\cdot)$ as possible. Thus, for any initial condition $x_0 \in \mathcal{X}$, an ideal safety filter would return a piecewise-continuous input signal u that solves the optimization problem

$$u(\cdot) = \underset{v(\cdot)}{\operatorname{argmin}} \quad \int_{t=0}^{\infty} \|v(t) - u_{\text{des}}(t)\| \, \mathrm{d}t \tag{4a}$$

$$\text{s.t.} \quad v(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U}) \tag{4b}$$

$$x(0) = x_0, \tag{4c}$$

$$\text{for all } t \in \mathbb{R}_{\geq 0}:$$

$$\dot{x}(t) = f(x(t), v(t)), \tag{4d}$$

$$x(t) \in \mathcal{X}. \tag{4e}$$

While (4) characterizes an ideal safety filter, it is rarely possible to tractably implement for the following reasons:

- The desired input signal $u_{\text{des}}(\cdot)$ is typically not known *a priori*, and can only be accessed at the current time t during closed-loop operation. Examples include when $u_{\text{des}}(\cdot)$ is specified by a feedback controller, learning-based control applications with randomized inputs applied during exploration, and applications with humans in the loop directly providing the desired control action, for example, in the teleoperation of robots or for driver assistant systems.
- The optimization problem (4) is not necessarily feasible for each initial condition $x_0 \in \mathcal{X}$. Thus, initial conditions will need to be restricted to a subset $\mathcal{S} \subseteq \mathcal{X}$ of the state constraint set for which (4) is known to be feasible, and the evolution of the system must be constrained to remain in the set $\mathcal{S}$.

We will next tackle these challenges through permissive approximations of the ideal safety filter formulation (4).

SAFETY FILTER METHODOLOGIES

In this section we review three main approaches for approximating the idealized safety filter defined by the optimization problem (4). We begin by reviewing the fundamental notion of set invariance, which underlies all of the presented approaches. The first approach we present builds upon the foundational result of Nagumo's theorem to build a switching safety filter. The conservative nature of this approach is then improved by constructing invariant sets using Hamilton-Jacobi reachability. We next review control barrier functions (CBFs) which rely on a Lyapunov-like derivative condition to smoothly enforce safety of a system. Lastly, we review recent advances in predictive safety filters (PSFs), which utilize a receding-horizon optimal control problem to effectively balance safety with using the desired control input. We outline the strengths

TABLE 1 Overview of safety filter literature. This table presents references regarding the historical development, core results, and recent data-driven research for each of the three safety filter methodologies presented in this work. While it is not a complete description of all related work on these methodologies, this collection of works serves to highlight the strengths of each approach and is a natural starting point for forming a deeper technical understanding of the results presented in this work.

	Hamilton-Jacobi Reachability	Control Barrier Functions	Predictive Filters
Historical Development	[7], [25]–[29]	[11], [30]–[32]	[13], [33], [34]
Core Results	[9], [10], [35], [36]	[12], [37]–[40]	[14], [41]–[43]
Data-driven Safety Filters	[22], [44], [45]	[24], [46]–[54]	[55]–[60]

and weaknesses of each method and apply them to a simple example problem for comparison in "Safety Filter Design Example". We conclude this section by highlighting recent research focused on combining the aforementioned techniques in an effort to overcome the limitations facing each method.

Set Invariance

Set invariance [8] is a well-established notion for studying whether the state of a dynamic system is contained in a prescribed set for all time, and is thereby instrumental in synthesizing safety filters. Given a feedback controller $\kappa : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_u}$, we may construct a closed-loop system

$$\dot{x} = f(x, \kappa(x, u_{\text{des}}(t))) \quad t \in \mathbb{R}_{\geq 0}, \tag{5}$$

allowing the following definition:

Definition 1 (*Set Invariance*)
A set $\mathcal{S} \subset \mathbb{R}^{n_x}$ is said to be *(forward) invariant* for the system (5) if for any initial condition $x_0 \in \mathcal{S}$, we have that $x(t) \in \mathcal{S}$ for all $t \in \mathbb{R}_{\geq 0}$.

If a set $\mathcal{S} \subset \mathbb{R}^{n_x}$ is forward invariant for the system (5), and satisfies $\mathcal{S} \subseteq \mathcal{X}$, then we may conclude that for any initial condition $x_0 \in \mathcal{S}$, we have $x(t) \in \mathcal{X}$ for all $t \in \mathbb{R}_{\geq 0}$. Thus, satisfying the state-related part of the safety constraint (2) can be achieved by constructing a controller κ and a corresponding forward invariant set $\mathcal{S}$ contained in the state constraint set $\mathcal{X}$. We note that this construction via invariance requires not only a stronger condition on the initial condition x_0, in that it must lie in $\mathcal{S}$ rather than just $\mathcal{X}$, but it also yields a stronger statement since $x(t) \in \mathcal{S}$ for all $t \in \mathbb{R}_{\geq 0}$ rather than just $x(t) \in \mathcal{X}$. Thus, the particular construction of the feedback controller κ and the forward invariant set $\mathcal{S}$ impacts the resulting performance of the system because the use of a conservative set $\mathcal{S}$ may unnecessarily limit the behavior of the system.

The notion of a control invariant set captures the possibility of controlling the open-loop system (1) in a safe manner, without being confined to using a predefined feedback controller κ and then determining a forward invariant set $\mathcal{S}$ for the closed-loop system under κ:

Definition 2 (*Controlled Set Invariance*)
A set $\mathcal{S} \subset \mathbb{R}^{n_x}$ is said to be *control invariant* for the system (1) if for any initial condition $x_0 \in \mathcal{S}$, there exists a piecewise-continuous input signal $u(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathbb{R}^{n_u})$ such that we have $x(t) \in \mathcal{S}$ for all $t \in \mathbb{R}_{\geq 0}$. If $\mathcal{S} \subseteq \mathcal{X}$ contains all initial conditions $x_0 \in \mathcal{X}$ such that there exists a piecewise continuous input signal $u(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathbb{R}^{n_u})$ yielding $x(t) \in \mathcal{X}$ for all $t \in \mathbb{R}_{\geq 0}$, we say that $\mathcal{S}$ is the *maximal control invariant set in* $\mathcal{X}$ for the system (1).

This definition enables various safety-filter design techniques given a control invariant set $\mathcal{S}$. Conversely, since finding a control invariant set $\mathcal{S}$ is not restricted to any specific controller, choosing $\mathcal{S}$ can also be done in a more constructive manner. The performance achieved using a safety filter will directly depend on the size of the control invariant set. Ideally, the set $\mathcal{X}$ would be used, however, this is typically not a control invariant set for (1) and merely represents the design goal. We will see in the following subsections that available safety-filter techniques produce different feedback controllers and (control) invariant sets that permit varying degrees of performance.

Nagumo's Theorem and Switching Safety Filters

The first safety filter design that we consider is that of a switching safety filter. Although this design is relatively simple and often overly conservative, it highlights key elements that arise in the three advanced safety filter approaches presented next.

Consider a feedback controller

$$\kappa_{\mathcal{S}} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_u}, \tag{6}$$

and a set $\mathcal{S} \subseteq \mathcal{X}$ defined as the 0-superlevel set of a continuously differentiable function $h_{\mathcal{S}} : \mathbb{R}^{n_x} \to \mathbb{R}$

$$\mathcal{S} = \{x \in \mathbb{R}^{n_x} \mid h_{\mathcal{S}}(x) \geq 0\}, \tag{7a}$$

$$\text{int}(\mathcal{S}) = \{x \in \mathbb{R}^{n_x} \mid h_{\mathcal{S}}(x) > 0\}, \tag{7b}$$

$$\partial\mathcal{S} = \{x \in \mathbb{R}^{n_x} \mid h_{\mathcal{S}}(x) = 0\}. \tag{7c}$$

Suppose the set $\mathcal{S}$ is forward invariant for the closed-loop system

$$\dot{x} = f(x, \kappa_{\mathcal{S}}(x)) \tag{8}$$

and that $\kappa_{\mathcal{S}}(x) \in \mathcal{U}$ for all $x \in \mathcal{S}$. A classic example of this setting is when $\kappa_{\mathcal{S}}$ is a locally stabilizing controller for

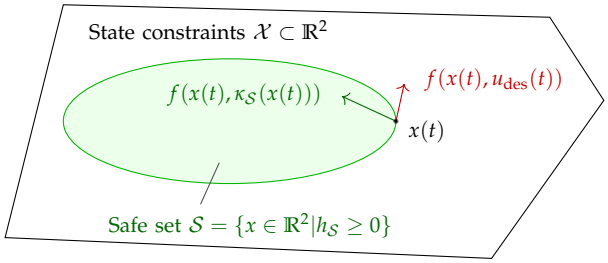


FIGURE 3 Geometric interpretation of Nagumo's Theorem. The switching safety filter (11) builds directly off the condition (10) in Nagumo's Theorem to enforce safety. At the state $x(t)$, the desired input $u_{\text{des}}(t)$ will cause the system to leave the safe set $\mathcal{S}$ since the vector $f(x(t), u_{\text{des}}(t))$ points outward with respect to the set $\mathcal{S}$. Switching to the safe control law (6) as dictated by (11) leads to the system remaining inside the set since the vector $f(x(t), \kappa_{\mathcal{S}}(x(t)))$ points inward with respect to the set $\mathcal{S}$.

some equilibrium point $x_e \in \text{int}(\mathcal{X})$ and $\mathcal{S}$ is the sublevel-set of a corresponding Lyapunov function. In this setting, $\kappa_{\mathcal{S}}$ is often synthesized based on a linearization of the nonlinear dynamics (1) at the equilibrium point x_e, and thus the level of the Lyapunov function must be chosen relatively small. A small set leads to conservative behavior of the safety filter, and will motivate later constructions with Hamilton-Jacobi reachability and predictive safety filters.

Expressing $\mathcal{S}$ as the 0-superlevel set of the continuously differentiable function $h_{\mathcal{S}}$ allows us to consider a fundamental result in studying set invariance established in 1942 and known as *Nagumo's Theorem* [61]:

Theorem 1

Consider the closed-loop system (8) and a set $\mathcal{S} \subseteq \mathcal{X}$ defined as the 0-superlevel set of a continuously differentiable function $h_{\mathcal{S}} : \mathbb{R}^{n_x} \to \mathbb{R}$ with $\text{int}(\mathcal{S}) \neq \emptyset$ and

$$\nabla h_{\mathcal{S}}(x) \triangleq \frac{\partial h_{\mathcal{S}}}{\partial x}(x) \neq 0 \tag{9}$$

for all $x \in \partial\mathcal{S}$. Then the set $\mathcal{S}$ is forward invariant for (8) if and only if

$$\dot{h}_{\mathcal{S}}(x) \triangleq \nabla h_{\mathcal{S}}(x) f(x, \kappa_{\mathcal{S}}(x)) \geq 0, \tag{10}$$

for all $x \in \partial\mathcal{S}$.

The requirement (10) of Nagumo's Theorem has a simple geometric interpretation as seen in Figure 3. In particular, the vector given by the closed-loop dynamics (8) must point into the set $\mathcal{S}$ at each point on its boundary. Moreover, it is a necessary and sufficient condition for the forward invariance of the set $\mathcal{S}$, implying that the inequality in (10) is satisfied for all $x \in \partial\mathcal{S}$ since $\mathcal{S}$ is forward invariant for the closed-loop dynamics (8).

This property on the boundary of the set $\mathcal{S}$ allows the construction of a simple safety filter that switches between using the desired control input signal u_{des} and the controller $\kappa_{\mathcal{S}}$. More precisely we construct a safety filter $\kappa_F : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathcal{U}$ as

$$\kappa_F(x, u_{\text{des}}(t)) = \begin{cases} \kappa_{\mathcal{S}}(x), & x \in \partial\mathcal{S} \text{ or } u_{\text{des}}(t) \notin \mathcal{U}, \\ u_{\text{des}}(t), & \text{else.} \end{cases} \tag{11}$$

Such a switching-based safety mechanism was originally proposed in [62]. It is straightforward to see that

$$\nabla h_{\mathcal{S}}(x) f(x, \kappa_F(x, u_{\text{des}}(t))) \geq 0, \tag{12}$$

for all $x \in \partial\mathcal{S}$ by virtue of $\kappa_{\mathcal{S}}$ satisfying (10) for all $x \in \partial\mathcal{S}$. Thus we may conclude that $\mathcal{S}$ is forward invariant using the proposed safety filter by Nagumo's Theorem.

We note that the form of the safety filter (11) is not entirely rigorous because instantaneous switches at the boundary of the system may not yield a piecewise-continuous input signal if the switches occur infinitely often in a finite period of time (commonly known as Zeno behavior). This issue can be resolved both theoretically and practically by requiring the controller $\kappa_{\mathcal{S}}$ to be used for a short time interval when activated. The choice of this time interval has practical consequences, since short intervals can yield undesirable chattering behavior, while large intervals can lead to the safety filter rarely using the desired control input signal $u_{\text{des}}(\cdot)$. The main benefit of constructing the switching-based safety filter (11) is its simplicity of implementation whenever access to a controller $\kappa_{\mathcal{S}}$ and a corresponding forward invariant set $\mathcal{S}$ is available.

Hamilton-Jacobi Reachability Analysis for Safe Set Synthesis

In this section we seek to reduce the conservativeness of the preceding switching safety filter design by constructively synthesizing the maximal control invariant set $\mathcal{S}$ in $\mathcal{X}$. We will achieve this through the use of Hamilton-Jacobi (HJ) reachability [10]. The notion of the maximal control invariant set in $\mathcal{X}$ was first developed in the context of viability theory [7], resulting in the definition of sets known as viability kernels introduced below. The first efforts in characterizing viability kernels were led by the viability theory community [7], with a focus on characterizing the sets geometrically. The connections between viability kernels and reachable sets were then established [29], leading to the development of effective computational frameworks [9], [35] for discovering viability kernels through dynamic programming-based algorithms [25]. In this article, we focus on the category of reachability and viability concepts related to the problem of state constraint satisfaction. However, it must be noted that reachability analysis captures a broad collection of set-based concepts relevant for system verification, such as reach-avoid sets [36], and has been extended to hybrid systems [27], [28], [63]. The application of HJ reachability to verify safety of autonomous aerial and mobile vehicles is detailed in "Hamilton-Jacobi Reachability Safety Filter Applications".

We now present the definition of viability kernels:

Definition 3 (*Viability Kernel*)
Given a state constraint set $\mathcal{X}$ and a time horizon $T \in \mathbb{R}_{\geq 0}$, we define the *viability kernel of* $\mathcal{X}$ as

$$\mathcal{VK}_T(\mathcal{X}) = \{x_0 \in \mathcal{X} \mid \text{there exists } u(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U}) \text{ s.t. } x(t) \in \mathcal{X} \text{ for all } t \in [0, T]\}. \quad (13)$$

The viability kernel captures all initial conditions x_0 in the set $\mathcal{X}$ for which it is possible to choose a feasible input signal $u(\cdot)$ that ensures the closed-loop system remains within the set $\mathcal{X}$ over the time horizon T. Naturally, we can extend this idea to an infinite horizon by requiring the state to remain in $\mathcal{X}$ for all time, yielding a set $\mathcal{VK}_\infty(\mathcal{X}) \subseteq \mathcal{X}$ which is by definition the maximal control invariant set in $\mathcal{X}$ for which it is possible to satisfy input constraints.

The connections between viability kernels and reachable sets (to be defined) allow the computation of the viability kernel to be posed as an optimal control problem [29]. To see this, let $s_{\mathcal{X}} : \mathbb{R}^{n_x} \to \mathbb{R}$ denote the signed-distance function for the set $\mathcal{X}$ (see Notation at the beginning of this article). The satisfaction of state constraints requires that $s_{\mathcal{X}}(x(t)) \leq 0$ for all $t \in \mathbb{R}_{\geq 0}$. Equivalently, we may consider a cost functional $J : \mathbb{R}^{n_x} \times \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U}) \to \mathbb{R}$ defined as

$$J(x_0, u(\cdot)) = \inf_{t \in \mathbb{R}_{\geq 0}} -s_{\mathcal{X}}(x(t)), \quad (14)$$

where the state constraints are satisfied if and only if $J(x_0, u(\cdot)) \geq 0$. This cost functional enables us to define a value function $V : \mathbb{R}^{n_x} \to \mathbb{R}$ as

$$V(x_0) = \sup_{u(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U})} J(x_0, u(\cdot)). \quad (15)$$

The value function defines an optimal control problem with the objective of maximizing (14) across all feasible control input signals, ultimately to ensure it is nonnegative and thereby implying the satisfaction of state and input constraints. This value function captures several core concepts for the reachability-based safety filter design. First, it serves as a metric for quantifying safety margins, with negative values indicating violation of safety at some point in the future, and with larger positive values of the value function reflecting more margin (because it is possible to keep the system's state further from the boundary of $\mathcal{X}$ through control), as captured in the following result [22]:

Theorem 2

For any $\epsilon \in \mathbb{R}_{\geq 0}$, we have that the ϵ-superlevel set of the value function $V : \mathbb{R}^{n_x} \to \mathbb{R}$ defined in (15), denoted as

$$\mathcal{S}_\epsilon = \{x \in \mathcal{X} \mid V(x) \geq \epsilon\}, \quad (16)$$

is a control invariant set for (1), and $\mathcal{S}_0$ is the *maximal control invariant set* in $\mathcal{X}$ for (1). Moreover, if V is differentiable and $\mathcal{U}$ is compact, for all $x \in \mathcal{S}_0$ we have that

$$\max_{u \in \mathcal{U}} \nabla V(x) f(x, u) \geq 0. \quad (17)$$

The preceding theorem establishes that we can construct the maximal control invariant set in $\mathcal{X}$ for (1), $\mathcal{S}_0$, through the value function V. The complement of this set captures all the states from which the system will inevitably reach the set $\mathcal{X}^c$ (the complement of $\mathcal{X}$) regardless of the choice of control, thereby violating state constraints. This establishes the connection between the viability kernel of $\mathcal{X}$ and the backward reachable tube of the set $\mathcal{X}^c$ as

$$\mathcal{S}_0 = \mathcal{VK}_\infty(\mathcal{X}) = (\mathcal{BRT}_\infty(\mathcal{X}^c))^c, \quad (18)$$

where the (minimal) backward reachable tube $\mathcal{BRT}$ of a set $\mathcal{C} \subset \mathbb{R}^n$ is defined as

$$\mathcal{BRT}_T(\mathcal{C}) = \{x_0 \in \mathbb{R}^n \mid \text{for all } u(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U}), \text{ there exists } t \in [0, T] \text{ s.t. } x(t) \in \mathcal{C}\}. \quad (19)$$

Thus, the boundary of the maximal control invariant set $\mathcal{S}_0$, which is characterized by the 0-level set of V, discriminates the region of the state space in which violating safety is inevitable from the region in which satisfying the safety constraints is feasible. In practice, using a control invariant set for (1) smaller than $\mathcal{S}_0$, which can be produced by considering ϵ-superlevel sets of V as noted in Theorem 2, provides a tunable buffer for accommodating errors when V is numerically approximated.

The value function can also be used to synthesize a control policy which can be directly incorporated into the safety filter design. If the value function V is differentiable, we can construct an optimal safe policy $\kappa_V^* : \mathbb{R}^{n_x} \to \mathcal{U}$ satisfying

$$\nabla V(x) f(x, \kappa_V^*(x)) = \max_{u \in \mathcal{U}} \nabla V(x) f(x, u). \quad (20)$$

By construction, we have that

$$\dot{V}(x) = \nabla V(x) f(x, \kappa_V^*(x)) \geq 0, \quad (21)$$

from (17). We note that if for some $\epsilon \in \mathbb{R}_{\geq 0}$ we have that $\nabla V(x) \neq 0$ for all $x \in \partial \mathcal{S}_\epsilon$, this condition coincides with the necessary and sufficient condition of Nagumo's Theorem for the set $\mathcal{S}_\epsilon$ to be forward invariant under the control law κ_V^*.

Similarly to the switching safety filter in (11), given a desired $\epsilon \in \mathbb{R}_{\geq 0}$ we can design a switching safety filter based on the value of $V(x(t))$,

$$\kappa_F(x, u_{\text{des}}(t)) = \begin{cases} \kappa_V^*(x), & V(x) \leq \epsilon \text{ or } u_{\text{des}}(t) \notin \mathcal{U} \\ u_{\text{des}}(t), & \text{else.} \end{cases} \quad (22)$$

If we take $\epsilon = 0$, this safety filter is least restrictive [28], [64] in the sense that the filter only intervenes at the boundary of the (approximate) maximal control invariant set in $\mathcal{X}$. As before, it is necessary to use the controller κ_V^* for a short period of time when it is activated to avoid rapid

switching, though this controller often practically displays less chattering than the naive switching safety filter (11).

Computing the value function V is the main task in constructing the safety filter (22), since it determines the ϵ-superlevel sets $\mathcal{S}_\epsilon$ and the optimal safe policy κ_V^*. The value function can be characterized as a solution of a Hamilton-Jacobi Variational Inequality (HJ-VI)

$$0 = \min\left\{ -s_{\mathcal{X}}(x) - V(x),\ \max_{u\in\mathcal{U}} \nabla V(x) f(x,u) \right\}, \quad (23)$$

that can be derived from the dynamic programming principle [36]. The HJ-VI (23) does not necessarily admit unique solutions. In practice the existence of a unique solution can be ensured by using a discounted formulation of the HJ-VI [65], [66], or using a finite-horizon value function (replacing the time horizon in (14) with $[0, T]$) which approximates V for sufficiently large $T \in \mathbb{R}_{>0}$ [22]. Furthermore, if V defined as in (15) is not differentiable, it is still the viscosity solution of (23), which is a standard type of weak solution for partial differential equations not necessarily possessing a differentiable solution [67]. In the presence of such non-differentiability, the optimal safe policy κ_V^* can be constructed using the notion of sub- and super-differentials [26, Ch. III.3.4]. We note that under mild assumptions on the dynamics (1) and the signed distance function $s_{\mathcal{X}}$, the discounted and finite-horizon value function used to approximate the infinite-time value function are almost everywhere differentiable, implying the applicability of κ_V^* satisfying (20).

Algorithms for numerically computing the value function have been well developed [68], primarily through the notion of viscosity solutions [26], [67] and level-set methods for solving partial differential equations [69]. These algorithms typically rely upon forming a grid on the set $\mathcal{X}$ and evaluating the value function, its gradient, and the Hamiltonian (the left-hand side of (17)) at each grid point. Consequently, these approaches face challenges with problems possessing high-dimensional state spaces, a traditional challenge in dynamic programming known as the "curse of dimensionality" [25]. Recent research efforts have attempted to alleviate this challenge by using state decompositions [70], warm starting [71], approximating solutions with neural networks [72], or learning through reinforcement learning [73]. Other works attempt to compute the maximal control invariant set approximately without relying on the HJ reachability formulation through sums-of-squares programming [74], [75], or polytopes [76], ellipsoids [77] and zonotopes [78], [79]-based set operations.

Similarly to the switching safety filter in (11), the reachability-based safety filter in (22) relies on instantaneously switching the control input from $u_{\text{des}}(t)$ to $\kappa_V^*(x(t))$ when the system encounters the boundary of $\mathcal{S}_\epsilon$. This switching mechanism is often vulnerable to disturbances affecting the system dynamics (1). The instantaneous jumps in the control input can also produce chattering which is infeasible on real-world systems due to actuator dynamics and wear. Reachability theory may be extended to incorporate disturbances, resulting in an invariant set $\mathcal{S}_\epsilon$ that is robust to disturbances [35], [36]. Additionally, the transition from $u_{\text{des}}(t)$ to $\kappa_V^*(x)$ in (22) as $V(x)$ approaches ϵ can be moderated in a smooth manner by blending the two control input values.

Safety Filters using Control Barrier Functions

Control barrier functions (CBFs) provide an alternative framework for constructing safety filters based on the comparison principle, a fundamental idea in the study of nonlinear systems [80]. Through this approach it is possible to construct safety filters that smoothly modify a desired input control signal as the boundary of a set is approached, rather than switching to a safe controller only at the boundary. Moreover, the shared use of the comparison principle establishes deep connections between CBFs and Lyapunov functions, allowing a large set of tools developed in the context of stabilization to be adapted for the goal of achieving set invariance.

Historically, barrier methods were first developed in the context of constrained optimization [81], wherein constraint satisfaction could be achieved through increasingly large penalties on constraint violation. The idea to use barrier certificates in the context of nonlinear dynamical systems was first proposed in [30] for certifying the forward invariance of a set for a closed-loop system. This result was further developed in [11], yielding the first definition of CBFs as a tool for simultaneously synthesizing a safety-critical controller and a barrier certificate for the corresponding closed-loop system. The controller presented in this work was based on a structured design developed with control Lyapunov functions for stabilization in [82]. A consequence of this structured design was that the controller could not accommodate a desired control input signal that focused on performance instead of safety, making it unamenable for use as a safety filter.

A change to the formulation of CBFs that increased their potential for use as safety filters was proposed in [32]. The first component of this change was incorporating an extended class $\mathcal{K}$ function into the CBF time derivative condition required for safety. This change allowed the system state to approach the boundary of the safe set as long as it displayed a safe degree of "braking", reducing the conservative nature of the original definition of CBFs. The second component of this change was realizing that for control-affine systems, the CBF time derivative was affine in the control input, and thus could be directly incorporated as a constraint in a convex optimization problem. This resulted in a way to optimally filter a desired control input signal while meeting safety constraints.

We now review this formulation of CBFs as presented

in [12]. We study a broad subset of the class of systems described by (1) in the form of a control-affine nonlinear system

$$\dot{x} = f(x) + g(x)u, \tag{24}$$

making similar assumptions on differentiability and the existence and uniqueness of solutions as made for (1). Given a feedback controller $\kappa : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_u}$, we may construct a closed-loop system

$$\dot{x} = f(x) + g(x)\kappa(x, u_{\text{des}}(t)), \tag{25}$$

for which we have the following definition:

Definition 4 (*Barrier Function*)
Let $\mathcal{S} \subseteq \mathcal{X}$ be defined as the 0-superlevel set of a continuously differentiable function $h_{\mathcal{S}} : \mathbb{R}^{n_x} \to \mathbb{R}$. The function $h_{\mathcal{S}}$ is a *barrier function* (BF) for (25) on $\mathcal{S}$ if there exists $\alpha \in \mathcal{K}^{\text{e}}$ such that for any $x \in \mathbb{R}^{n_x}$, we have that

$$\dot{h}_{\mathcal{S}}(x,t) \triangleq \nabla h_{\mathcal{S}}(x)(f(x) + g(x)\kappa(x, u_{\text{des}}(t))) \geq -\alpha(h_{\mathcal{S}}(x)). \tag{26}$$

The following theorem is proven through comparison principles (as opposed to the boundary conditions seen in Nagumo's theorem), and establishes how a barrier function serves as a certificate of set invariance [38]:

Theorem 3

Let $\mathcal{S} \subset \mathcal{X}$ be defined as the 0-superlevel set of a continuously differentiable function $h_{\mathcal{S}} : \mathbb{R}^{n_x} \to \mathbb{R}$. If $h_{\mathcal{S}}$ is a BF for (25) on $\mathcal{S}$, then the set $\mathcal{S}$ is forward invariant for the system (25).

This theorem states that if the closed-loop dynamics (25) satisfy the inequality in (26) at each point in the state space, the set $\mathcal{S}$ is forward invariant for (25). We observe two notable properties of the requirement in (26). The first property is that the time derivative of $h_{\mathcal{S}}$ must be lower bounded by a quantity that increases as $h_{\mathcal{S}}$ gets smaller. This induces a "braking" effect on the system, where it may not approach the boundary of $\mathcal{S}$ too quickly. The second property is that the time derivative of $h_{\mathcal{S}}$ must be positive outside of the set $\mathcal{S}$. This induces a type of asymptotic stability of the set $\mathcal{S}$, and plays a fundamental role in CBF safety filters robustness to disturbances and model uncertainty [39].

As previously discussed, it is often easier to synthesize a safety filter given a control invariant set, rather than constructing a forward invariant set given a feedback controller. To this end, we define the notion of CBFs:

Definition 5 (*Control Barrier Function*)
Let $\mathcal{S} \subseteq \mathcal{X}$ be defined as the 0-superlevel set of a continuously differentiable function $h_{\mathcal{S}} : \mathbb{R}^{n_x} \to \mathbb{R}$. The function $h_{\mathcal{S}}$ is a *control barrier function* (CBF) for (24) on $\mathcal{S}$ if there exists $\alpha \in \mathcal{K}^{\text{e}}$ such that for any $x \in \mathbb{R}^{n_x}$, we have that

$$\sup_{u \in \mathcal{U}} \nabla h_{\mathcal{S}}(x)(f(x) + g(x)u) > -\alpha(h_{\mathcal{S}}(x)). \tag{27}$$

The strict inequality in this constraint is critical for proving regularity properties such as Lipschitz continuity of resulting controller designs [40]. Given a CBF for (24) on $\mathcal{S}$, we can define a pointwise set

$$K_{\text{CBF}}(x) = \{u \in \mathcal{U} \mid \nabla h_{\mathcal{S}}(x)(f(x) + g(x)u) \geq -\alpha(h_{\mathcal{S}}(x))\}, \tag{28}$$

and state the following result regarding the connection between a CBF and a BF:

Theorem 4

Let $\mathcal{S} \subset \mathcal{X}$ be defined as the 0-superlevel set of a continuously differentiable function $h_{\mathcal{S}} : \mathbb{R}^{n_x} \to \mathbb{R}$. If $h_{\mathcal{S}}$ is a CBF for (25) on $\mathcal{S}$, then the set $K_{\text{CBF}}(x)$ is non-empty for all $x \in \mathbb{R}^{n_x}$, and for any locally Lipschitz continuous controller $\kappa : \mathbb{R}^{n_x} \to \mathbb{R}^{n_u}$ with $\kappa(x) \in K_{\text{CBF}}(x)$ for all $x \in \mathbb{R}^{n_x}$, the function $h_{\mathcal{S}}$ is a BF for (25) on $\mathcal{S}$.

Given this result, we may directly use a CBF to synthesize a safety filter through the convex optimization problem

$$\begin{aligned} \kappa_F(x, u_{\text{des}}(t)) = \underset{u \in \mathcal{U}}{\text{argmin}} \; & \frac{1}{2}\|u - u_{\text{des}}(t)\|_2^2 \\ \text{s.t. } & \nabla h_{\mathcal{S}}(x)(f(x) + g(x)u) \geq -\alpha(h_{\mathcal{S}}(x)). \end{aligned} \tag{29}$$

This controller is a convex quadratic program that may be efficiently solved. By construction it satisfies $\kappa_F(x, u_{\text{des}}(t)) \in K_{\text{CBF}}(x)$ for all $x \in \mathbb{R}^{n_x}$, such that the conditions of Theorem 4 are met, and thus by Theorem 3 we can conclude the set $\mathcal{S}$ is forward invariant for (25). Moreover, it allows the desired input signal $u_{\text{des}}(\cdot)$ to be minimally modified, such that $u_{\text{des}}(t)$ is not used only when it is unsafe, and the input $u(t)$ actually used is as close as possible to $u_{\text{des}}(t)$.

The preceding controller has been deployed in several experimental contexts, including mobile robots [83], robotic swarms [84], aerial vehicles [85], robotic arms [86], robotic manipulators [87], quadrupedal robots [88], bipedal robots [50], and automotive systems [89]. A more detailed overview of some of these applications can be found in "Control Barrier Function Safety Filter Applications". This collection of successful practical applications indicate that CBFs are a powerful tool for safety filter design for complex, high-dimensional nonlinear systems.

Despite these successes, there remain challenges and limitations facing CBF-based safety filters. A key challenge lies in constructively synthesizing CBFs and verifying that the condition in (27) can be met over the state space (or over some limited part of the state space), especially with bounded inputs. For relatively simple sys-

tems it is often possible to check this condition analytically, but it can be difficult to verify for more complex high-dimensional systems. Recent attempts to solve this challenge have considered numerical optimization-based approaches through sums-of-squares programming [90], [91], using reduced-order models coupled with approaches for handling the full-order system dynamics [92], [93], or learning CBFs from data [94]. Still, well-established and principled methodologies for finding CBFs remains an open research question.

A further limitation of CBFs is their myopic approach for approximating the ideal safety filter in (4). In particular, the safety filter in (29) modifies the desired input signal $u_{\text{des}}(\cdot)$ to minimize an instantaneous deviation, rather than minimizing the total deviation over a time horizon. This can lead to sub-optimal closed-loop behavior because the CBF safety filter considers how the chosen input will impact the system in the future only as captured by an instantaneous derivative condition. Recovering performance over a time horizon while achieving safety motivates studying predictive safety filters that plan inputs over a time horizon, as presented next.

Predictive Safety Filters

The previously discussed safety filter methods rely on an explicit characterization of the safe set. The underlying computations are typically limited in scalability as in the case of HJ reachability, or are myopic in nature as with CBF-based methods. Recent concepts such as active set methods [41], SHERPA [42], model predictive safety certification (MPSC) [14], predictive safety filters (PSF) [55], and predictive shielding [43] aim at addressing this challenge and provide a trade-off between scalability and performance by a just-in-time computation of predictive backup plans. We specifically focus on predictive safety filters (PSFs) [14], [55] in the following due to their close relation with (data-driven) model predictive control literature [23], [60], [95]. This connection provides PSFs with an extensive theoretical background covering a variety of system model classes with uncertainties and data-driven estimates, and efficient computational tool sets for their implementation [13].

PSFs are based on the idea of extending a potentially conservative control invariant terminal safe set $\mathcal{S}^t$ using predictive backup plans. More precisely, for a time $t_0 \in \mathbb{R}_{\geq 0}$, consider the system state $x(t_0)$ and the desired input $u_{\text{des}}(t_0)$. Letting $T \in \mathbb{R}_{>0}$ be a prediction horizon, the safety of the desired input $u_{\text{des}}(t_0)$ is certified by searching for a state trajectory $x(\cdot) \in \mathcal{C}([t_0, t_0+T], \mathcal{X})$ and an input signal $u(\cdot) \in \mathcal{PC}([t_0, t_0+T], \mathcal{U})$ satisfying the system dynamics (1) and the boundary conditions $x(t_0) = x(t_0)$, $x(t_0+T) \in \mathcal{S}^t$, and $u(t_0) = u_{\text{des}}(t_0)$. If such a state trajectory and input signal exists, it is possible to use the desired control input and bring the system from the state

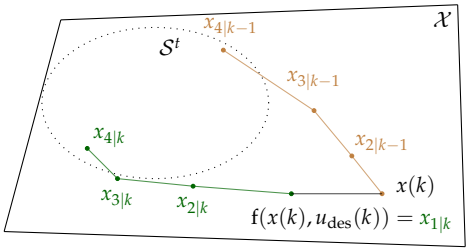


FIGURE 4 Mechanism of predictive safety filters. The current system state $x(k)$ is shown with a safe backup plan (brown) from the solution at time $k-1$. A desired input signal $u_{\text{des}}(k)$ is passed through unfiltered if a feasible backup trajectory (green) can be obtained from the resulting $\mathrm{f}(x(k), u_{\text{des}}(k))$ via optimization problem (31).

$x(t_0)$ into the set $\mathcal{S}^t$ within the finite prediction horizon T, while respecting state and input constraints. If the desired input $u_{\text{des}}(t_0)$ can not be certified as safe, a safe control input is chosen for which the system can be brought to the terminal safe set $\mathcal{S}^t$.

We note that because the PSF is implemented with a receding horizon, the actual evolution of the system is not required to follow the backup plan into the terminal safe set $\mathcal{S}^t$. Rather, the system will evolve using the input at the beginning of the predictive horizon (which must be consistent with a backup plan that returns to S^t further in the horizon), after which it will compute a new backup plan. In this way, the system is allowed to freely evolve according to $u_{\text{des}}(\cdot)$, and does not need to actually return to $\mathcal{S}^t$, as long as it remains possible to return to $\mathcal{S}^t$ in the future, see Figure 4 for an illustration. Consequently, the set kept forward invariant is implicitly defined by the feasible set of the resulting predictive control problem and thereby by the PSF parameters and terminal safe set $\mathcal{S}^t$, rather than explicitly represented as the superlevel set of a function.

Implementing a PSF requires solving a predictive control problem online. While efficient solvers are available [13], they require a non-negligible evaluation time period compared with CBF or HJ reachability-based safety filters. During an evaluation time period $\Delta T \in \mathbb{R}_{>0}$, the previous input is typically held constant, resulting in zero-order-hold input signals, that is, $u(t) = \kappa_F(x(k\Delta T), u_{\text{des}}(k\Delta T))$ for all $t \in [k\Delta T, (k+1)\Delta T)$, where $k \in \mathbb{N}$ denotes the corresponding sampling time step. Applying a standard Euler discretization to (1) yields an approximate discrete-time, zero-order hold formulation of the continuous-time system model (1)

$$x(k+1) = x(k) + \Delta T f(x(k), u(k)) = \mathrm{f}(x(k), u(k)). \quad (30)$$

Let N be the discrete-time prediction horizon. At the sampling time step k the construction of a safe backup trajectory $\{x_{i|k}\}$ for $i = 1, \ldots, N$ toward the terminal safe

set $\mathcal{S}^t$ is formulated as

$$\begin{aligned}
\min_{u_{i|k}} \quad & \|u_{\text{des}}(k) - u_{0|k}\| && \text{(31a)}\\
\text{s.t.} \quad & x_{i+1|k} = \mathrm{f}(x_{i|k}, u_{i|k}), && \text{(31b)}\\
& x_{0|k} = x(k), && \text{(31c)}\\
& x_{i|k} \in \mathcal{X}, \qquad \text{for } i = 0, ..., N-1, && \text{(31d)}\\
& x_{N|k} \in \mathcal{S}^t, && \text{(31e)}\\
& u_{i|k} \in \mathcal{U}, \qquad \text{for } i = 0, ..., N-1, && \text{(31f)}
\end{aligned}$$

where $i|k$ denotes planned states and inputs computed at time step k predicted i time steps into the future that satisfy the dynamic constraint (31b). An illustration of this planned sequence of states can be seen in Figure 4. The remaining constraints (31c)-(31f) ensure that backup plans lead the system into a safe terminal controlled invariant set $\mathcal{S}^t$ ((31e) is referred to as the terminal constraint in model predictive control [13]) while satisfying the state and input constraints. The following assumption on the terminal safe set $\mathcal{S}^t$ ensures this optimization problem yields a safe input:

Assumption 1 (Terminal Control Invariant Set)
Consider the system (30). There exists a terminal set $\mathcal{S}^t \subseteq \mathcal{X}$ such that for all $x \in \mathcal{S}^t$, there exists an input $u \in \mathcal{U}$ such that $\mathrm{f}(x, u) \in \mathcal{S}^t$.

Assumption 1 establishes that the terminal set $\mathcal{S}^t$ is a discrete-time control invariant set, similar to the continuous-time version in Definition 2 and thereby ensures safety for all times. A trivial choice for $\mathcal{S}^t$ is any equilibrium point $x_e = \mathrm{f}(x_e, u_e)$ satisfying $x_e \in \mathcal{X}$ and $u_e \in \mathcal{U}$. Methods for computing less restrictive terminal sets satisfying Assumption 1 can be obtained through discrete-time control barrier function design [18], [96] or can be computed as invariant sets under a local control law [13, Section 2.5.3.2], [97]. We provide a corresponding design procedure in the next section.

The resulting PSF for the discrete-time system (30) is then given by $\kappa_F(x(k), u_{\text{des}}(k)) = u^*_{0|k}$ with $u^*_{0|k}$ being the first element of the optimal backup control sequence obtained from (31). The formal closed-loop safety guarantee under application of $u(k) = \kappa_F(x(k), u_{\text{des}}(k))$ follows from an induction argument. In particular, assume that (31) was feasible at time $k-1$ with the corresponding optimal input sequence $\{u^*_{i|k-1}\}$. Under application of $u(k-1) = \kappa_F(x(k-1), u_{\text{des}}(k-1)) = u^*_{0|k-1}$, the system evolves to the state $x(k) = x^*_{1|k-1}$. Because the terminal set is a control invariant set, we can construct a feasible candidate sequence at time step k given by $\{u^*_{1|k-1}, u^*_{2|k-1}, .., u^*_{N-1|k-1}, \bar{u}\}$ with $\bar{u} \in \mathcal{U}$ such that $\mathrm{f}(x^*_{N-1|k-1}, \bar{u}) \in \mathcal{S}^t$, thereby satisfying all constraints in (31). By induction, we may conclude feasibility of (31), and consequently, satisfaction of state and input constraints due to (31d) and (31f), if (31) is initially feasible at $k = 0$. This result also implies that the set of feasible initial conditions

$$\mathcal{S}^{\text{PSF}}_N = \{x(k) \in \mathbb{R}^{n_x} | (31\text{b}) - (31\text{f})\}, \tag{32}$$

implicitly defines a control invariant set.

Similar to (29), the objective (31a) implements the desired safety filter property, that is, it minimizes the deviation between the first element of the input backup sequence $u_{0|k}$ and the desired input $u_{\text{des}}(k)$, such that $u_{0|k} = u_{\text{des}}(k)$ if $u_{\text{des}}(k)$ is safe. Conceptually, the PSF design can further be improved in terms of approximating the desired safety filter formulation (4) by adding future deviations $\sum_{i=0}^{N-1} \|u_{i|k} - u_{\text{des}}(k+i)\|$ to the objective (31a) if u_{des} is a known open-loop signal or policy. Furthermore, if a task-specific performance metric $\ell(x, u)$ is available, replacing (31a) with $\sum_{i=0}^{N-1} \ell(x_{i|k}, u_{i|k})$ recovers standard model predictive control formulations and directly approximates the underlying optimal control problem [13].

While PSFs provide a flexible framework for approximately optimal safety filtering and approximate optimal control, the central challenge is to solve (31) reliably in real-time. This is addressed theoretically and through software tools [13, Section 8] and is a central part of ongoing model predictive control research.

Nominal Terminal Invariant Set Design (Assumption 1):
We now detail a procedure for designing a terminal invariant set $\mathcal{S}^t$ according to Assumption 1 following [97]. Assume that the dynamics (1) are twice continuously differentiable and that there exists an equilibrium point at the origin such that $\mathrm{f}(0, 0) = 0$, with $0 \in \text{int}(\mathcal{X})$ and $0 \in \text{int}(\mathcal{U})$. We consider polytopic state and input constraints $\mathcal{X} = \{x \in \mathbb{R}^{n_x} | A_x x \leq b_x\}$ and $\mathcal{U} = \{u \in \mathbb{R}^{n_u} | A_u u \leq b_u\}$ and parameterize the terminal invariant set as an ellipsoidal set $\mathcal{S}^t_\gamma = \{x \in \mathbb{R}^{n_x} | \gamma - x^\top P x \geq 0\}$ with $\gamma \in (0, 1]$. The design procedure considers the linearization of (30) around the origin, denoted by $A = (\partial/\partial x)\mathrm{f}(x)|_{(0,0)}$, $B = (\partial/\partial u)\mathrm{f}(x, u)|_{(0,0)}$, and higher-order error terms $r(x, u) = \mathrm{f}(x, u) - Ax - Bu$. Assume that $K \in \mathbb{R}^{m \times n}$ renders $A - BK$ Schur stable. The forward invariance of $\mathcal{S}^t_\gamma$ under $u = -Kx$ can be imposed through the sufficient Lyapunov condition

$$(A_K x + r_K(x))^\top P (A_k + r_K(x)) - x^\top P x \leq 0, \tag{33}$$

where $A_K = (A - BK)$ and $r_K(x) = \mathrm{f}(x, -Kx) - A_K x$. Splitting the linear and nonlinear terms in (33) yields

$$x^T (A_K^\top P A_K - P) x + r_K(x)^\top P r_K(x) + 2x^\top A_K^\top P r_K(x) \leq 0. \tag{34}$$

Since the term $r_K(x)$ is higher-order, for a value $c \in (0, 1)$ we have that

$$\underbrace{r_K(x)^\top P r_K(x) + 2x^\top A_K^\top P r_K(x) - c x^\top P x}_{=R(x)} \leq 0, \tag{35}$$

locally around the origin. This allows the separation of the requirement (34) into the requirement $R(x) \leq 0$ and

the matrix inequality requirement

$$A_K^\top P A_K - (1-c)P \preceq 0. \tag{36}$$

We first compute P satisfying (36), $\mathcal{S}_1^t \subset \mathcal{X}$, and $K\mathcal{S}_1^t \subset \mathcal{U}$. Because $R(x)$ satisfies (35) locally about the origin, we then ensure (34) is met for the nonlinear system by iteratively reducing γ from a value of 1 until $\max_{x \in \mathcal{S}_\gamma^t} R(x) \leq 0$.

The search for a large terminal safe set $\mathcal{S}^t$ based on the linearized system is typically formulated as a convex optimization problem [98, Section 2.4.1], [97]. This design procedure can be applied on high-dimensional systems [99] and is demonstrated in the "Safety Filter Design Example" sidebar. It should be noted that the permissiveness of a PSF is typically governed by the size of the implicit safe set (32) and therefore depends on the size of the terminal safe set $\mathcal{S}^t$. As a result, various research directions have investigated efficient ways to extend terminal safe sets online using previous solutions of (31) [100], [101].

Practical Considerations

A practical challenge when implementing a PSF arises if disturbances drive the plant into a state for which the problem (31) is infeasible and no safe control input can be computed. A systematic method for dealing with infeasibility is to "soften" the constraints by including slack variables into the problem, as commonly done in model predictive control [102]. For instance, when the state and terminal constraints can be described by $\mathcal{X} = \{x \in \mathbb{R}^{n_x} | a^{\mathcal{X}}(x) \leq 0\}$ and $\mathcal{S}^t = \{x \in \mathbb{R}^{n_x} | a^{\mathcal{S}^t}(x) \leq 0\}$ for some functions $a^{\mathcal{X}}, a^{\mathcal{S}^t}$ respectively, the soft constrained reformulation of the PSF problem (31) is

$$\begin{aligned} \min_{u_{i|k}, \xi_{i|k}} \quad & \|u_{\text{des}}(k) - u_{0|k}\| + \sum_{i=0}^{N} l_\xi(\xi_{i|k}) \\ \text{s.t.} \quad & \text{(31b),(31c),(31f)}, \\ & \xi_{i|k} \geq 0, && \text{for } i = 0, ..., N, \\ & a^{\mathcal{X}}(x_{i|k}) \leq \xi_{i|k}, && \text{for } i = 0, ..., N-1, \\ & a^{\mathcal{S}^t}(x_{N|k}) \leq \xi_{N|k}. \end{aligned} \tag{37}$$

The non-negative slack variables $\{\xi_{i|k}\}$ ensure feasibility for any $x(k)$ and any input sequence $u_{i|k} \in \mathcal{U}$. The corresponding penalty function l_ξ can, for example, be selected as $l_\xi(\xi) = \|\xi\|^2 + \rho_\xi \|\xi\|$, where ρ_ξ is a positive constant. The goal is to select ρ_ξ large enough such that the second term in (37) admits an exact penalty function, implying that the slack variables are only non-zero if constraint satisfaction of the corresponding constraints is not possible. If the original, hard-constrained problem (31) is feasible, the soft-constrained problem should produce the same control input [102]. It should be noted that the slack variables are, however, not guaranteed to vanish in closed-loop, that is the system may not return to the implicit safe set defined by (31). Current research efforts in model predictive control [103], [104] and predictive safety filters [19] investigate such cases, for example, by connecting PSF and CBF theory [19], see also the discussion below.

Discussion

In this section we provide a brief overview of the relationship between HJ reachability, CBFs, and PSFs, with a focus on recent works at the intersection of the approaches.

Hamilton-Jacobi Reachability + Control Barrier Functions

Both the methods of HJ reachability and CBFs are built on determining an explicit representation of a control invariant set, typically through the superlevel sets of a continuous scalar function. This similarity leads to connections between the two approaches, both theoretically and in practical behavior. Moreover, the combination of these approaches is complementary since HJ reachability can increase the size of a control invariant set used in a safety filter, while CBFs provide a succinct approach for smoothly filtering a desired input signal.

The use of viability kernels in the construction of CBFs was first explored in [105], in which smooth polynomial approximations of the viability kernel were constructed via sums-of-squares programming and used as a CBF. This type of numerical approach enabled for a large control invariant set while preserving the smoothness properties needed by CBFs. The work in [15] conducts a comparative study of the control invariant sets found using HJ reachability and backup CBF methods (discussed more below in the paragraph on CBFs and PSFs). This work finds that given an adequately designed backup controller and backup set, the control invariant sets found with backup CBF approaches closely approximate the maximal control invariant sets found through HJ reachability.

Other recent work has looked at how elements from CBF-based safety filters can be directly incorporated into Hamilton-Jacobi reachability computations. The work in [16] integrates the comparison function seen in CBF-based safety filters into the HJ-VI (23) that is solved numerically, allowing for the synthesis of CBFs through the tool sets typically used in HJ reachability. In this new reachability formulation, the minimum norm safety filter (29) based on the resulting value function is verified to be the optimal policy of the value function. This allows the reachability community to expand their choice of the safety filters from the primary switching safety filter (22) to those in the CBF community which have better practical behaviors. The work in [106] integrates with this previous work by making use of the ability to warm-start the process of numerically solving the HJ-VI (23) to use dynamic programming to iteratively update a CBF candidate until it converges to a valid CBF. Though these approaches benefit from the strengths of both HJ reachability and CBFs, their numerical approach still faces challenges with high-dimensional systems. Other recent work has sought

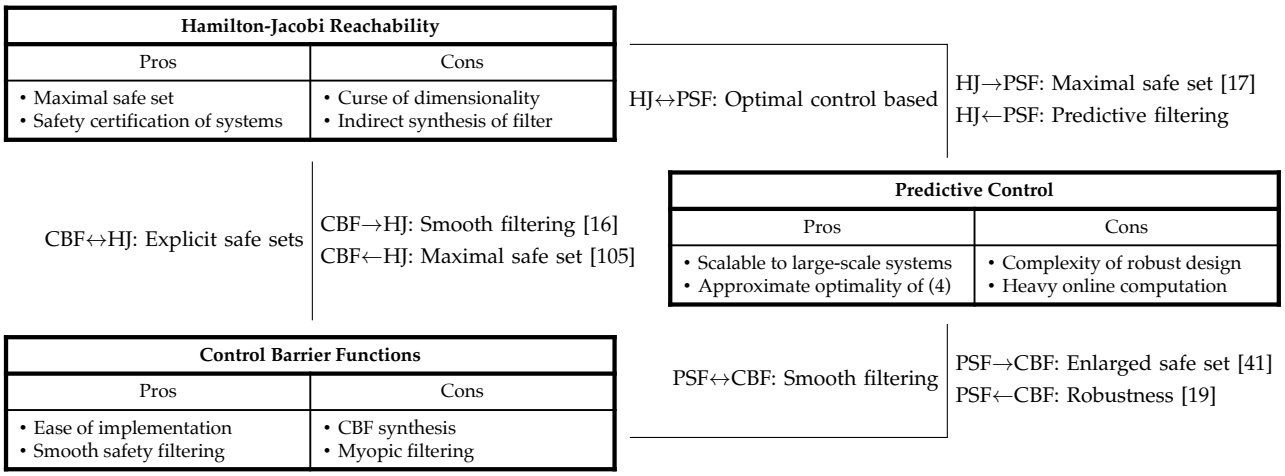


FIGURE 5 Key advantages and drawbacks of Hamilton-Jacobi reachability, control barrier functions, and predictive safety filters. The relationship between them is outlined (↔) and how techniques enhance each other, for example, CBF techniques can be used to improve HJ safety filters (CBF→HJ).

to alleviate these challenges by using learning to construct CBFs [107], [108].

Beyond finding efficient approaches for tackling the curse of dimensionality in this context, an open research direction at this intersection focuses on rigorously studying the regularity properties of CBFs constructed through reachability frameworks. Such an effort would rigorously codify the regularity properties achieved by weak viscosity solutions to the HJ-VI, and develop similarly rigorous results connecting the resulting CBFs and safety in the face of these regularity limitations, similarly to those in [109].

Hamilton-Jacobi Reachability + Predictive Safety Filters

While both HJ Reachability and PSFs aim to ensure safety through an optimal control problem formulation, there are differences in their respective problem structures and corresponding algorithms. First, HJ reachability incorporates safety constraints through an appropriate value function (15), whereas PSFs consider them as part of a constrained optimization problem (31d). As a result, HJ reachability-based safety filters decouple safe set synthesis and filter design, while PSFs implicitly capture a safe set and filter inputs through a single optimization problem.

Second, HJ reachability uses the machinery of dynamic programming [110] to find an optimal solution offline for all states. Typically, the value function (15) is explicitly computed as the solution of a HJ partial differential equation, which can be done for a class of optimal control problems [26] including both reachability formulations and state-constrained general-cost problems [111]. The computation of the value function globally for all states faces the curse of dimensionality. In return, it explicitly characterizes the maximal control invariant set in $\mathcal{X}$ before deploying the controller, allowing for controller verification when employed on safety-critical systems [9].

In contrast, PSFs leverage online optimization to approximately solve a state-constrained optimal control problem (31) using only the current state and a receding horizon principle [13]. The approximate nature of solving the state-constrained optimal control problem originates from two sources. First, the predictive time horizon is only finite and typically much shorter than the actual horizon over which safety is required. In order to still guarantee safety over the entire horizon, PSFs employ the terminal safe set constraint (31e) at the end of the horizon, restricting the degrees of freedom of the filter.

Second, a PSF solves the open-loop optimal control problem (31) in a receding horizon fashion, that is, at every time step based on the current state. Solving the optimal control problem (31) online avoids requiring an explicit pre-computation of an optimal control policy, thereby circumventing the curse of dimensionality. Instead, PSFs require efficient nonlinear programming solvers working in real-time with significant system processing power. If sufficient computation power is available online, PSFs can provide a near-optimal safety filter even for high-dimensional systems. Additionally, evaluating whether the system will be safe given an initial state can only be verified by evaluating feasibility of the optimization problem (31). Thus, an explicit representation of the safe set is not available, but rather an implicit representation defined by feasibility of the optimization problem (31).

Despite these differences between the two approaches, there are similarities between the approaches that suggest the potential for integrating them. In particular, the implicit safe set defined by a PSF using a sufficiently long planning

horizon coincides with the explicit safe set (viability kernel) from HJ reachability. This effect is demonstrated in "Safety Filter Design Example". Finally, recent approaches are exploring various ways of exploiting the benefits of both methods, see for example, [17], where condition (21) from HJ reachability is incorporated as a constraint in a predictive controller.

Control Barrier Functions + Predictive Safety Filters

Control barrier functions and predictive safety filter techniques naturally complement each other in a way that reduces the weakness of each individual method. The predictive horizon present in predictive safety filters can help to reduce poor closed-loop behavior induced by the myopic nature of a CBF-based safety filter. This improvement comes with the burden of solving a nonlinear optimization problem in real-time, which substantially increases the complexity of the safety filter design and implementation compared to CBF-based filters. Furthermore, predictive safety filters do not provide intrinsic robustness properties, which often result in rather complicated design procedures to ensure safety despite disturbances.

This complementary relationship has yielded several recent results integrating the two methods. Integrating CBF constraints directly into the optimization problem specifying the predictive filter, either as a instantaneous derivative condition [88], or a decrement condition [18], [112]–[114], leads to the dynamic "braking" typical of CBFs and often yields robust behavior. In addition, the use of CBFs as a terminal constraint can formally render the sum of slack variables in the predictive safety filter problem (37) into a 'predictive' CBF [19]. Further approaches include multirate architectures, in which a high-level predictive controller provides a desired input signal that is filtered using a CBF-based safety filter [20], [88], [115]. These approaches allow for the complex nonlinear predictive optimization problem to be solved at slower frequencies since the CBF-based filter keeps the system close to the planned trajectory at a high frequency. Other approaches have introduced predictive elements to consider safety along solution trajectories [116], or used predictive elements for trajectory tracking and CBFs for obstacle avoidance [117].

The thread of work in [41], [118], [119] focuses on the notion of backup set methods using CBF-based safety filters. This approach shares conceptual elements with predictive safety filters by using a backup set which can be kept forward invariant with a backup controller to implicitly define a larger control invariant set. The backup set methods consider a predictive horizon over which a CBF constraint must be enforced, ensuring the system can always reach the backup set. Structural differences between these backup set approaches and predictive filters often lead to different approximations for tractably handling the use of a predictive horizon, suggesting a distinction between the two methods.

DATA-DRIVEN SAFETY FILTERS

The safety filter techniques developed in the first half of this article were presented assuming perfect knowledge of the system dynamics (1). However, in most practical settings, high-fidelity system models are difficult to construct and systems are subject to external disturbances, which can lead to loss of safety guarantees. This challenge has been a topic of significant research interest from the perspective of data-driven control, in which empirical information about the unknown system is integrated into various elements of the safety filter synthesis process. We now present this problem setting, and a selection of data-driven results related to HJ reachability, CBFs, and PSFs.

Consider the nonlinear control system

$$\dot{x} = f_{\text{true}}(x, u), \tag{38}$$

where $x \in \mathbb{R}^{n_x}$ is the system state, $u \in \mathbb{R}^{n_u}$ is the control input, and $f_{\text{true}} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$, which for simplicity we assume to be continuously differentiable in its arguments. For many systems in engineering, first principles, such as Lagrangian mechanics or the laws of thermodynamics, allow for the derivation of simplified model structures to construct the function f in (1). In many real-world applications, generating a sufficiently accurate first-principles model can, however, require significant engineering effort, or may not even be possible, for instance in biological or chemical applications, leading to a discrepancy between the model f and the actual dynamics of the system, given by f_{true}.

We note that this mathematical formulation can describe parametric uncertainty. Even for systems for which the structure of f accurately characterizes f_{true}, there may be errors between parameters of the model and parameters of the actual system. For instance, different cars of the same vehicle type typically have the same model structure, but may differ in the model parameters due to manufacturing tolerances, wear of components, or replacement parts such as different tire rubbers. Manually identifying parameters through laboratory testing is often difficult and costly, and designs that are robust to large parameter uncertainties are often conservative. Various data-driven techniques specialized for addressing safety in the face of parametric uncertainty have been proposed including adaptive control [120], [121] and Bayesian estimation [122], [123].

This section discusses how to leverage data-driven techniques to improve a model obtained from first principles, given by (1), to more accurately reflect (38) and presents selected techniques for using these concepts in the context of safety filters. To this end, consider a sequence of measured states, inputs, and state time derivatives

$$D = \{(x_k, u_k, \dot{x}_k)\}_1^{n_D} \triangleq \{(x(kT_s), u(kT_s), \dot{x}(kT_s)\}_1^{n_D}, \tag{39}$$

at sampling time steps kT_s. We note that the state and state derivative measurements are subject to noise, which is typically assumed to be contained in a known set or modeled as an independent and identically distributed random variable. We also note that data of the form (39) can equally handle episodic measurements, including multiple resets of the system state, enabling an iterative model refinement. While this article mainly focuses on learning the system dynamics model (38), such a data set could be used for other elements of the safety filter process such as learning control invariant sets [94], [107], [108].

Model Uncertainty Decomposition

System modeling by domain experts using physical principles is typically the first step of safety filter design and yields an imperfect nominal model f as in (1). Using this model, we can rewrite the actual system dynamics (38) as

$$\dot{x} = f(x,u) + \underbrace{f_{\text{true}}(x,u) - f(x,u)}_{=e^n(x,u)}, \tag{40}$$

where the function $e^n : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ captures all errors between the system model and the actual system. While assumptions on the uncertainty of the system may be used to construct a state and input dependent set $\mathcal{E}^n(x,u)$ such that $e^n(x,u) \in \mathcal{E}^n(x,u)$, this set often significantly over approximates the model error, yielding robust designs that are excessively conservative. Data-driven techniques tackle this challenge by reducing the model error e^n to a smaller learning error, $e^l : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ using a learning-based correction term $\mathrm{f}^l : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$, that is

$$\dot{x} = f(x,u) + f^l(x,u) + \underbrace{e^n(x,u) - f^l(x,u)}_{=e^l(x,u)}. \tag{41}$$

Learning f^l from the data (39) to mitigate e^l can be posed as a classical regression problem, which can be divided into parametric approaches with a fixed number of parameters independent of the number of measurements n_D, and non-parametric approaches that have a variable number of parameters that grows with n_D. Parametric approaches are particularly suitable when the structure f_{true} in (38) is well understood and when fast predictions are required. In contrast, non-parametric approaches can compensate for both parametric and structural uncertainty but can be computationally more expensive to use for prediction. In addition to reducing the model error e^n, some learning techniques provide a bound on the residual learning error e^l in the form of a state and input dependent set, such that $e^l(x,u) \in \mathcal{E}^l(x,u)$. The availability of such a set and its structure allows for a distinction of data-driven safety filter approaches as follows [23], [60], [95, Section II].

Deterministic Models

The first class of approaches consider the integration of deterministic data-driven models into safety filter design, without any specific quantification of the residual learning error e^l. Such approaches can provide good predictive performance and are commonly employed in practical applications. Examples include parametric models with simple least-squares regression or (recurrent) artificial neural networks and non-parametric techniques based on k-nearest-neighbors techniques [124]. Since these approaches typically do not provide an explicit bound on the residual learning error, safety according to (2) is practically achieved using tightened constraints of the form $\alpha\mathcal{X}$, $\alpha\mathcal{U}$ with $\alpha \in (0,1)$ in safety filter design. The resulting safety margin $(1-\alpha)$ is then hand-tuned to achieve constraint satisfaction. Examples include the use of deterministic models with CBFs [24], [47], [50] (see the examples "Data-Driven Control Barrier Function Safety Filter Applications") and soft constrained PSFs [125] (see the miniature race car example in "Predictive Safety Filter Applications: Experimental Race Cars and Simulated Quadrotors").

Robust Models

The second class of approaches directly incorporate an explicit bound on the residual learning error e^l into the safety filter design, yielding robust safety filters. As previously noted, certain data-driven models bound the residual learning error through a state and input dependent set, such that $e^l(x,u) \in \mathcal{E}^l(x,u)$. Safety filter design is done such that the system in (41) is safe for all possible residual learning error values in the set $\mathcal{E}(x,u)$. Error quantification for parametric methods often uses regularity properties of a class of parametric learning models, such as the use of spectral normalization and Lipschitz constants with recurrent neural networks [126], [127]. Non-parametric methods often use assumptions on the actual dynamics of the system (such as Lipschitz continuity) in conjunction with data to synthesize robust safety filters [128], [23, Section 3.1.2]. Such approaches have been taken using HJ reachability through a differential game formulation [129], using CBFs through robust optimization [51], [130], and using PSFs by determining an appropriate constraint tightening mechanism [56].

Probabilistic Models

The preceding robust approaches guarantee safety of a system, but they can often be unnecessarily conservative because they must capture all possible residual learning errors. Moreover, they tend to neglect the fact that the measurements composing D are noisy, and that resulting guarantees on learning accuracy are inherently probabilistic. The third class of approaches uses distributional information about the residual learning error in the safety filter design process, permitting practical designs that can balance the need for safety with strong performance. The corresponding data-driven models typically provide a data-driven description of the residual learning error e^l in the form of a probability distribution, $p(e^l|D)$. An

overview of parametric and non-parametric probabilistic regression techniques often used in control can be found in [95, Section II], [23], [60] and references therein. A common learning technique to estimate the model error e^n in the context of safety is based on Gaussian process regression [59], [131]–[133].

Though it may be possible to construct probabilistic descriptions of structural uncertainties, parametric uncertainties, and external disturbances, it can be challenging to translate these descriptions into a safety filter formulation. A common simplification is to consider overall safety guarantees from a probabilistic perspective by considering *robustness at a certain probability level* [23, Section 3.2], [58], [59], [133]–[135]. To this end, we construct a state and input dependent uncertainty set $\mathcal{E}^l(x,u)$ based on available data D (39) similar to the robust case, which is, however, only valid in probability, such that

$$\Pr\left(\underbrace{e^l(x,u) \in \mathcal{E}^l(x,u) \text{ for all } x \in \mathcal{X} \text{ and } u \in \mathcal{U}}_{\star}\right) \geq p_s, \tag{42}$$

at some desired probability level p_s. We note that compared to the robust approach, we do not require $e^l(x,u) \in \mathcal{E}^l(x,u)$ with certainty, but rather only at the specified probability level p_s. In practice this can serve to eliminate the need to address extremely unlikely scenarios that lead to conservative behavior of robust approaches. Recalling the safety specification given in (2), any robust design that is safe for all possible residual learning error values in $\mathcal{E}^l(x,u)$ yields that $\Pr((2) \mid \star) = 1$ where $\star$ is defined in (42), implying

$$\Pr((2)) \geq \Pr((2), \star) = \Pr((2) \mid \star) \Pr(\star) \geq p_s, \tag{43}$$

such that

$$\Pr\left(x(t) \in \mathcal{X} \text{ and } u(t) \in \mathcal{U} \text{ for all } t \in \mathbb{R}_{\geq 0}\right) \geq p_s. \tag{44}$$

The relation between the probabilistic error bound (42) and constraint satisfaction in probability (44) provide an intuitive way for trading-off safety and permissiveness, since lower probability levels p_s typically lead to a smaller learning error bound $\mathcal{E}^l$ and less conservative robust safety filter designs. The type of probabilistic condition in (43) has been utilized in the design of probabilistic safety filters through HJ reachability [22], [45], CBFs [49], [52], [135], and PSFs [55], [58].

In theory, any of the advanced safety filter techniques presented can be combined with the preceding model classes. In practice, some safety filter techniques naturally lend themselves to being used with a specific type of model class, as we highlight in the following sections. We also note that systems are often subject to unmodeled external disturbances caused by environmental perturbations, such as wind acting on an airplane or changing road friction coefficients for a ground vehicle. In contrast to uncertainty in the model, these disturbances often do not have an underlying structure that can be discovered by data. Rather, data is often used to quantify the magnitude of disturbances, which is then used for a robust design. For simplicity, the following formulation is presented in the absence of such disturbances, but we note that the following methods for developing safety filters that are robust to learning error can be used (and in fact, originated) for robustness to disturbances.

Data-driven Hamilton-Jacobi Reachability

Due to the inherent separation of safety from performance in HJ reachability, reachability-based safety filter designs can be used together with any type of controller emitting the desired control input signal. In particular, reachability-based safety filters are suitable for filtering learning-enabled systems like autonomous vehicles throughout the process of training learning-based components in the system. We describe such a HJ reachability-based safety framework for uncertain systems as proposed in [22]. Several extensions and variants of this framework have been proposed to demonstrate the applicability of the framework to high-dimensional systems [45], [136]. We highlight simulation and experimental results utilizing this framework in "Reachability-based Safe Learning Framework: Experimental Results" to demonstrate the effectiveness of reachability-based frameworks in real-world applications.

Hamilton-Jacobi Reachability With Learning Error

First, we describe the HJ reachability analysis that is extended to account for learning errors by using a differential game based formulation [137], resulting in a characterization of the maximal control invariant set and an associated optimal safe policy that are robust to bounded learning error. For the sake of simplified exposition, consider a setting where the model error e^n in (40) does not depend on the input u. Consequently, a learning model $f^l : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$, a learning error $e^l : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$, and a pointwise set $\mathcal{E}^l(x) \subset \mathbb{R}^{n_x}$ such that $e^l(x) \in \mathcal{E}^l(x)$ for all $x \in \mathcal{X}$ can be considered. As the value of the learning error is unknown, it is desirable for a safety filter design to be robust to all possible learning errors permitted by the pointwise set $\mathcal{E}^l(x)$. To this end, consider the dynamics

$$\dot{x} = f(x,u) + f^l(x) + d, \tag{45}$$

where $d \in \mathcal{E}^l(x)$ is a disturbance term which captures the possible effect of $e^l(x)$ on the system dynamics. To construct the maximal control invariant set contained in $\mathcal{X}$ in the setting with disturbances, we consider a cost functional $J_d : \mathbb{R}^{n_x} \times \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U}) \times \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathbb{R}^{n_x}) \to \mathbb{R}$ similar to (14)

$$J_d(x_0, u(\cdot), d(\cdot)) = \inf_{t \in \mathbb{R}_{\geq 0}} -s_{\mathcal{X}}(x(t)), \tag{46}$$

where $x(\cdot)$ is the solution to (45) with the initial condition x_0, an input signal $u(\cdot)$, and a disturbance signal $d(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathbb{R}^{n_x})$.

The set of non-anticipative disturbance strategies, denoted by $\mathcal{D}$ is defined as the set of all functionals $\beta : \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U}) \to \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathbb{R}^{n_x})$ that satisfy

$$\beta[u](t) \in \mathcal{E}^l(x(t)) \text{ for all } u(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U}) \text{ and } t \in \mathbb{R}_{\geq 0} \tag{47}$$

and

$$\begin{aligned} &\beta[u_1](t) = \beta[u_2](t) \text{ for almost all } t \in \mathbb{R}_{\geq 0}, \\ &\text{for all } u_1(\cdot), u_2(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U}) \text{ s.t.} \\ &u_1(t) = u_2(t) \text{ for almost all } t \in \mathbb{R}_{\geq 0}. \end{aligned} \tag{48}$$

Intuitively, the disturbance signal $d(\cdot)$ resulting from the strategy β should satisfy the learning error bound $d(t) \in \mathcal{E}^l(x(t))$ for all time, and the non-anticipative restriction prohibits $d(\cdot)$ from depending on the future information of the control signal $u(\cdot)$ [137].

A value function $V_d : \mathbb{R}^{n_x} \to \mathbb{R}$ that accounts for disturbances can be constructed similarly to (15) through a zero-sum differential game

$$V_d(x_0) = \inf_{\beta[u] \in \mathcal{D}} \sup_{u(\cdot) \in \mathcal{PC}(\mathbb{R}_{\geq 0}, \mathcal{U})} J(x_0, u(\cdot), \beta[u](\cdot)). \tag{49}$$

Computation of V_d can be done by solving the HJ-VI [36]:

$$\begin{aligned} 0 = \min\{ &- s_{\mathcal{X}}(x) - V_d(x), \\ &\max_{u \in \mathcal{U}} \min_{d \in \mathcal{E}^l(x)} \nabla V_d(x)(f(x,u) + d)\}, \end{aligned} \tag{50}$$

which has a viscosity solution that characterizes V_d. Similarly to Theorem 2, the value function V_d (49) can be used to characterize control invariant sets in $\mathcal{X}$ that are robust to learning errors. More precisely, for any $\epsilon \in \mathbb{R}_{\geq 0}$, the set $\mathcal{S}_\epsilon = \{x \in \mathcal{X} \mid V_d(x) \geq \epsilon\}$ is a control invariant set that is robust to learning errors, and $\mathcal{S}_0$ characterizes the maximal control invariant set contained in $\mathcal{X}$ that is robust to learning errors [22]. Finally, the robust optimal safe policy $\kappa^*_{V_d} : \mathbb{R}^{n_x} \to \mathcal{U}$ can be constructed as

$$\kappa^*_{V_d}(x) = \operatorname*{argmax}_{u \in \mathcal{U}} \min_{d \in \mathcal{E}^l(x)} \nabla V_d(x)(f(x,u) + d), \tag{51}$$

which ensures the set $\mathcal{S}_\epsilon$ is forward invariant in the presence of learning errors. Compared to the optimal safe policy defined in (20), this controller introduces the term $\min_{d \in \mathcal{E}^l(x)}$, which considers the worst-case uncertainty d at the current state when synthesizing the safe control input.

Reachability-based Safe Learning Framework

The safe set $\mathcal{S}_\epsilon$ and the safe policy $\kappa^*_{V_d}(x)$ in the above formulation can be overly conservative when the set $\mathcal{E}^l(x)$ is overestimated. Moreover, under-approximating the set $\mathcal{E}^l(x)$ in the construction of the value function V_d can lead to failure of the system to remain safe in the presence of learning errors. This motivates incorporating the data-driven techniques that accurately characterize $\mathcal{E}^l(x)$ into the differential game formulation. The framework in [22] employs Gaussian process (GP) regression [138], a class of probabilistic learning models. GP regression is further selected because it is compatible with Bayesian inference, allowing for confidence in a learned model to be determined as new data is acquired. It is worth noting that any robust or probabilistic data-driven models that provide accurate characterization of model uncertainty can function well in this reachability framework.

The objective of GP regression is to construct a learned model f^l that approximates the model error e^n with its mean prediction and capture the residual learning error e^l with its posterior variance. The input data is a sequence of measured states $\{x_k\}_{k=1}^{n_D}$ and the output data is noisy measurements of $e^n(x_k)$, taking the form $y_k = \hat{\dot{x}}_k - f(x_k, u_k)$, where $\hat{\dot{x}}_k$ is the noisy estimate of the state derivative based on numerical differentiation. The posterior distribution of the GP regression in the j^{th} state dimension is a normal distribution $\mathcal{N}(\mu_j(x), \sigma_j(x))$, representing the estimated distribution of $e^n_j(x)$ at a state $x \in \mathbb{R}^{n_x}$. Thus, the learned model f^l is described by the vector $[\mu_1(x) \cdots \mu_{n_x}(x)]^T \in \mathbb{R}^{n_x}$ consisting of the mean predictions in each state dimension, and the set $\mathcal{E}^l(x)$ is constructed as the hyperbox $\mathcal{E}^l(x) = \prod_{j=1}^{n_x} \left[-z\sigma^j(x), z\sigma^j(x)\right]$ by taking the confidence intervals of the posterior distribution in each state dimension multiplied by the quantile $z \in \mathbb{R}_{>0}$.

Given a choice of the quantile z, a probabilistic relationship is established between the actual learning error, e^l, and the estimate of possible learning errors captured by the set $\mathcal{E}^l(x)$ as in (42) [22, Section C]. While a conservative estimate of the possible learning errors $\mathcal{E}^l(x)$ may satisfy (42) with a high probability p_s, reducing the conservativeness of an estimate of the possible learning errors while maintaining (42) with a high probability can permit better performance. The following result on the differential game form of HJ reachability establishes a property of HJ reachability-based safety filter designs relating two estimates of possible learning errors [22, Proposition 5]:

Theorem 5

Consider two estimates of possible learning errors $\mathcal{E}^l_1(x), \mathcal{E}^l_2(x)$ such that for all $x \in \mathbb{R}^{n_x}$, $\mathcal{E}^l_2(x) \subseteq \mathcal{E}^l_1(x)$. If a set $\mathcal{S} \subset \mathbb{R}^{n_x}$ is a control invariant set that is robust for all possible learning errors in $\mathcal{E}^l_1(x)$, then it is a control invariant set that is robust for all possible learning errors in $\mathcal{E}^l_2(x)$.

Theorem 5 serves as the central principle underlying the safe learning framework. When the safe learning framework is initiated, the learning model has little to no data and the estimate of possible learning errors $\mathcal{E}^l(x)$ is typically quite large. The resulting control invariant set constructed through HJ reachability that is robust to these learning errors is conservative and limits the performance of the system. As learning proceeds, data is incorporated into the learning model and the control invariant set may be updated to require robustness to smaller estimates

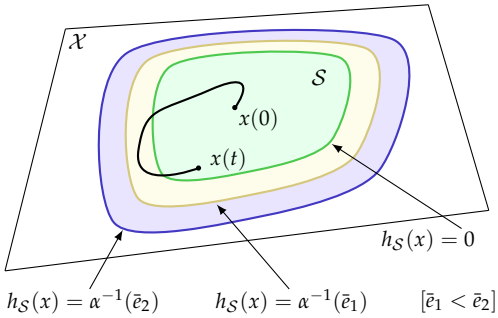


FIGURE 6 Schematic of input-to-state safety. In the presence of residual learning error, a controller that satisfies the CBF constraint using the learning model (53) may not render the set $\mathcal{S}$ forward invariant. Rather, a larger set that scales with the magnitude of the learning error is kept forward invariant, reflected by the two nested sets for the progressively larger learning error bounds $\bar{e}_1$ and $\bar{e}_2$.

$\mathcal{E}^l(x)$. Ideally, learning the smallest set of possible errors results in a control invariant set that is the maximal control invariant set in $\mathcal{X}$ that can be made robust to the presence of learning error.

Data-driven Control Barrier Functions

The use of data-driven techniques with control barrier functions has been an active area of research interest, with a wide range of a approaches including using models that are deterministic [24], [47], [48], [50], [139], robust [51], and probabilistic [46], [49], [52]–[54], [84], [135]. An underlying robustness property of CBF-based safety filter design known as input-to-state safety (ISSf) [39], [139] manifests in each of these approaches, which we now present in a general context.

Consider the control-affine model (24) with the introduction of a learning model f^l and a corresponding learning error

$$\dot{x} = f(x) + g(x)u + f^l(x,u) + e^l(x,u). \tag{52}$$

Let $\mathcal{S}$ be defined as the 0-superlevel set of a continuously differentiable function $h_{\mathcal{S}} : \mathbb{R}^{n_x} \to \mathbb{R}$, and suppose that using the learned model f^l, we design a safety filter $\kappa_{\mathcal{S}} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathcal{U}$ such that there exists an $\alpha \in \mathcal{K}^{\mathrm{e}}$ satisfying

$$\begin{aligned}\nabla h_{\mathcal{S}}(x)\big(f(x) + g(x)\kappa(x, u_{\mathrm{des}}(t)) \\ + f^l(x, \kappa(x, u_{\mathrm{des}}(t)))\big) \geq -\alpha(h_{\mathcal{S}}(x)),\end{aligned} \tag{53}$$

for all $x \in \mathbb{R}^{n_x}$ and $t \in \mathbb{R}_{\geq 0}$. This safety filter is designed to meet the original safety specification encoded by the barrier function $h_{\mathcal{S}}$ and the function α, but does so incorporating the learned model f^l. Let us further suppose that there exists an $\bar{e} \in \mathbb{R}_{\geq 0}$ such that

$$|\nabla h_{\mathcal{S}}(x) e^l(x, \kappa(x, u_{\mathrm{des}}(t))| \leq \bar{e}, \tag{54}$$

for all $x \in \mathbb{R}^{n_x}$ and $t \in \mathbb{R}_{\geq 0}$. This inequality implies that the effect of the residual learning error on the time derivative of the barrier function $h_{\mathcal{S}}$ is bounded by a constant $\bar{e}$.

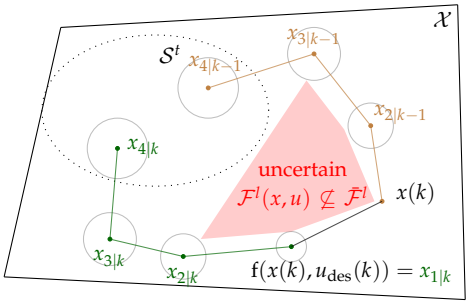


FIGURE 7 Learning-based enhanced predictive safety filter. Uncertain model regions (red) are avoided when planning backup trajectories. An additional safety margin (circles) allows for compensation of the remaining uncertainty during closed-loop operation.

Intuitively, this bound can be made smaller through more accurate learning models.

Combining these two properties, we have that

$$\dot{h}_{\mathcal{S}}(x,t) \geq -\alpha(h_{\mathcal{S}}(x)) - \bar{e}, \tag{55}$$

for all $x \in \mathbb{R}^{n_x}$ and $t \in \mathbb{R}_{\geq 0}$. Noting that $\alpha \in \mathcal{K}^{\mathrm{e}}$ implies it has an inverse $\alpha^{-1} \in \mathcal{K}^{\mathrm{e}}$, we have the implication that

$$h_{\mathcal{S}}(x) \leq \alpha^{-1}(-\bar{e}) \implies \dot{h}_{\mathcal{S}}(x,t) \geq 0. \tag{56}$$

This preceding implication states that time derivative of the barrier function $h_{\mathcal{S}}$ is non-negative on the boundary of the $\alpha^{-1}(\bar{e})$-superlevel set of $h_{\mathcal{S}}$

$$\mathcal{S}_{\bar{e}} = \{x \in \mathbb{R}^{n_x} \mid h_{\mathcal{S}}(x) \geq \alpha^{-1}(-\bar{e})\}, \tag{57}$$

and thus we can conclude via Nagumo's Theorem (we have $\nabla h_{\mathcal{S}}(x) \neq 0$ when $h_{\mathcal{S}}(x) < 0$ [140]) that $S_{\bar{e}}$ is forward invariant. This analysis highlights a fundamental robustness property of CBF-based safety filter designs since the set kept forward invariant does not increase dramatically with small amounts of residual learning error, but rather scales proportionally. Moreover, this expansion can be controlled by reducing residual learning error through more data and better learning models that serve to reduce $\bar{e}$. This notion of a safe set that scales with residual learning error is captured by the idea of ISSf [39]. We note that not only can ISSf describe the impact of model error in (40) (without introducing learning models), but provides insight into robust controller design that allows regulation over the growth of the forward invariant set [89], [140].

Data-driven Predictive Safety Filters

The close relation between the nominal predictive safety filter formulation (31) and common model predictive controllers (MPCs) using a terminal set [33] allows to take advantage of existing advances in the field of robust [13, Section 3], [34, Section 7] and learning-based model predictive control [23], [59], [95, Section 5]. While the focus in the case of PSF is to provide formal guarantees regarding constraint satisfaction, most of the underlying mechanisms applied originate from robust model predictive

control literature. Data-driven PSFs have been developed for linear robust (distributed) models [56], [57] and linear (distributed) stochastic models with unbounded process noise [58], [141, Remark 5]. The support of nonlinear system dynamics and exploration beyond available data has been enabled through leveraging probabilistic state- and input-dependent system models [55], [59]. While the precise details of each these methods vary, they all operate using the idea that instead of directly working with the original safety specifications along predictions (31d), the constraints are enforced with an additional safety margin. This margin is designed to compensate for prediction model errors and disturbances in closed-loop without violating the original safety constraints of the system. The rigorous computation of these safety margins is at the core of robust-, stochastic- and learning-based model predictive control methods. In the following, we specifically focus on the technique introduced in [55], which provides a computationally efficient way to combine PSFs with robust and probabilistic learning models.

Similar to the nominal PSF formulation and consistent with learning-based model predictive control literature [23], [95], we work with a discrete-time version of the learning-based model (41)

$$x(k+1) = \mathrm{f}(x(k), u(k)) + \mathrm{f}^l(x(k), u(k)) + \mathrm{e}^l(k), \tag{58}$$

where we use $\mathrm{e}^l(k)$ to denote $\mathrm{e}^l(x(k), u(k))$. The learning-based model in (58) and an uncertainty bound of the form (42) can be estimated with the previously discussed GP regression using measurements of the form $y^k = x_{k+1} - x_k + \epsilon_k$ with ϵ_k independent and identically distributed noise, see "Data-driven Hamilton-Jacobi Reachability". The central idea of the following approach is to restrict backup trajectories $\{x_{i|k}\}$, $\{u_{i|k}\}$ to high-confidence subsets of the state and input space by imposing

$$\mathcal{E}^l(x_{i|k}, u_{i|k}) \subseteq \bar{\mathcal{E}}^l, \quad \text{for } i = 0, .., N \tag{59}$$

along predictions, where $\bar{\mathcal{E}}^l \subset \mathbb{R}^{n_x}$ captures a tolerable amount of one-step prediction errors. This mechanism causes trajectories to avoid regions with low model confidence due to sparse data coverage, as seen in Figure 7. We note that (59) can be reformulated as a set of inequality constraints in the case of Gaussian processes or Gaussian linear regression, and becomes a convex constraint in the case of linear features [134, Section 4.1], [55, Section 5.1].

While various existing robust predictive control techniques can be used to obtain robustness in probability (43), we focus on a constraint tightening approach based on [142], [143]. The idea is to introduce increasing safety margins for all constraints along the prediction horizon, ensuring recursive feasibility and constraint satisfaction in closed-loop. In the case of polytopic state, input, terminal, and confident subspace constraints of the form $\{x \in \mathbb{R}^n | Ax \leq \mathbb{1}^{n_A}\}$ with $A \in \mathbb{R}^{n_A \times n}$ the tightening of the constraint sets is

$$\bar{\mathcal{X}}_i = \{x \in \mathbb{R}^{n_x} \mid A^x x \leq (1 - \epsilon_i)\mathbb{1}^{n_{A_x}}\}, \tag{60a}$$

$$\bar{\mathcal{U}}_i = \{u \in \mathbb{R}^{n_u} \mid A^u u \leq (1 - \epsilon_i)\mathbb{1}^{n_{A_u}}\}, \tag{60b}$$

$$\bar{\mathcal{E}}_i^l = \{x \in \mathbb{R}^{n_x} \mid A^{\mathcal{E}} x \leq (1 - \epsilon_i)\mathbb{1}^{n_{A_\mathcal{E}}}\}, \tag{60c}$$

with $\mathbb{1}^n$ denoting the vector of ones of dimension n and with a monotonically increasing tightening sequence ϵ_i satisfying $\epsilon_0 = 0$ and $\epsilon_{i+1} > \epsilon_i$. Integrating the learning-based model (58) and the tightened constraints (60) into the predictive safety filter problem (31) yields

$$\min_{u_{i|k}} \|u_{\mathrm{des}}(k) - u_{0|k}\| \tag{61a}$$

$$\text{s.t. } x_{i+1|k} = \mathrm{f}(x_{i|k}, u_{i|k}) + \mathrm{f}^l(x_{i|k}, u_{i|k}), \tag{61b}$$

$$x_{0|k} = x(k), \tag{61c}$$

$$x_{i|k} \in \bar{\mathcal{X}}_i, \qquad \text{for } i = 0, ..., N-1, \tag{61d}$$

$$x_{N|k} \in \mathcal{S}_N^t, \tag{61e}$$

$$u_{i|k} \in \bar{\mathcal{U}}_i, \qquad \text{for } i = 0, ..., N-1, \tag{61f}$$

$$\mathcal{E}^l(x_{i|k}, u_{i|k}) \subseteq \bar{\mathcal{E}}_i^l, \qquad \text{for } i = 0, ..., N-1. \tag{61g}$$

Similar to the nominal case, constraint satisfaction under application of $u(k) = u_{0|k}^*$ can be shown through recursive feasibility of (61) using the tightened constraints together with a robust terminal invariant $\mathcal{S}^t$ set:

Assumption 2 (Robust Terminal Control Invariant Set) Consider the system (58). There exists a polytopic terminal set $\mathcal{S}^t \subseteq \bar{\mathcal{X}}_N$ and a Lipschitz continuous terminal control law $\kappa^t : \mathcal{S}^t \to \mathbb{R}^{n_u}$ such that for all $x \in \mathcal{S}^t$ it holds that

1) $\mathcal{E}^l(x, \kappa^t(x)) \subseteq \bar{\mathcal{E}}_N^l$,
2) $\kappa^t(x) \in \bar{\mathcal{U}}_N$, and
3) $\mathrm{f}(x, \kappa^t(x)) + \mathrm{f}^l(x, \kappa^t(x)) + \mathrm{e} \in \mathcal{S}^t$, for all $\mathrm{e} \in \bar{\mathcal{E}}_N^l$.

If $0 \in \mathrm{int}(\mathcal{X} \times \mathcal{U})$ and the linearization of (58) at the origin is stabilizable, then a sufficiently small learning error $\mathcal{E}^l(x, u)$ allows the construction of a terminal set $\mathcal{S}^t$ and terminal controller κ^t satisfying Assumption 2 [13, 3.3.2]. Compared with the nominal PSF terminal set assumption (Assumption 1), Assumption 2 ensures forward invariance of a polytopic terminal set $\mathcal{S}^t$ for all possible learning errors and requires κ^t to be Lipschitz continuous. Combining Assumption 2 with continuity assumptions on $\mathcal{E}^l(x, u)$ and the dynamics model (58) enables a characterization of a data-driven PSF as follows [55, Theorem 4.6]:

Theorem 6

Let Assumption 2 hold and assume that (58) and the corresponding uncertainty bound $\mathcal{E}^l(x, u)$ satisfying (42) are Lipschitz continuous mappings with Lipschitz constants $L_f, L_\mathcal{E}$. Consider (60) with constraint tightening sequence

$$\epsilon_i = \epsilon \frac{1 - \sqrt{L_f}^i}{1 - \sqrt{L_f}} \text{ for some } \epsilon > 0, \tag{62}$$

and allowable disturbance bound $\bar{\mathcal{E}}_\gamma^l = \{x \in \mathbb{R}^n | A^{\mathcal{E}} x \leq \gamma \mathbb{1}^{n_\varepsilon}\} \subset \mathbb{R}^{n_x}$ with scaling factor $\gamma > 0$. If $L_{\mathcal{E}} \leq c\epsilon$ for some $c > 0$, then there exists a $\gamma > 0$ small enough that initial feasibility of (61) ensures safe system operation for all future times according to (2) at probability level p_s.

Theorem 6 states that Lipschitz continuity allows designing the learning-based PSF problem (61) using the iterative constraint tightening sequence (62) in combination with the admissible disturbance bound $\bar{\mathcal{E}}_\gamma^l$ along backup trajectories. The remaining tuning parameters are therefore limited to the scalars ϵ and γ. Furthermore, if $L_{\mathcal{E}}$ is small enough for a selected ϵ, a sufficiently small $\gamma > 0$ exists such that initial feasibility implies constraint satisfaction at probability level p_s for all times. Intuitively, $L_{\mathcal{E}}$ sufficiently small means that the difference between $\mathcal{E}(x, u)$ and $\mathcal{E}(x + \Delta x, u + \Delta u)$ must be small for small values $\Delta x, \Delta u$, such that the error bound is not allowed to change rapidly. In the case of GP Regression using a squared exponential kernel, this relates either to a sufficiently large length-scale parameter or homogeneous data coverage [138].

If problem (61) is not initially feasible due to the confident subset constraint (61g), either the model needs to be refined using additional data, or the probability level p_s can be lowered, since $p_s \to 0$ typically implies $\mathcal{E}(x, u) \to \{0\}$. While the exact values of L_f, $L_{\mathcal{E}}$, c, γ are challenging to compute explicitly, the discussion in [55, Section 4.3] using $\rho = L$ provides an extensive practical tuning guideline with a statistical verification procedure. Note that conservativeness can further be reduced using incremental Lyapunov functions [143] instead of Lipschitz continuity of (58) [55].

CONCLUSION

This article provides an introduction to three approaches for constructing safety filters for safety-critical control design, and discusses recent research that has sought to unify these techniques. The prospect of bridging the gap between first-principle models and real-world systems through data is a topic on the forefront of research in control theory and applications. We highlight how the three safety-filter techniques can be integrated with learning-based models to yield theoretical and practical safety guarantees in the face of model uncertainty. Applications demonstrating each of the safety filter techniques are presented and show that the proposed approaches are promising solutions for real engineering challenges.

While we have provided an overview of standard forms and data-driven extensions of Hamilton-Jacobi reachability, control barrier functions, and predictive safety filters, there remain several interesting directions for research, both in and outside of a data-driven paradigm. The design of hierarchical control structures for achieving safety that blend the three techniques may be able to capitalize on advantages regarding scalability, optimality, and computational efficiency present in each technique to produce both performant and robust safety filter designs. A second set of questions relates to the safe collection of data for a dynamic system, the processing of large data sets for efficient evaluation of learning models and uncertainty estimates to enable integration into high-frequency closed-loop controllers, and facing the challenges of systems for which previous data becomes obsolete as the system changes over time. We believe that each of the questions can not be answered in isolation, but rather is best answered by considering the impact of data through the lens of safety filter design.

AUTHOR INFORMATION

Kim P. Wabersich (wabersich@kimpeter.de) received a BSc. and MSc. degree in engineering cybernetics from the University of Stuttgart in Germany in 2015 and 2017, respectively. He completed his doctoral studies at ETH Zurich in 2021 and is currently a postdoctoral researcher with the Institute for Dynamic Systems and Control (IDSC) at ETH Zurich. His research interests include learning-based model predictive control and safe model-based reinforcement learning.

Andrew J. Taylor (ajtaylor@caltech.edu) received the B.S. and M.S. degrees in aerospace engineering from the University of Michigan, Ann Arbor, in 2016 and 2017, respectively. He is currently pursuing a Ph.D. degree at the Caltech in Control and Dynamical Systems. His research interests include safety-critical control for robotic systems and data-driven control techniques for nonlinear systems. He is a student member of IEEE.

Jason J. Choi (jason.choi@berkeley.edu) received the B.S. degree in mechanical engineering from Seoul National University in 2019. He is currently pursuing a Ph.D. degree at University of California Berkeley in Mechanical Engineering. His research interests center on optimal control theories for nonlinear and hybrid systems, data-driven methods for safe control, and their applications to robotics and autonomous mobility.

Koushil Sreenath (koushils@berkeley.edu) is an Associate Professor of Mechanical Engineering, at UC Berkeley. He received a Ph.D. degree in Electrical Engineering and Computer Science and a M.S. degree in Applied Mathematics from the University of Michigan at Ann Arbor, MI, in 2011. He was a Postdoctoral Scholar at the GRASP Lab at University of Pennsylvania from 2011 to 2013 and an Assistant Professor at Carnegie Mellon University from 2013 to 2017. His research interest lies at the intersection of highly dynamic robotics and applied nonlinear control. His work on dynamic legged locomotion on the bipedal robot MABEL was featured on The Discovery Channel, CNN, ESPN, FOX, and CBS. His work on dynamic aerial manip-

ulation was featured on the IEEE Spectrum, New Scientist, and Huffington Post. His work on adaptive sampling with mobile sensor networks was published as a book entitled Adaptive Sampling with Mobile WSN (IET). He received the NSF CAREER, Hellman Fellow, Best Paper Award at the Robotics: Science and Systems (RSS), and the Google Faculty Research Award in Robotics.

Claire J. Tomlin (tomlin@eecs.berkeley.edu) is the Charles A. Desoer Professor of Engineering in the Department of Electrical Engineering and Computer Sciences (EECS), University of California Berkeley (UC Berkeley). She was an Assistant, Associate, and Full Professor in Aeronautics and Astronautics at Stanford University from 1998 to 2007, and in 2005, she joined UC Berkeley. Claire works in the area of control theory and hybrid systems, with applications to air traffic management, UAV systems, energy, robotics, and systems biology. She is a MacArthur Foundation Fellow (2006), an IEEE Fellow (2010), and in 2017, she was awarded the IEEE Transportation Technologies Award. In 2019, Claire was elected to the National Academy of Engineering and the American Academy of Arts and Sciences.

Aaron D. Ames (ames@caltech.edu) is the Bren Professor of Mechanical and Civil Engineering and Control and Dynamical Systems at Caltech. Prior to joining Caltech in 2017, he was an Associate Professor at Georgia Tech in the Woodruff School of Mechanical Engineering and the School of Electrical & Computer Engineering. He received a B.S. in Mechanical Engineering and a B.A. in Mathematics from the University of St. Thomas in 2001, and he received a M.A. in Mathematics and a Ph.D. in Electrical Engineering and Computer Sciences from UC Berkeley in 2006. He served as a Postdoctoral Scholar in Control and Dynamical Systems at Caltech from 2006 to 2008, and began his faculty career at Texas A&M University in 2008. At UC Berkeley, he was the recipient of the 2005 Leon O. Chua Award for achievement in nonlinear science and the 2006 Bernard Friedman Memorial Prize in Applied Mathematics, and he received the NSF CAREER award in 2010, the 2015 Donald P. Eckman Award, and the 2019 IEEE CSS Antonio Ruberti Young Researcher Prize. His research interests span the areas of robotics, nonlinear, safety-critical control and hybrid systems, with a special focus on applications to dynamic robots -— both formally and through experimental validation.

Melanie N. Zeilinger (mzeilinger@ethz.ch) is an Assistant Professor at ETH Zurich, Switzerland. She received the Diploma degree in engineering cybernetics from the University of Stuttgart, Germany, in 2006, and the Ph.D. degree with honors in electrical engineering from ETH Zurich, Switzerland, in 2011. From 2011 to 2012 she was a Postdoctoral Fellow with the École Polytechnique Fédérale de Lausanne (EPFL), Switzerland. She was a Marie Curie fellow and Postdoctoral Researcher with the Max Planck Institute for Intelligent Systems, Tübingen, Germany until 2015 and with the Department of Electrical Engineering and Computer Sciences at the University of California at Berkeley, CA, USA, from 2012 to 2014. From 2018 to 2019 she was a professor at the University of Freiburg, Germany. Her current research interests include safe learning-based control, as well as distributed control and optimization, with applications to robotics and human-in-the-loop control. She is a member of IEEE.

Safety Filter Design Example

This sidebar illustrates and compares the basic safety filter methodologies by applying each of them to the inverted pendulum system

$$\underbrace{\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\theta\\ \dot\theta\end{bmatrix}}_{\dot x} = \underbrace{\begin{bmatrix}\dot\theta\\ \frac{g}{\ell}\sin\theta\end{bmatrix}}_{f(x)} + \underbrace{\begin{bmatrix}0\\ \frac{1}{m\ell^2}\end{bmatrix}}_{g(x)} u, \tag{S1}$$

where the pendulum angle and angular velocity $[\theta, \dot\theta] = [x_1, x_2] = x$ define the system state and u is the input torque applied at the base of the pendulum. The system parameters consist of the mass $m = 2$ kg, length $\ell = 1$ m, and gravitational acceleration $g = 10$ m/s^2. The physical input limitation is a maximum applicable torque of 3 N-m, that is, $\mathcal{U} = \{u \in \mathbb{R} \mid |u| \leq 3\}$. The safety constraints are defined as pendulum angle and angular velocity constraints of the form $\mathcal{X} = \{x \in \mathbb{R}^2 \mid |x_1| \leq 0.3, |x_2| \leq 0.6\}$.

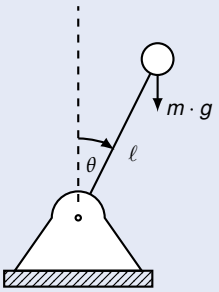


FIGURE S1 Inverted pendulum control system.

DESIRED CONTROL INPUT SIGNAL

To compare the different safety filter designs with respect to the 'ideal' safety filter objective (4), we use the desired control signal

$$u_{\mathrm{des}}(t) = \begin{cases} 3, & t \in [0,2),\\ -3, & t \in [2,4),\\ 3, & t \in [4,6),\\ m\ell^2\left(-\frac{g}{\ell}\sin x_1 - [1.5,\ 1.5]x\right), & \text{else.}\end{cases} \tag{S2}$$

By alternating between maximum and minimum torque the desired input signal (S2) tries to violate the system constraints, requiring safety filter intervention. The adversarial input section is followed by a stabilizing feedback control law, which does not consider constraint satisfaction explicitly.

SWITCHING SAFETY FILTER

This section demonstrates how to construct the switching safety filter (11) using a linear-quadratic regulator (LQR) of the form $\kappa_{\mathcal{S}}(x) = -Kx$. The design of $\kappa_{\mathcal{S}}$ is based on the linearization of the system dynamics (S1) around the upward equilibrium point

$$\Delta\dot x = \underbrace{\begin{bmatrix}0 & 1\\ \frac{g}{\ell} & 0\end{bmatrix}}_{A}\Delta x + \underbrace{\begin{bmatrix}0\\ \frac{1}{m\ell^2}\end{bmatrix}}_{B}\Delta u. \tag{S3}$$

Using the state cost $Q = 25I_2$ and input cost $R = 1$, we obtain the gain $K = [40.62,\ 13.69]$. An invariant set for (S3) is selected as the sublevel set of the LQR Lyapunov function [80, Chapter 4]

$$\mathcal{S}_\gamma = \{x \in \mathbb{R}^2 | \gamma - x^\top P x \geq 0\}, \tag{S4}$$

for some $\gamma > 0$ and the positive definite matrix

$$P = \begin{bmatrix}282.26 & 81.23\\ 81.23 & 27.38\end{bmatrix}. \tag{S5}$$

To ensure that $\mathcal{S}_\gamma$ is forward invariant under $\kappa_{\mathcal{S}}$ in the presence of the input constraints $\mathcal{U}$, the level set γ must be selected such that $\kappa_{\mathcal{S}}(x) \in \mathcal{U} \Leftrightarrow |-Kx| \leq 3$ for all $x \in \mathcal{S}_\gamma$ and $\mathcal{S}_\gamma \subseteq \mathcal{X}$. Using the support function of $\mathcal{S}_\gamma$ [8], we obtain a maximal value of $\gamma = 1.31$, for which we denote the safe set $\mathcal{S} \triangleq \mathcal{S}_{1.31}$. To certify forward invariance of $\mathcal{S}$ with respect to the nonlinear system (S1), we verify that

$$\max_{x\in\mathcal{S}} -2x^\top P(f(x) - g(x)Kx) \geq 0, \tag{S6}$$

through nonlinear programming [33]. The resulting safe set is depicted in Figure S2 (top).

The previously described constructions allow us to implement the switching-based safety filter in (11) as

$$\kappa_F(x, u_{\mathrm{des}}(t)) = \begin{cases} -Kx, & x \in \partial\mathcal{S} \text{ or } |u_{\mathrm{des}}(t)| > 3,\\ u_{\mathrm{des}}(t), & \text{else.}\end{cases} \tag{S7}$$

The safety controller is used for 0.01 s when it is activated. The closed-loop simulation of the resulting control structure as depicted in Figure 2 together with the desired input signal (S2) are shown in Figure S2. After significant intervention during the first six seconds, the desired control input signal meets safety requirements and input bounds (for $t \in [6, 10]$) and is used.

Even though safety is achieved during the entire evolution of the system, limitations that motivate the advanced techniques presented in this paper may be observed. The derived safe set $\mathcal{S}$ and safe controller $\kappa_{\mathcal{S}}$ yield a conservative safety filter, which can be seen by the overly large safety margin between the safe set and the angular constraints in Figure S2 (top). To reduce such conservativeness, Hamilton-Jacobi (HJ) reachability and predictive safety filters (PSFs) integrate optimal control based approaches as demonstrated in the upcoming sections. Furthermore, the switching-based safety control law (S7), derived from (11) can result in significant input chattering behavior near the boundary of the safe set as seen in Figure S2 (bottom). Such behavior is not desirable in practice. To this end, control barrier functions (CBF) enable a safety filter formulation which yields a smooth control input signal.

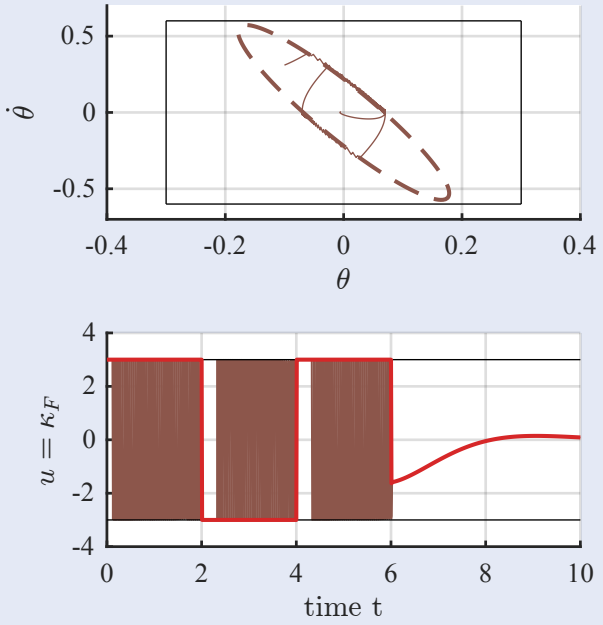


FIGURE S2 Application of the switching safety filter (S7) to the inverted pendulum example (S1). **(Top)** The desired safety constraints $\mathcal{X}$ (solid black line), the switching safety filter safe set (dashed brown line), and closed-loop trajectory (solid brown line). **(Bottom)** Applied input trajectory (brown) and desired input signal (red). The constraints are indicated with solid black lines in each plot. While safety is maintained, the switching safety filter (S7) causes undesirable input chattering and the safe set only covers a small portion of $\mathcal{X}$.

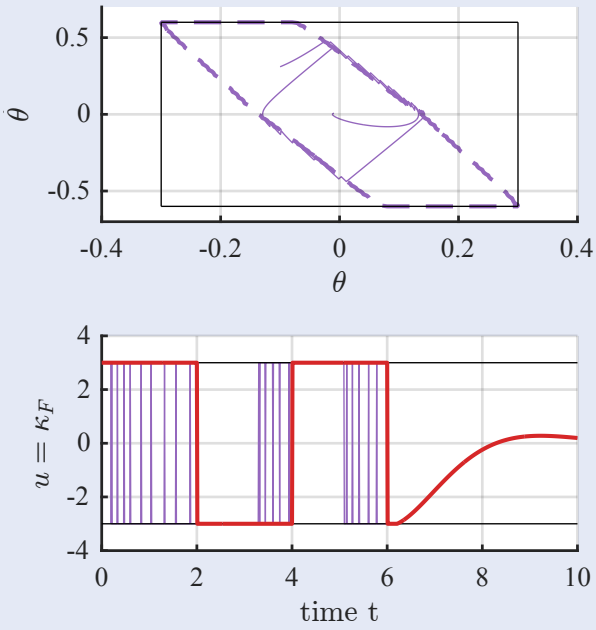


FIGURE S3 Application of the Hamilton-Jacobi based safety filter to the inverted pendulum example (S1). **(Top)** The Hamilton-Jacobi based safe set (dashed purple line) and closed-loop trajectory (solid purple line). **(Bottom)** Applied input trajectory (purple) and desired input signal (red). The safety filter is significantly less intrusive compared to the switching safety filter due to the larger safe set $\mathcal{S}$, and displays notably less chattering in the control input signal.

HAMILTON-JACOBI REACHABILITY SAFETY FILTER

This section demonstrates how HJ reachability allows reducing the conservativeness of the switching safety filter (S7). The value function V defined in (15), which describes the maximal viability kernel of $\mathcal{X}$, is computed by solving the HJ Variational Inequality (23) numerically using a sufficiently large finite time horizon T with the HJ optimal control toolbox (helperOC) [S1] and the Level Set Toolbox [68]. A 101×201 grid is constructed on the set $\mathcal{X}$ and a finite horizon $T = 2.5$ s is used for this computation, which takes roughly a minute on a standard laptop. The safe set (16) resulting from Theorem 2 is

$$\mathcal{S} = \{x \in \mathbb{R}^2 | V(x) \geq \epsilon\}, \tag{S8}$$

with $\epsilon = 0.02$ to account for numerical approximation errors. See Figure S3 (top) for an illustration of $\mathcal{S}$, which represents an approximation of the maximal control invariant set in $\mathcal{X}$ based on Theorem 2. The HJ safety filter is implemented as in (22), and shows a larger safe set than the switching safety filter, leading to fewer interventions (and correspondingly less chatter in the input signal) when $t \in [0, 6]$, as seen in Figure S3.

CONTROL BARRIER FUNCTION SAFETY FILTER

With the goal of reducing the undesirable input chattering of the previous techniques seen in Figures S2 and S3, we next construct a safety filter using CBFs. To this end, we follow the example presented in [S3] and select

$$h_{\mathcal{S}}(x) = 1 - x^\top \begin{pmatrix} 1/a^2 & 0.5/ab \\ 0.5/ab & 1/b^2 \end{pmatrix} x \tag{S9}$$

with parameters $a, b > 0$ as a candidate CBF, yielding a 0-superlevel set

$$\mathcal{S} = \{x \in \mathbb{R}^{n_x} \mid h_{\mathcal{S}}(x) \geq 0\} \tag{S10}$$

describing the safe set, similarly to (S4). The quantities a and b must be selected to ensure that $h_{\mathcal{S}}$ satisfies the condition (27) for some $\alpha \in \mathcal{K}^e$. We consider a function $\alpha \in \mathcal{K}^e$ of the form $\alpha(r) = c_\alpha r$ with $c_\alpha > 0$ to be determined. The CBF supremum condition (27) can be equivalently (modulo input constraints) expressed as the implication [S3]

$$\nabla h_{\mathcal{S}}(x) g(x) = 0 \Rightarrow \nabla h_{\mathcal{S}}(x) f(x) + \alpha(h_{\mathcal{S}}(x)) > 0, \tag{S11}$$

which in the inverted pendulum setting appears as

$$\nabla h_{\mathcal{S}}(\overline{x}) g(\overline{x}) = 0 \Rightarrow \overline{x}_2 = -\frac{b}{2a}\overline{x}_1, \tag{S12}$$

for $\overline{x} \in \mathbb{R}^2$.

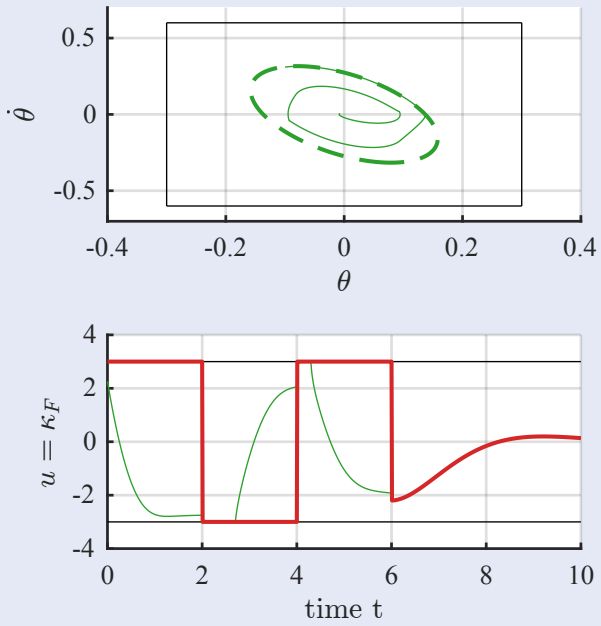


FIGURE S4 Application of the control barrier function based safety filter to the inverted pendulum example (S1). **(Top)** The control barrier function based safe set (dashed green line) and closed-loop trajectory (solid green line). **(Bottom)** Applied input trajectory (green) and desired input signal (red). The safety filter smoothly modifies the desired control input signal while ensuring that the system remains safe, though the safe set is smaller than the Hamilton-Jacobi approach.

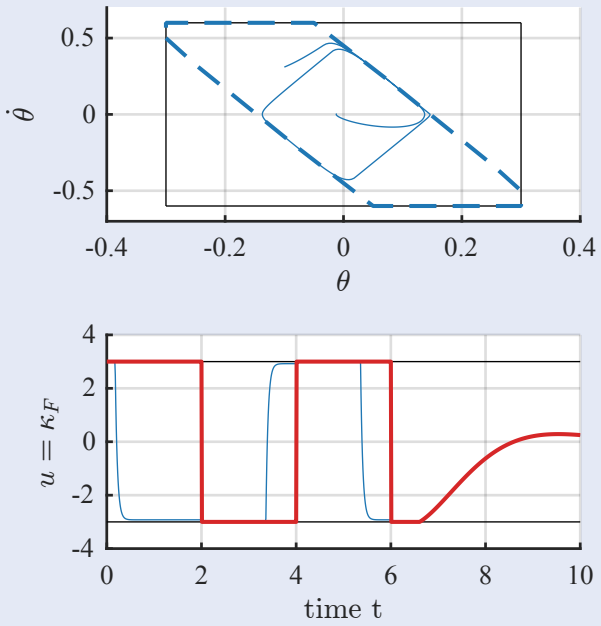


FIGURE S5 Application of the predictive safety filter to the inverted pendulum example (S1). **(Top)** The implicit predictive safe safe set with a horizon length of $N = 20$ (dashed blue line) and closed-loop trajectory (solid blue line). **(Bottom)** Applied input trajectory (blue) and desired input signal (red). The safety filter anticipates jumps in the desired control input signal and changes the input preemptively, yielding smooth behavior.

For an $\overline{x}$ such that $\nabla h_S(\overline{x})g(\overline{x}) = 0$, we have that

$$\nabla h_S(\overline{x})f(\overline{x}) + \alpha(h_S(\overline{x})) = c_\alpha + \frac{3}{4a^2}\left(\frac{b}{a} - c_\alpha\right)\overline{x}_1{}^2. \quad \text{(S13)}$$

We see that the required implication is satisfied by choosing $c_\alpha \leq \frac{b}{a}$. We select the values $a = 0.075$ and $b = 0.15$ considering the state and input constraints $\mathcal{X}$ and $\mathcal{U}$, respectively, and select $c_\alpha = 0.2$. The resulting safe set is visualized in Figure S4 (top). The safety filter can be implemented by solving (29) numerically using the standard YALMIP solver [S4]. We note that the system is kept safe, and the chattering in the control input signal is eliminated (with jumps only occurring at discontinuities in the desired control input signal), as seen Figure S4. We note that the safe set obtained using this approach is notably smaller than the one used with the HJ reachability safety filter. Developing constructive approaches for synthesizing less conservative CBFs to bridge this gap is a topic of ongoing research.

PREDICTIVE SAFETY FILTER

We next implement a PSF that uses a receding horizon approach to enable smooth filtering of control inputs while maintaining a large control invariant set. The first step to construct a PSF as in (31) is taking the Euler time discretization of the dynamics (30). Using a discretization time of $\Delta T = 0.05$ yields the discrete dynamics

$$x(k+1) = x(k) + 0.5(f(x(k)) + g(x(k)))u(k). \quad \text{(S14)}$$

We next construct a terminal invariant set $\mathcal{S}^t \subset \mathcal{X}$. Application of the linearization-based approach on page 10 at the origin with $c = 0.2$ and $\gamma = 1$ yields a terminal invariant set

$$\mathcal{S}_1^t = \{x \in \mathbb{R}^{n_x} | 1 - x^\top P^t x \geq 0\} \quad \text{(S15)}$$

with

$$P^t = \begin{bmatrix} 128.10 & 41.13 \\ 41.13 & 15.98 \end{bmatrix} \quad \text{(S16)}$$

We use IPOPT [S5] with the automatic differentiation tool Casadi [S6] to verify using randomly selected warm-starts that

$$x^* = \underset{x \in \mathcal{S}_1^t}{\operatorname{argmax}}\, R(x) < 0. \quad \text{(S17)}$$

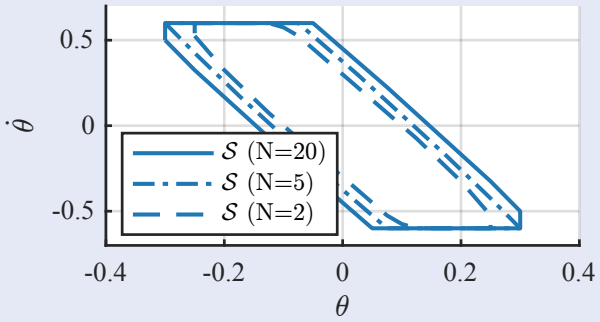


FIGURE S6 Implicit safe set $\mathcal{S}_N^{\text{PSF}}$ (32) of the predictive safety filter (31) for different planning horizons N. Longer horizons lengths increase the size of $\mathcal{S}_N^{\text{PSF}}$ until it converges to the maximal control invariant set in $\mathcal{X}$.

We implement (31) with a planning horizon of $N = 20$ also using IPOPT [S5] and Casadi [S6]. While solve time is not critical in simulation, real-world applications may require tailored algorithms and software packages, see for example, [34, Section 12] and references therein. Figure S5 illustrates the resulting safe set and closed-loop trajectories. While the PSF is both permissive and smoothly filters the desired control input signal, the required online computations increase by multiple orders of magnitude. The computational load can typically be balanced by reducing the planning horizon, which, however, also reduces the corresponding implicit safe set as illustrated in Figure S6.

COMPARISON OF APPROACHES

We compare the various safety filter implementations in Figure Figure S7. In the top figure, we compare the different safe sets (including two for the predictive safety filter using different horizons). We see that the HJ reachability safety filter (purple) contains the safe sets for the switching safety filter, CBF safety filter, and the PSF using the shorter horizon length. The PSF using the longer horizon contains the HJ reachability safe set, which is due to using $\epsilon = 0.02$ to account for numerical error when finding the HJ reachability safe set. The bottom figure shows the integral of the deviation of the input from the desired control input signal (4a). The switching safety filter and PSF with a short horizon have the biggest deviation, while the methods resulting in the largest safe sets modify the desired control input signal the least.

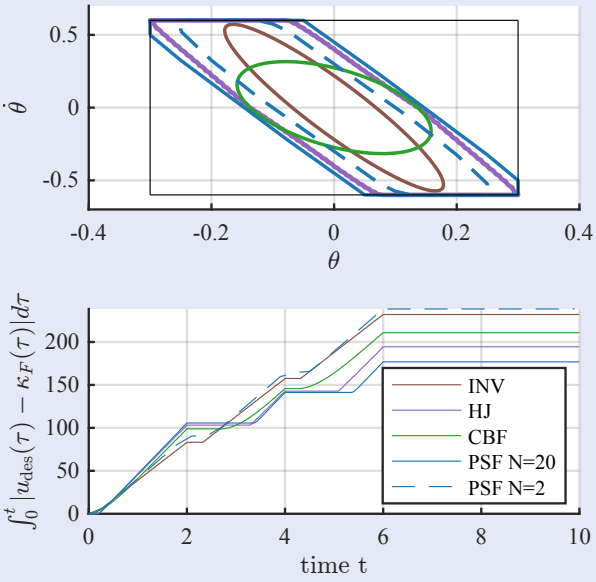


FIGURE S7 Comparison of various safety filters methods. **(Top)** The safe sets associated with each technique using the color codes in Figures S2-S6. **(Bottom)** The value of the safety filter objective (4a). Lower values of quantity indicate that the safety filter permits more use of the desired control input signal.

REFERENCES

[S1] , Bansal, S., Chen, M., Herbert, S., & Tomlin, C. J. "Hamilton-jacobi reachability: A brief overview and recent advances",in *Proc. IEEE 56th Conf. on Decision and Control (CDC)*, Melbourne, VIC, Australia, 2017, pp. 2242-2253

[S2] , Mitchell, I. M. & Templeton, J. A., "A toolbox of Hamilton-Jacobi solvers for analysis of nondeterministic continuous and hybrid system", in *Int. Work. on Hybrid Sys.: Computation and Control*, Springer, 2005, pp.480-494.

[S3] Alan, A., Taylor, A. J., He, C. R., Ames, A. D., & Orosz, G. "Control Barrier Functions and Input-to-State Safety with Application to Automated Vehicles", arxiv:2206.03568, 2022.

[S4] Lofberg, J., "YALMIP: A toolbox for modeling and optimization in MATLAB," in *Proc. IEEE Int. Conf. Robotics and Automation*., New Orleans, LA, USA, 2004, pp. 284-289.

[S5] Wächter, A., & Biegler, L., "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming." in *Mathematical Programming*, pp. 25-57, 2006.

[S6] Andersson, J. A., Gillis, J., Horn, G., Rawlings, J. B., & Diehl, M., "CasADi: a software framework for nonlinear optimization and optimal control." in *Mathematical Programming*, pp. 1-36, 2019.

Hamilton-Jacobi Reachability Safety Filter Applications

Hamilton-Jacobi (HJ) reachability provides an effective tool for guaranteeing and verifying performance and safety properties of a system. The notion of a reachable set can be used to describe regions in the state space from which achieving performance goals or satisfying safety constraints is feasible. Such sets are often characterized as level sets of a value function of an optimal control or differential game problem, for instance, as in Theorem 2. Moreover, when a controller is not pre-specified, reachability formulations can be used to synthesize controllers that achieve safety and performance in an optimal manner as in (20). Finally, model uncertainty and exogenous disturbances can be directly incorporated into reachability formulations, permitting the construction of robust control invariant sets. The availability of tools [68] for computing value functions establishes HJ reachability as a framework for constructive verification and safe control synthesis. This have led to the application of HJ reachability in safety-critical real-world settings such as aircraft traffic management [9], real-time motion planning [144], verification of acrobatic drones [S2] as seen in Figures S8 and S9, and autonomous vehicle navigation [S3] as seen in Figure S10.

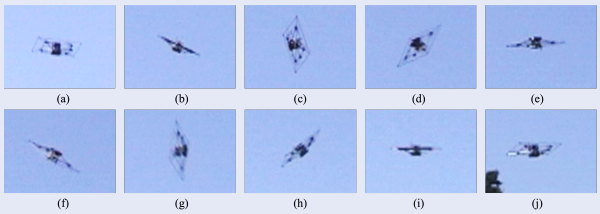


FIGURE S8 A mosaic of an autonomous back-flip for the STARMAC quadrotor [S1]. The controller takes the system through a sequence of mode transition from initiating the impulse mode (a), in which the vehicle rotation is induced from strong motor thrust, entering the drift mode (b), where it turn offs the motors and continues free-falling in (b)–(f), and entering the recovery mode (f), in which quadrotor returns to hovering in (f)–(j) [S2].

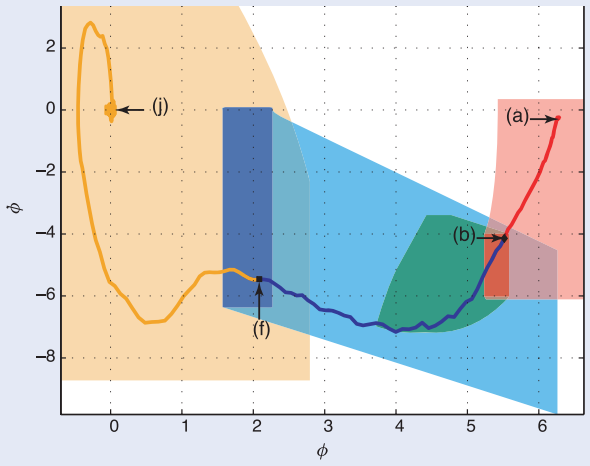


FIGURE S9 Reachable sets in the pitch angle ϕ and pitch rate $\dot{\phi}$ of the drone back-flip maneuver seen in Figure S8. A pitching thrust is applied in the light red region, and the drone transitions from a pitch thrust to drifting in the dark red region. The drone transitions from drifting to recovering in the dark blue region, after which it arrives at an equilibrium configuration. The ability to perform the back-flip while ensuring a safety constraint on the minimum altitude of the vehicle are verified by analyzing reachable sets for the full system during the impulse, drift, and recovery stages of the vehicle. [S2].

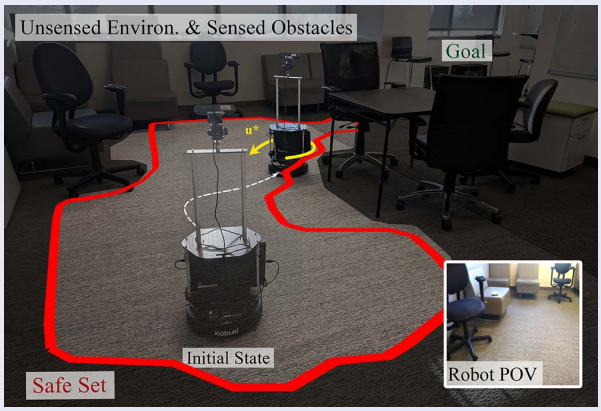


FIGURE S10 Safe autonomous navigation framework for an *a priori* unknown environment based on HJ reachability. The framework treats unexplored portions of the environment as an obstacle and uses HJ reachability to compute the safe region and the safe controller for the vehicle, which is updated in real-time as the vehicle explores the environment. A vision-based, learning-based planner is deployed to reach the navigation goal while the HJ reachability-based safety filter (22) keeps the robot safe when it is at risk of colliding with obstacles [S3].

REFERENCES

[S1] Hoffmann, G., Rajnarayan, D. G., Waslander, S. L., Dostal, D., Jang, J. S., & Tomlin, C. J. "The Stanford testbed of autonomous rotorcraft for multi agent control (STARMAC)," in *Proc. AIAA/IEEE 23rd Digital Avionics Sys. Conf.*, Salt Lake City, UT, USA, 2004, pp. 12-E.

[S2] Gillula, J.H., Hoffman, G.M, Huang, H., Vitus, M.P., & Tomlin, C.J. "Applications of hybrid reachability analysis to robotic aerial vehicles," in *The Int. J. of Robotics Research*, vol. 30, no. 3, pp. 335-354, 2011.

[S3] Bajcsy, A., Bansal, S., Bronstein, E., Tolani, V., & Tomlin, C.J. "An efficient reachability-based framework for provably safe autonomous navigation in unknown environments," in *Proc. IEEE 58th Conf. on Decision and Control (CDC)*, Nice, France, 2019, pp. 1758-1765.

Reachability-Based Safe Learning Framework: Experimental Results

The Hamilton-Jacobi (HJ) reachability-based safe learning framework proposed in [22] has been demonstrated on a quadrotor subjected to unknown dynamics due to wind effects. The quadrotor attempts to track a reference trajectory using either a linear quadratic regulator or a tracking policy learned online. A HJ reachability-based safety filter is utilized to prevent the quadrotor from colliding with its environment. However, a safety filter that is overly conservative may hinder not only the tracking performance, but also the training of the learning-based policy by preventing it from adequate exploration. To reduce conservativeness, the safety filter must address the unknown dynamics by learning from the actual system data, revealing a balance between safety and learning that must be achieved.

Figure S11 shows a phase portrait of the vertical position and velocity coordinates of the quadrotor as it learns a tracking policy. The conservativeness of the safe set is reduced over time as the learning model improves, eventually allowing the learning-based policy to successfully perform the tracking task while avoiding collisions. In Figure S12, a strong wind is introduced near the ground, which the system has not encountered before. Reliance on the previously learned policy that is unaware of this disturbance leads to a deterioration of safety as seen in scenario (a). However, when the accuracy of the learned model is validated online using data that captures the new unknown dynamics, the system is kept away from the region where the model is unreliable until a new model can be trained, thus leading to safety as seen in scenario (b).

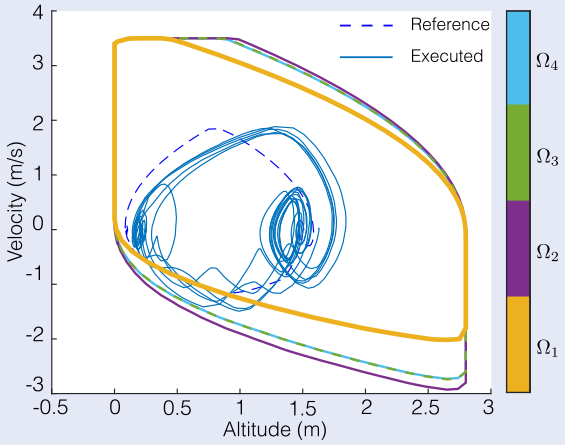
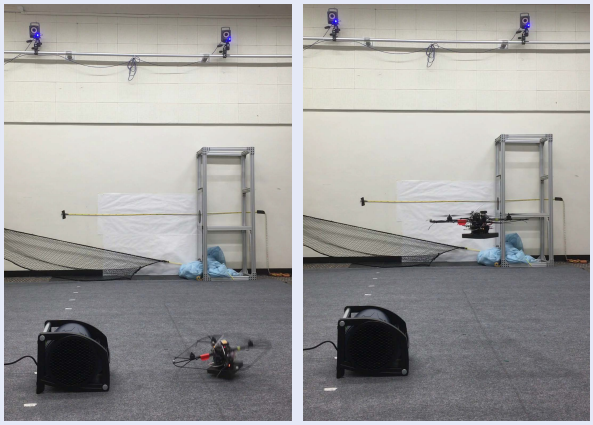


FIGURE S11 Quadrotor altitude HJ reachability safe sets being updated online through learning. The sets progress from $\mathcal{S}^1$ to $\mathcal{S}^4$ as the system gathers data and successively improve the learned dynamics model [22].

Finally, the experiment is extended to simulation with the quadrotor tracking a figure-eight reference trajectory in 3D space. While in the previous scenarios the HJ safe set computation is done only for the vertical dynamics, to ensure safety constraints in the full 3D environment, the HJ safe set computation is done online for the 10D full quadrotor dynamics. The computation is facilitated by incorporating modern reachability computational techniques including state-decomposition [70], warm-starting [71], and adaptive gridding [45], which took an average of 206.6 s to update the safe set online.



(a) w/o online guarantee validation (b) with online guarantee validation

FIGURE S12 Quadrotor learning a vertical flight policy while avoiding collisions with the ground. When the fan is turned on, the system experiences unknown dynamics which have not appeared in previous data, which can lead to a ground collision using the previous learned policy. An online validation method detects that the previously learned model fails to describe the new unknown dynamics, and utilizes a safe controller that avoids regions of the state space (close to the fan) where the new unknown dynamics are present [22].

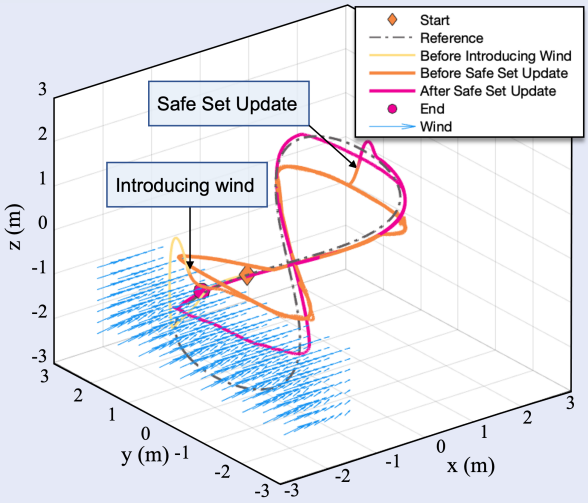


FIGURE S13 Trajectories of the quadrotor tracking a reference trajectory using a linear quadratic regulator in 3D space. The quadrotor begins in yellow, and then experiences a sudden change in wind (blue arrows). While the safe set is updated to account for the new unknown dynamics, online validation of the learned model prevents the trajectory from passing the uncertain wind area (orange trajectory), until the safe set update is complete (pink trajectory) [45].

Control Barrier Function Safety Filter Applications

Control barrier function (CBF) based safety filters have seen extensive use in real-world applications, including mobile robots [83], robotic swarms [84], aerial vehicles [85], robotic arms [86], robotic manipulators [87], quadrupedal robots [88], bipedal robots [50], and automotive systems [89]. Practical safety tasks can often be encoded using the notion of forward invariance, such safe foot placement on viable footholds, maintaining a safe following distance, avoiding obstacles in a complex dynamic environment, or respecting positioning constraints, as seen in the various examples in Figures S14-S17.

In each of these dynamic applications safe control computations must be performed quickly. The formulation of CBF-based safety filters via convex optimization programs such as in (29) permits a reliable and efficient means to quickly filter desired control input signals. Several of these examples incorporate horizon-based elements seen in Hamilton Jacobi and predictive filter methods, either in the use of low-rate predictive filters (Figure S14), offline reference trajectories (Figure S16) or backup set CBFs (Figure S17) to achieve both strong closed-loop performance in addition to safety.

FIGURE S14 Control barrier function (CBF) safety filter on a quadruped. A multilayered safety filter design is used that integrates predictive safety filters with CBFs to ensure safe foot placement on viable footholds while maintaining system stability. CBF constraints are integrated into both a mid-level predictive filter and a low-level CBF based filter given by (29), ensuring a consistent safety specification across planning and control layers [88].

FIGURE S16 Control barrier function (CBF) safety filter on a robotic arm in an industrial kitchen. Maintaining safety in a dynamic work environment shared with human personnel requires online modification of arm reference trajectories, but directly recomputing trajectories online is computationally intractable for real-time operation. CBF-based safety filters are used to efficiently modify trajectories given ongoing changes in the environment. [86].

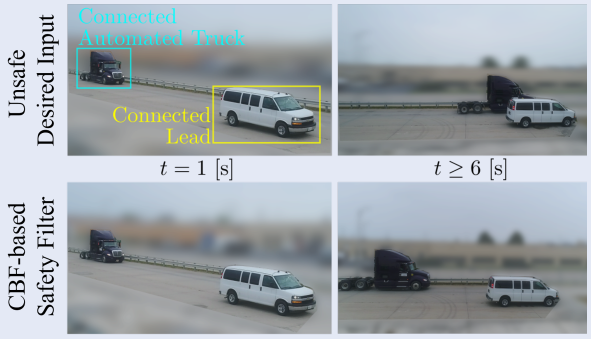


FIGURE S15 Control barrier function (CBF) safety filter on a connected automated semi-trailer truck. The desired control input signal u_{des} is derived from an expert-designed controller that balances speed tracking with passenger comfort, but does not keep the system safe. A CBF-based safety filter constructed using the input-to-state safety notion of robustness ensures safety of the truck in the presence of complex unmodeled braking system dynamics [89].

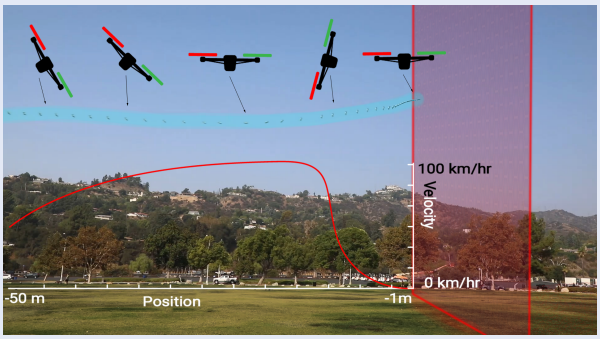


FIGURE S17 Control barrier function (CBF) safety filter for geofencing of high speed drone. A backup set CBF safety filter is designed to safely filter pilot inputs for a racing drone flying at high speeds (100 $[\text{km/h}]$), enabling high-risk acrobatic maneuvers while maintaining safety. The lightweight nature of CBF-based safety filters permits the use of only onboard sensing and computation, enabling beyond line-of-sight operation and robustness to ground communication failures [145].

Data-Driven Control Barrier Function Safety Filter Applications

While the application of CBF-based safety filters does not require data-driven methods to deal with imperfect system models, there have been several applications where the incorporation of data has led to improvements in the safety of a systems. In this sidebar we highlight successful experimental implementations of data-driven CBF-based safety filters. In Figures S18 and S19 we see applications where learning models are used to mitigate the error between a system model and the physical system. In both examples, a baseline CBF-based safety filter given by (29) is modified with learning models, yielding safe behavior. Figure S20 shows an example of preference-based learning [146] being used to tune parameters of a robust CBF-based safety filter. By iteratively incorporating designer preferences on closed-loop system behavior, a CBF-based safety filter that balances performance with safety can be synthesized. These results demonstrate the potential of data-driven CBF-based safety-critical control design methodologies.

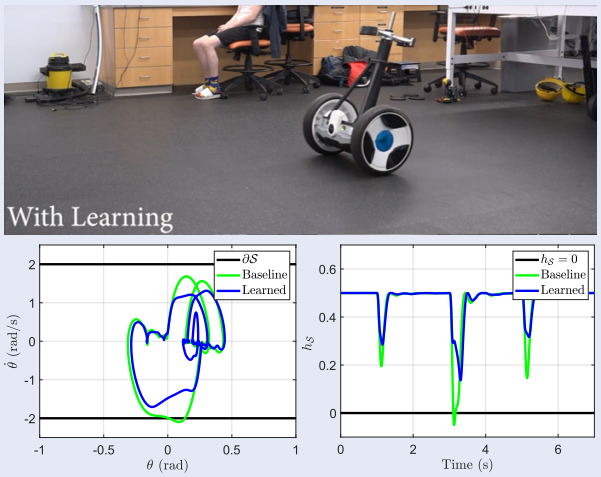


FIGURE S19 Learning control barrier function (CBF) time derivatives on a Segway robot. The baseline CBF-based safety filter (green curves) does not respect safety constraints on the pitch and pitch rate of the Segway due to error between the system model and the physical system. By integrating data-driven learning models into the CBF-based safety filter, safety of the system is achieved (blue curves). We note that although the baseline CBF-based safety filter does not respect safety constraints, the system remains close to the safe set, indicating input-to-state safe behavior with respect to model uncertainty inherent in CBF-based safety filters [24].

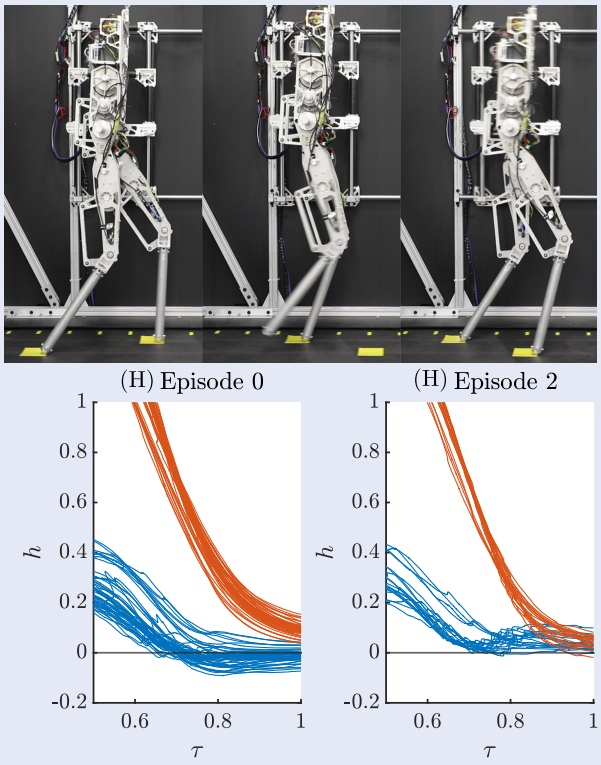


FIGURE S18 Learning control barrier function (CBF) time derivatives on the AMBER-3M bipedal robot. Walking robots often possess model uncertainty, making it difficult to satisfy precise foot placement constraints. By learning the impact of this model uncertainty on the dynamics of the CBF defining foot placement constraints, a CBF-based safety filter using learning models can be synthesized that reduces constraint violation. The two colored curves correspond to CBF values for constraints on each foot across multiple steps, with the constraint corresponding to the blue curves improving (remaining above zero) after incorporating learning models [50].

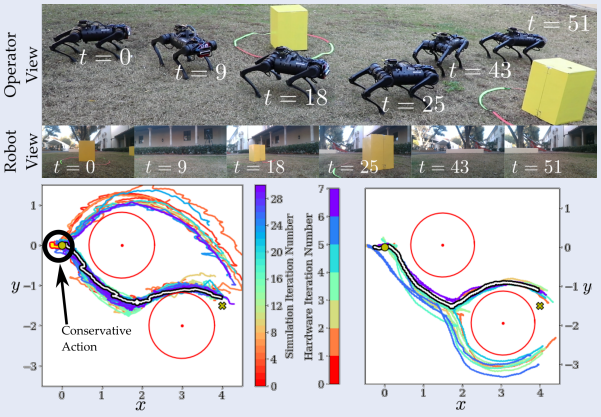


FIGURE S20 Preference-based learning for human-in-the-loop control barrier function (CBF) safety filter tuning. Facing uncertainty, the design of CBF-based safety filters must balance robustness and performance. Preference-based learning can translate a designer's evaluation of closed-loop behavior into controller parameter updates that achieve this balance. The initial CBF-based safety filter design overestimates uncertainties and yields conservative behavior with the quadruped remaining stationary. Incorporating user preferences to modify safety filter parameters allows the quadruped to navigate safely navigate obstacles, thus balancing safety and performance [147].

Predictive Safety Filter Applications: Experimental Race Cars and Simulated Quadrotors

In the following, we demonstrate two applications of predictive safety filters (PSFs). The first example considers an experimental miniature race car application as in [S1], which implements the soft constrained PSF (37) to enhance either a human driver or an imitation learning-based policy with safety guarantees. Parameters used in the drive-train dynamics and the Pacejka [S4, Section 13.5] tire model are identified from measurements. The second example demonstrates a probabilistic PSF formulation for a quadrotor as in [S2]. The constraints in (61) are implemented using a Bayesian model to ensure safety in probability during online controller tuning, during which ground crashes would occur without the filter in place.

SAFE MINIATURE RACE CAR OPERATION AND IMITATION LEARNING

We consider a dynamic bicycle model [S3, Section 2] with states $x = [p_x, p_y, \psi, v_x, v_y, r]$ and inputs $u = [\delta, \tau]$ as described in Table S1 and dynamics given by

$$\dot{x} = \begin{bmatrix} v_x \cos(\psi) - v_y \sin(\psi) \\ v_x \sin(\psi) + v_y \cos(\psi) \\ r \\ \frac{1}{m}\left(F_x - F_{yf}\sin(\delta) + m v_y r\right) \\ \frac{1}{m}\left(F_{yr} + F_{yf}\cos(\delta) - m v_x r\right) \\ \frac{1}{I_z}\left(F_{yf} l_f \cos(\delta) - F_{yr} l_r\right) \end{bmatrix}, \tag{S18}$$

where the lateral forces are modeled according to a Pacejka tire model [S4, Section 13.5] as

$$\alpha_f = \arctan\left(\frac{v_y + l_f r}{v_x}\right) - \delta, \quad \alpha_r = \arctan\left(\frac{v_y - l_r r}{v_x}\right), \tag{S19}$$

and

$$F_{yf/yr} = D_{f/r}\sin(C_{f/r}\arctan(B_{f/r}\alpha_{f/r})), \tag{S20}$$

and a drive-train model is used for the longitudinal force

$$F_x = C_1\tau + C_2\tau^2 + C_3 v_x + C_4 v_x^2 + C_5 \tau v_x. \tag{S21}$$

All parameters are described in Table S1, which have been identified using least-squares regression.

The input is limited by the maximum steering angle and maximum drive-train authority and the safety constraints require the vehicle to stay within track boundaries as depicted in Figure S21. The constraint set $\mathcal{X}$ is formulated in track-relative error states, which also simplifies the computation of the terminal invariant set according to Assumption 1 using convex approximations techniques [S1]. The PSF is implemented in a nominal fashion using soft constraints (37) to ensure practical feasibility. We consider a driver-assistance scenario as an experiment, with the desired input signal $u_{\text{des}}(k)$ provided by a human driver that is potentially unsafe with respect to the track boundary safety requirements. The PSF provides necessary interventions online to keep the vehicle safe in a minimally invasive fashion, yielding control of the vehicle to the driver as long as the driver's actions remain safe.

TABLE S1 States, Inputs and Parameters of Vehicle

State symbol	Quantity
$p_{x/y}$	x-y coordinates of the car
Ψ	Heading angle
$v_{x/y}$	Velocity in car frame
r	Yaw rate in car frame
Input symbol	**Quantity**
δ	Steering angle
τ	Drive-train command
Parameter symbol	**Quantity**
m	Mass
I_z	Yaw moment of intertia
$l_{f/r}$	Distance between center of gravitiy and front/rear axles
$D_{f/r}, C_{f/r}, B_{f/r}$	Pacejka tire model parameters
C_1, C_2, C_3, C_4, C_5	Drive-train model parameters

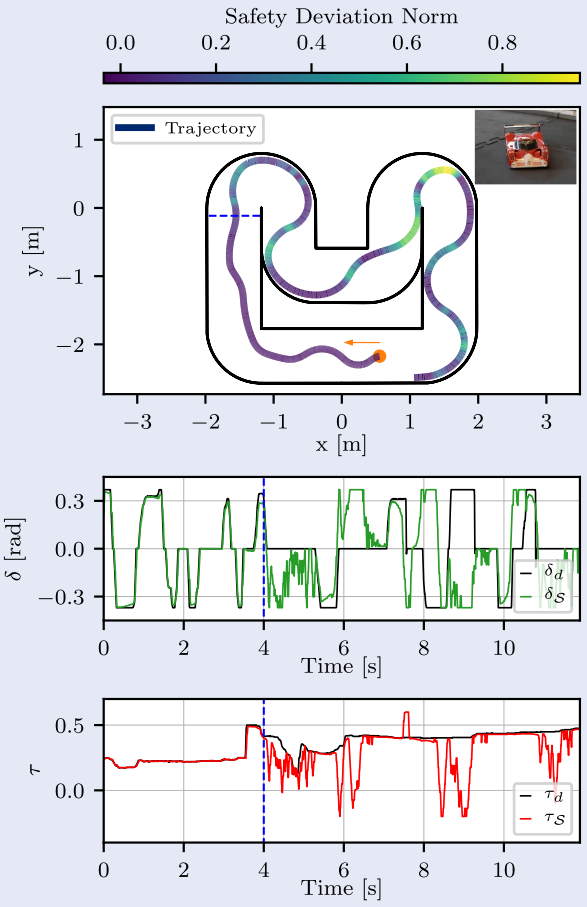


FIGURE S21 Miniature race car example. **(Top)** Vehicle trajectory with the magnitude of safety filter intervention. **(Middle, Bottom)** Human driver control inputs providing the desired control signal by a joystick, as well as control inputs resulting from the PSF. The dashed blue line indicates the transition from safe driver inputs to unsafe inputs.

Figure S21 illustrates a corresponding experiment with safety intervention magnitudes along a closed-loop trajectory. The input comparison plot shows the proposed desired input signals and the filtered, applied input signals. The human performs safe driving during the first four seconds, which can be seen by the unfiltered application of the proposed input signals. In contrast, after this initial time period, the driver purposefully applies unsafe actions, which do not pass the PSF and get modified to ensure safety as desired. As shown in the plot, the PSF keeps the vehicle within track boundaries at all times.

In addition to the driver assistance scenario, [S1, Section VI.B] demonstrates the combination of the same PSF with an imitation learning algorithm that reproduces a carefully selected expert policy using a deep neural network approximation. The PSF successfully keeps the system safe during so-called DAgger learning episodes [148] and shows minimal intervention after convergence to an approximately optimal control policy.

PREDICTIVE SAFETY FILTERS USING BAYESIAN MODEL ESTIMATES FOR SAFE QUADROTOR TUNING

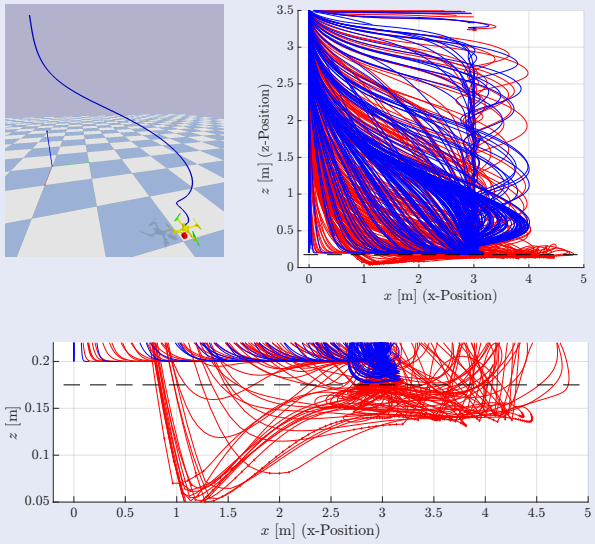


FIGURE S22 Safe Quadrotor Gain Tuning. **(Top-Left)** PyBullet quadrotor simulation, showing the optimal safe trajectory (blue line). **(Top-Right)** Learning episode trajectories without (red lines) and with (blue lines) the safety filter. **(Bottom)** Ground collisions are indicated with red squares.

In the second example [S2], we consider the AscTec Hummingbird drone, simulated in the Bullet Physics SDK [S5] as seen in Figure S22 (Top-left). A two-layer control structure enables position tracking, where the inner control loop takes pitch, roll, and vertical acceleration as input and commands motor torques. The outer controlled system model [S6] consists of states $x \in \mathbb{R}^{10}$, inputs $u \in \mathbb{R}^3$, and dynamics of the form $x(k+1) = \theta_{\text{true}}^\top \phi(x,u)$. The safety constraint is to stay above the ground, while the learning task is to efficiently tune an outer saturated PD controller to approach a specific landing position x_d, y_d, z_d. The outer PD controller takes the form

$$\pi_{\text{des}}(x;p,d) = \begin{cases} \text{clip}(p_{12}(x_d - x) + d_{12}\dot{x}, -1, 1), \\ \text{clip}(p_{12}(y_d - y) + d_{12}\dot{y}, -1, 1), \\ \text{clip}(p_3(z_d - z) + d_3\dot{z}, -1, 1), \end{cases}$$

where $\text{clip}(x, c_1, c_2) = \max(\min(x, c_2), c_1)$, with PD-controller gains $p_{12}, p_3 \in [0, 10]$, and $d_{12}, d_3 \in [-10, 0]$. A Bayesian optimization algorithm [S7] episodically adjusts the PD gains to minimize $|x_d - x| + |y_d - y| + |z_d - z| + 100\|\pi_{\text{des}}(x) - u^*_{0|k}\|$, where safety ensuring actions are largely penalized during the learning process. As depicted in Figure S22 (bottom), the direct application of the learning procedure results in ground crashes.

The learning-based safety filter model (58) is obtained from hovering data at a safe altitude and inferred using Gaussian process regression in a parametric fashion. The learning-based PSF of the form (61) was designed using $L = 0.999$ (based on an incremental stabilizability argument instead of Lipschitz continuity) with constraint tightening fraction $\epsilon = 0.01$. The confident subset constraint was designed to achieve constraint satisfaction with probability $p_s = 0.9$. The terminal set was formulated as a subset of the value function corresponding to a linear quadratic regulator for the hovering position using a linearization of (58). The Bayesian optimization PD tuning results with the safety filter are shown in Figure S22, where safety is ensured during all 240 learning episodes. The learned controller achieves good performance and does not require safety interventions after completion of learning.

REFERENCES

[S1] Tearle, B., Wabersich, K. P., Carron, A., Zeilinger, M. N. "A predictive safety filter for learning-based racing control," in *IEEE Robot. and Automat. Lett.*, pp. 7635–7642, 2021.

[S2] Wabersich, K. P., Zeilinger, M.N. "A predictive safety filter for learning-based control of constrained nonlinear dynamical systems," in *Automatica*, 2021.

[S3] Liniger, A., Domahidi, A., and Morari, M. "Optimization-based autonomous racing of 1:43 scale rc cars," in *Optim. Control Appl. and Methods*, pp. 628-647, 2015.

[S4] Rajamani, R. "Vehicle dynamics and control," in *Springer Science & Business Media*, 2011.

[S5] Coumans, E. and Bai, Y. "PyBullet, a Python module for physics simulation for games, robotics and machine learning," *technical report*, 2016.

[S6] Hu, H., Feng, X., Quirynen, R, Villanueva, M., and Houska, B. "Real-Time Tube MPC Applied to a 10-State Quadrotor Model," in *Proc. Amer. Control Conf.*, 2018.

[S7] Neumann-Brosig, M., Marco, A., Schwarzmann, D., and Timpe, S. "Data-Efficient Autotuning With Bayesian Optimization: An Industrial Control Study," in *IEEE Trans. Control Syst. Technol.*, 2019.

REFERENCES

[1] O. J. Ayamolowo, P. Manditereza, and K. Kusakana, "Exploring the gaps in renewable energy integration to grid," *Energy Reports*, vol. 6, pp. 992–999, 2020.

[2] S. Robla-Gómez, V. M. Becerra, J. R. Llata, E. González-Sarabia, C. Torre-Ferrero, and J. Pérez-Oria, "Working together: A review on safe human-robot collaboration in industrial environments," *IEEE Access*, vol. 5, pp. 26 754–26 773, 2017.

[3] E. Dassau, T. Hennings, J. Fazio, E. Atlas, and M. Phillip, "Closing the loop," *Diabetes Tech. & Therapeutics*, vol. 15, no. S1, pp. S–29–S–39, 2013.

[4] R. Hovorka, V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, T. Vering *et al.*, "Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes," *Physiological Measurement*, vol. 25, no. 4, pp. 905–920, 2004.

[5] P. Englert, N. A. Vien, and M. Toussaint, "Inverse kkt: Learning cost functions of manipulation tasks from demonstrations," *The Int. J. of Robotics Research*, vol. 36, no. 13-14, pp. 1474–1488, 2017.

[6] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, "Concrete problems in ai safety," *arXiv:1606.06565*, 2016.

[7] J.-P. Aubin, "A survey of viability theory," *SIAM J. on Control and Optimization*, vol. 28, no. 4, pp. 749–788, 1990.

[8] F. Blanchini and S. Miani, *Set-theoretic methods in control*. Springer, 2008, vol. 78.

[9] C. Tomlin, I. Mitchell, A. Bayen, and M. Oishi, "Computational techniques for the verification of hybrid systems," in *Proc. of the IEEE*, vol. 91, no. 7, 2003, pp. 986–1001.

[10] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin, "Hamilton-jacobi reachability: A brief overview and recent advances," in *Proc. IEEE 56th Conf. on Decision and Control (CDC)*, Melbourne, VIC, Australia, 2017, pp. 2242–2253.

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

[12] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *Proc. IEEE 18th European Control Conf. (ECC)*, Naples, Italy, 2019, pp. 3420–3431.

[13] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Nob Hill Publishing, 2017.

[14] K. P. Wabersich and M. N. Zeilinger, "Linear model predictive safety certification for learning-based control," in *Proc. IEEE 57th Conf. on Decision and Control (CDC)*, Miami, FL, USA, 2018, pp. 7130–7135.

[15] Y. Chen, M. Jankovic, M. Santillo, and A. D. Ames, "Backup control barrier functions: Formulation and comparative study," in *Proc. IEEE 60th Conf. on Decision and Control (CDC)*, Austin, TX, USA, 2021, pp. 6835–6841.

[16] J. J. Choi, D. Lee, K. Sreenath, C. J. Tomlin, and S. L. Herbert, "Robust control barrier–value functions for safety-critical control," in *Proc. IEEE 60th Conf. on Decision and Control (CDC)*, Austin, TX, USA, 2021, pp. 6814–6821.

[17] K. Leung, E. Schmerling, M. Zhang, M. Chen, J. Talbot, J. C. Gerdes, and M. Pavone, "On infusing reachability-based safety assurance within planning frameworks for human–robot vehicle interactions," *The Int. J. of Robotics Research*, vol. 39, no. 10-11, pp. 1326–1345, 2020.

[18] J. Zeng, B. Zhang, and K. Sreenath, "Safety-critical model predictive control with discrete-time control barrier function," in *Proc. IEEE American Control Conf. (ACC)*, New Orleans, LA, USA, 2021, pp. 3882–3889.

[19] K. P. Wabersich and M. N. Zeilinger, "Predictive control barrier functions: Enhanced safety mechanisms for learning-based control," *IEEE Trans. on Automatic Control*, pp. 1–1, 2022.

[20] U. Rosolia, A. Singletary, and A. D. Ames, "Unified multi-rate control: From low-level actuation to high-level planning," *IEEE Trans. on Automatic Control*, 2022.

[21] A. Wigren, J. Wågberg, F. Lindsten, A. G. Wills, and T. B. Schön, "Nonlinear system identification: Learning while respecting physical models using a sequential monte carlo method," *IEEE Control Sys.*, vol. 42, no. 1, pp. 75–102, 2022.

[22] 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," *IEEE Trans. on Automatic Control*, vol. 64, no. 7, pp. 2737–2752, 2019.

[23] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger, "Learning-based model predictive control: Toward safe learning in control," *Ann. Rev. Control, Robotics, and Autonomous Sys.*, vol. 3, pp. 269–296, 2020.

[24] A. J. Taylor, A. Singletary, Y. Yue, and A. D. Ames, "Learning for safety-critical control with control barrier functions," *Proc. of Machine Learning Research (PMLR)*, vol. 120, pp. 708–717, 2020.

[25] R. Bellman, *Dynamic Programming*. Princeton university press, 1957.

[26] M. Bardi, I. C. Dolcetta *et al.*, *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*. Springer, 1997, vol. 12.

[27] C. Tomlin, J. Lygeros, and S. Sastry, "Synthesizing controllers for nonlinear hybrid systems," in *Int. Work. on Hybrid Sys.: Computation and Control*. Springer, 1998, pp. 360–373.

[28] J. Lygeros, C. Tomlin, and S. Sastry, "Controllers for reachability specifications for hybrid systems," *Automatica*, vol. 35, no. 3, pp. 349–370, 1999.

[29] J. Lygeros, "On reachability and minimum cost optimal control," *Automatica*, vol. 40, no. 6, pp. 917–927, 2004.

[30] S. Prajna, "Barrier certificates for nonlinear model validation," *Automatica*, vol. 42, no. 1, pp. 117–126, 2006.

[31] M. Krstic and M. Bement, "Nonovershooting control of strict-feedback nonlinear systems," *IEEE Trans. on Automatic Control*, vol. 51, no. 12, pp. 1938–1943, 2006.

[32] A. Ames, J. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *Proc. IEEE 53rd Conf. on Decision & Control (CDC)*, Los Angeles, CA, USA, 2014, pp. 6271–6278.

[33] H. Chen and F. Allgöwer, "A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability," *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.

[34] L. Grüne and J. Pannek, *Nonlinear model predictive control*. Springer, 2017.

[35] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games," *IEEE Trans. on Automatic Control*, vol. 50, no. 7, pp. 947–957, 2005.

[36] J. F. Fisac, M. Chen, C. J. Tomlin, and S. S. Sastry, "Reach-avoid problems with time-varying dynamics, targets and constraints," in *Proc. of the 18th Int. Conf. on Hybrid Sys.: Computation and Control (HSCC)*, Seattle, WA, USA, 2015, pp. 11–20.

[37] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2017.

[38] R. Konda, A. D. Ames, and S. Coogan, "Characterizing safety: Minimal control barrier functions from scalar comparison systems," *IEEE Control Sys. Let.*, vol. 5, no. 2, pp. 523–528, 2020.

[39] S. Kolathaya and A. D. Ames, "Input-to-state safety with control barrier functions," *IEEE Control Sys. Let.*, vol. 3, no. 1, pp. 108–113, 2018.

[40] M. Jankovic, "Robust control barrier functions for constrained stabilization of nonlinear systems," *Automatica*, vol. 96, pp. 359–367, 2018.

[41] T. Gurriet, M. Mote, A. D. Ames, and E. Feron, "An online approach to active set invariance," in *Proc. IEEE 57th Conf. on Decision & Control (CDC)*, Miami, FL, USA, 2018, pp. 3592–3599.

[42] T. Mannucci, E. J. van Kampen, C. de Visser, and Q. Chu, "Safe exploration algorithms for reinforcement learning controllers," *IEEE Trans. on Neural Networks and Learning Sys.*, vol. 29, no. 4, pp. 1069–1081, 2017.

[43] O. Bastani, "Safe reinforcement learning with nonlinear dynamics via model predictive shielding," in *Proc. IEEE American Control Conf. (ACC)*, New Orleans, LA, USA, 2021, pp. 3488–3494.

[44] A. K. Akametalu, J. F. Fisac, J. H. Gillula, S. Kaynama, M. N. Zeilinger, and C. J. Tomlin, "Reachability-based safe learning with gaussian processes," in *Proc. IEEE 53rd Conf. on Decision and Control (CDC)*, Los Angeles, CA, USA, 2014, pp. 1424–1431.

[45] S. Herbert, J. J. Choi, S. Sanjeev, M. Gibson, K. Sreenath, and C. J. Tomlin, "Scalable learning of safety guarantees for autonomous systems using hamilton-jacobi reachability," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Xi'an, China, 2021, pp. 5914–5920.

[46] P. Jagtap, G. J. Pappas, and M. Zamani, "Control barrier functions for unknown nonlinear systems using gaussian processes," in *Proc. IEEE 59th Conf. on Decision and Control (CDC)*, Jeju, South Korea, 2020, pp. 3699–3704.

[47] J. Choi, F. Castañeda, C. J. Tomlin, and K. Sreenath, "Reinforcement

learning for safety-critical control under model uncertainty, using control lyapunov functions and control barrier functions," in *Proc. Robotics: Science and Sys. (RSS)*, Bend, OR, USA, 2020.

[48] C. Folkestad, Y. Chen, A. D. Ames, and J. W. Burdick, "Data-driven safety-critical control: Synthesizing control barrier functions with koopman operators," *IEEE Control Sys. Let.*, vol. 5, no. 6, pp. 2012–2017, 2020.

[49] M. J. Khojasteh, V. Dhiman, M. Franceschetti, and N. Atanasov, "Probabilistic safety constraints for learned high relative degree system dynamics," in *Proc. of Machine Learning Research (PMLR)*, 2020, pp. 781–792.

[50] 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," *Proc. of Machine Learning Research (PMLR)*, vol. 144, pp. 1041–1053, 2021.

[51] A. J. Taylor, V. D. Dorobantu, S. Dean, B. Recht, Y. Yue, and A. D. Ames, "Towards robust data-driven control synthesis for nonlinear systems with actuation uncertainty," in *Proc. IEEE 60th Conf. on Decision and Control (CDC)*, Austin, TX, USA, 2021, pp. 6469–6476.

[52] F. Castañeda, J. J. Choi, B. Zhang, C. J. Tomlin, and K. Sreenath, "Gaussian process-based min-norm stabilizing controller for control-affine systems with uncertain input effects and dynamics," in *Proc. IEEE American Control Conference (ACC)*, New Orleans, LA, USA, 2021, pp. 3683–3690.

[53] V. Dhiman, M. J. Khojasteh, M. Franceschetti, and N. Atanasov, "Control barriers in bayesian learning of system dynamics," *IEEE Trans. on Automatic Control*, pp. 1–1, 2021.

[54] Y. Emam, P. Glotfelter, S. Wilson, G. Notomista, and M. Egerstedt, "Data-driven robust barrier functions for safe, long-term operation," *IEEE Trans. on Robotics*, pp. 1–1, 2021.

[55] K. P. Wabersich and M. N. Zeilinger, "A predictive safety filter for learning-based control of constrained nonlinear dynamical systems," *Automatica*, vol. 129, p. 109597, 2021.

[56] A. Didier, K. P. Wabersich, and M. N. Zeilinger, "Adaptive model predictive safety certification for learning-based control," in *Proc. IEEE 60th Conf. on Decision and Control (CDC)*, Austin, TX, USA, 2021, pp. 809–815.

[57] S. Muntwiler, K. P. Wabersich, A. Carron, and M. N. Zeilinger, "Distributed model predictive safety certification for learning-based control," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 5258–5265, 2020.

[58] K. P. Wabersich, L. Hewing, A. Carron, and M. N. Zeilinger, "Probabilistic model predictive safety certification for learning-based control," *IEEE Trans. on Automatic Control*, vol. 67, no. 1, pp. 176–188, 2021.

[59] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based model predictive control for safe exploration," in *Proc. 57th IEEE Conf. Decision and Control (CDC)*, Miami, FL, USA, 2018, pp. 6059–6066.

[60] L. Brunke, M. Greeff, A. W. Hall, Z. Yuan, S. Zhou, J. Panerati, and A. P. Schoellig, "Safe learning in robotics: From learning-based control to safe reinforcement learning," *Ann. Rev. Control, Robotics, and Autonomous Sys.*, vol. 5, pp. 411–444, 2021.

[61] M. Nagumo, "Über die lage der integralkurven gewöhnlicher differentialgleichungen," *Proc. of the Physico-Mathematical Society of Japan. 3rd Series*, vol. 24, pp. 551–559, 1942.

[62] D. Seto, B. Krogh, L. Sha, and A. Chutinan, "The simplex architecture for safe online control system upgrades," in *Proc. of IEEE American Control Conf. (ACC)*, vol. 6, Philadelphia, PA, USA, 1998, pp. 3504–3508.

[63] C. J. Tomlin, J. Lygeros, and S. S. Sastry, "A game theoretic approach to controller design for hybrid systems," *Proc. of the IEEE*, vol. 88, no. 7, pp. 949–970, 2000.

[64] C. Tomlin, G. Pappas, and S. Sastry, "Conflict resolution for air traffic management: a study in multiagent hybrid systems," *IEEE Trans. on Automatic Control*, vol. 43, no. 4, pp. 509–521, 1998.

[65] A. K. Akametalu, S. Ghosh, J. F. Fisac, and C. J. Tomlin, "A minimum discounted reward hamilton-jacobi formulation for computing reachable sets," *arXiv:1809.00706*, 2018.

[66] B. Xue, Q. Wang, N. Zhan, M. Fränzle, and S. Feng, "Reach-avoid differential games based on invariant generation," *arXiv:1811.03215*, 2018.

[67] M. G. Crandall, L. C. Evans, and P.-L. Lions, "Some properties of viscosity solutions of hamilton-jacobi equations," *Trans. of the American Mathematical Society*, vol. 282, no. 2, pp. 487–502, 1984.

[68] I. M. Mitchell and J. A. Templeton, "A toolbox of hamilton-jacobi solvers for analysis of nondeterministic continuous and hybrid systems," in *Int. Work. on Hybrid Sys.: Computation and Control*. Springer, 2005, pp. 480–494.

[69] J. A. Sethian, *Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science*. Cambridge university press, 1999, vol. 3.

[70] M. Chen, S. L. Herbert, M. S. Vashishtha, S. Bansal, and C. J. Tomlin, "Decomposition of reachable sets and tubes for a class of nonlinear systems," *IEEE Trans. on Automatic Control*, vol. 63, no. 11, pp. 3675–3688, 2018.

[71] S. L. Herbert, S. Bansal, S. Ghosh, and C. J. Tomlin, "Reachability-based safety guarantees using efficient initializations," in *Proc. IEEE 58th Conf. on Decision and Control (CDC)*, Nice, France, 2019, pp. 4810–4816.

[72] S. Bansal and C. J. Tomlin, "Deepreach: A deep learning approach to high-dimensional reachability," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, Xi'an, China, 2021, pp. 1817–1824.

[73] J. F. Fisac, N. F. Lugovoy, V. Rubies-Royo, S. Ghosh, and C. J. Tomlin, "Bridging hamilton-jacobi safety analysis and reinforcement learning," in *Int. Conf. on Robotics and Automation (ICRA)*, Montreal, QC, Canada, 2019, pp. 8550–8556.

[74] S. Singh, M. Chen, S. L. Herbert, C. J. Tomlin, and M. Pavone, "Robust tracking with model mismatch for fast and safe planning: an sos optimization approach," in *Int. Work. on the Algorithmic Foundations of Robotics*. Springer, 2018, pp. 545–564.

[75] S. Kousik, S. Vaskov, F. Bu, M. Johnson-Roberson, and R. Vasudevan, "Bridging the gap between safety and real-time performance in receding-horizon trajectory design for mobile robots," *The Int. J. of Robotics Research*, vol. 39, no. 12, pp. 1419–1469, 2020.

[76] I. Hwang, D. M. Stipanović, and C. J. Tomlin, "Polytopic approximations of reachable sets applied to linear dynamic games and a class of nonlinear systems," in *Advances in control, communication networks, and transportation systems*. Springer, 2005, pp. 3–19.

[77] A. B. Kurzhanski and P. Varaiya, "On ellipsoidal techniques for reachability analysis. part i: external approximations," *Optimization methods and software*, vol. 17, no. 2, pp. 177–206, 2002.

[78] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," *IEEE Trans. on Robotics*, vol. 30, no. 4, pp. 903–918, 2014.

[79] S. Kousik, P. Holmes, and R. Vasudevan, "Safe, aggressive quadrotor flight via reachability-based trajectory design," in *Proc. ASME Dynamic Sys. and Control Conf. (DSCC)*, vol. 59162, Park City, UT, USA, 2019, p. V003T19A010.

[80] H. K. Khalil and J. W. Grizzle, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice Hall, 2002, vol. 3.

[81] A. V. Fiacco and G. P. McCormick, *Nonlinear programming: sequential unconstrained minimization techniques*. SIAM, 1990.

[82] E. D. Sontag, "A 'universal'construction of artstein's theorem on nonlinear stabilization," *Sys. & Control Let.*, vol. 13, no. 2, pp. 117–123, 1989.

[83] X. Xu, T. Waters, D. Pickem, P. Glotfelter, M. Egerstedt, P. Tabuada, J. W. Grizzle, and A. D. Ames, "Realizing simultaneous lane keeping and adaptive speed regulation on accessible mobile robot testbeds," in *Proc. IEEE Conf. on Control Tech. and App. (CCTA)*, Kohala Coast, HI, USA, 2017, pp. 1769–1775.

[84] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Trans. on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.

[85] L. Wang, E. A. Theodorou, and M. Egerstedt, "Safe learning of quadrotor dynamics using barrier certificates," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia, 2018, pp. 2460–2465.

[86] A. Singletary, W. Guffey, T. G. Molnar, R. Sinnet, and A. D. Ames, "Safety-critical manipulation for collision-free food preparation," *arXiv:2205.01026*, 2022.

[87] W. S. Cortez, D. Oetomo, C. Manzie, and P. Choong, "Control barrier functions for mechanical systems: Theory and application to robotic grasping," *IEEE Trans. on Control Sys. Tech.*, vol. 29, no. 2, pp. 530–545, 2019.

[88] R. Grandia, A. J. Taylor, A. D. Ames, and M. Hutter, "Multi-layered safety for legged robots via control barrier functions and model predictive control," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Xi'an, China, 2021, pp. 8352–8358.

[89] A. Alan, A. J. Taylor, C. R. He, A. D. Ames, and G. Orosz, "Control

barrier functions and input-to-state safety with application to automated vehicles," *arXiv:2206.03568*, 2022.

[90] L. Wang, D. Han, and M. Egerstedt, "Permissive barrier certificates for safe stabilization using sum-of-squares," in *Proc. IEEE American Control Conf. (ACC)*, Milwaukee, WI, USA, 2018, pp. 585–590.

[91] A. Clark, "Verification and synthesis of control barrier functions," in *Proc. IEEE 60th Conf. on Decision and Control (CDC)*, Austin, TX, USA, 2021, pp. 6105–6112.

[92] T. G. Molnar, R. K. Cosner, A. W. Singletary, W. Ubellacker, and A. D. Ames, "Model-free safety-critical control for robotic systems," *IEEE Robotics and Automation Let.*, vol. 7, no. 2, pp. 944–951, 2021.

[93] A. J. Taylor, P. Ong, T. G. Molnar, and A. D. Ames, "Safe backstepping with control barrier functions," *arXiv:2204.00653*, 2022.

[94] A. Robey, H. Hu, L. Lindemann, H. Zhang, D. V. Dimarogonas, S. Tu, and N. Matni, "Learning control barrier functions from expert demonstrations," in *Proc. IEEE 59th Conf. on Decision and Control (CDC)*, Jeju, South Korea, 2020, pp. 3717–3724.

[95] A. Mesbah, K. P. Wabersich, A. P. Schoellig, M. N. Zeilinger, S. Lucia, T. A. Badgwell, and J. A. Paulson, "Fusion of machine learning and mpc under uncertainty: What advances are on the horizon?" in *Proc. IEEE American Control Conf. (ACC)*, Atlanta, GA, USA, 2022, pp. 342–357.

[96] A. Agrawal and K. Sreenath, "Discrete control barrier functions for safety-critical control of discrete systems with application to bipedal robot navigation." in *Proc. Robotics: Science and Sys. (RSS)*, vol. 13, Cambridge, MA, USA, 2017.

[97] M. Lazar and M. Tetteroo, "Computation of terminal costs and sets for discrete–time nonlinear mpc," *IFAC-PapersOnLine*, vol. 51, no. 20, pp. 141–146, 2018.

[98] C. Conte, C. N. Jones, M. Morari, and M. N. Zeilinger, "Distributed synthesis and stability of cooperative distributed model predictive control for linear systems," *Automatica*, vol. 69, pp. 117–125, 2016.

[99] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[100] F. D. Brunner, M. Lazar, and F. Allgöwer, "Stabilizing linear model predictive control: On the enlargement of the terminal set," in *Proc. IEEE European Control Conf.*, Zürich, Switzerland, 2013, pp. 511–517.

[101] U. Rosolia and F. Borrelli, "Learning model predictive control for iterative tasks. a data-driven control framework," *IEEE Trans. on Automatic Control*, vol. 63, no. 7, pp. 1883–1896, 2017.

[102] E. C. Kerrigan and J. M. Maciejowski, "Soft constraints and exact penalty functions in model predictive control," in *Proc. Control Conf.*, Cambridge, United Kingdom, 2000, pp. 2319–2327.

[103] M. N. Zeilinger, M. Morari, and C. N. Jones, "Soft constrained model predictive control with robust stability guarantees," *IEEE Trans. on Automatic Control*, vol. 59, no. 5, pp. 1190–1202, 2014.

[104] C. Feller and C. Ebenbauer, "Relaxed logarithmic barrier function based model predictive control of linear systems," *IEEE Trans. on Automatic Control*, vol. 62, no. 3, pp. 1223–1238, 2016.

[105] 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 *Proc. ACM/IEEE 9th Int. Conf. on Cyber-Physical Sys. (ICCPS)*, Porto, Portugal, 2018, pp. 98–106.

[106] S. Tonkens and S. Herbert, "Refining control barrier functions through hamilton-jacobi reachability," *arXiv:2204.12507*, 2022.

[107] C. Dawson, Z. Qin, S. Gao, and C. Fan, "Safe nonlinear control using robust neural lyapunov-barrier functions," *Proc. of Machine Learning Research (PMLR)*, vol. 164, pp. 1724–1735, 2022.

[108] Z. Qin, D. Sun, and C. Fan, "Sablas: Learning safe control for black-box dynamical systems," *IEEE Robotics and Automation Let.*, vol. 7, no. 2, pp. 1928–1935, 2022.

[109] P. Glotfelter, J. Cortés, and M. Egerstedt, "Nonsmooth barrier functions with applications to multi-robot systems," *IEEE Control Sys. Let.*, vol. 1, no. 2, pp. 310–315, 2017.

[110] D. P. Bertsekas, *Dynamic programming and optimal control: Approximate dynamic programming*, 4th ed. Athena Scientific Belmont, MA, 2012, vol. 2.

[111] D. Lee and C. J. Tomlin, "Hamilton-jacobi equations for two classes of state-constrained zero-sum games," *arXiv:2106.15006*, 2021.

[112] J. Zeng, Z. Li, and K. Sreenath, "Enhancing feasibility and safety of nonlinear model predictive control with discrete-time control barrier functions," in *IEEE 60th Conf. on Decision and Control (CDC)*, Austin, TX, USA, 2021, pp. 6137–6144.

[113] M. Davoodi, J. M. Cloud, A. Iqbal, W. J. Beksi, and N. R. Gans, "Safe human-robot coetaneousness through model predictive control barrier functions and motion distributions," *IFAC-PapersOnLine*, vol. 54, no. 20, pp. 271–277, 2021.

[114] A. Thirugnanam, J. Zeng, and K. Sreenath, "Safety-critical control and planning for obstacle avoidance between polytopes with control barrier functions," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, Philadelphia, PA, USA, 2022.

[115] U. Rosolia and A. D. Ames, "Multi-rate control design leveraging control barrier functions and model predictive control policies," *IEEE Control Sys. Let.*, vol. 5, no. 3, pp. 1007–1012, 2021.

[116] J. Breeden and D. Panagou, "Predictive control barrier functions for online safety critical control," *arXiv:2204.00208*, 2022.

[117] S. Brüggemann, D. Steeves, and M. Krstic, "Simultaneous lane-keeping and obstacle avoidance by combining model predictive control and control barrier functions," *arXiv:2204.06136*, 2022.

[118] A. Singletary, P. Nilsson, T. Gurriet, and A. D. Ames, "Online active safety for robotic manipulators," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Sys. (IROS)*, Macau, China, 2019, pp. 173–178.

[119] T. Gurriet, M. Mote, A. Singletary, P. Nilsson, E. Feron, and A. D. Ames, "A scalable safety critical control framework for nonlinear systems," *IEEE Access*, vol. 8, pp. 187 249–187 275, 2020.

[120] A. J. Taylor and A. D. Ames, "Adaptive safety with control barrier functions," in *Proc. IEEE American Control Conf. (ACC)*, Denver, CO, USA, 2020, pp. 1399–1405.

[121] B. T. Lopez, J. E. Slotine, and J. P. How, "Robust adaptive control barrier functions: An adaptive and data-driven approach to safety," *IEEE Control Sys. Let.*, vol. 5, no. 3, pp. 1031–1036, 2021.

[122] A. Mesbah, "Stochastic model predictive control with active uncertainty learning: A survey on dual control," *Ann. Rev. in Control*, vol. 45, pp. 107–117, 2018.

[123] E. Arcari, L. Hewing, M. Schlichting, and M. Zeilinger, "Dual stochastic mpc for systems with parametric and structural uncertainty," *Proc. of Machine Learning Research (PMLR)*, vol. 120, pp. 894–903, 2020.

[124] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer, 2009, vol. 2.

[125] B. Tearle, K. P. Wabersich, A. Carron, and M. N. Zeilinger, "A predictive safety filter for learning-based racing control," *IEEE Robotics and Automation Let.*, vol. 6, no. 4, pp. 7635–7642, 2021.

[126] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," in *Proc. Int. Conf. on Learning Representations (ICLR)*, Vancouver, BC, Canada, 2018, pp. 1–18.

[127] G. Shi, X. Shi, M. O'Connell, R. Yu, K. Azizzadenesheli, A. Anandkumar, Y. Yue, and S.-J. Chung, "Neural lander: Stable drone landing control using learned dynamics," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Montreal, QC, Canada, 2019, pp. 9784–9790.

[128] M. Milanese and C. Novara, "Set membership identification of nonlinear systems," *Automatica*, vol. 40, no. 6, pp. 957–975, 2004.

[129] M. Chen and C. J. Tomlin, "Hamilton–jacobi reachability: Some recent theoretical advances and applications in unmanned airspace management," *Ann. Rev. of Control, Robotics, and Autonomous Sys.*, vol. 1, no. 1, pp. 333–358, 2018.

[130] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust optimization*. Princeton university press, 2009.

[131] F. Berkenkamp, R. Moriconi, A. P. Schoellig, and A. Krause, "Safe learning of regions of attraction for uncertain, nonlinear systems with gaussian processes," in *Proc. IEEE 55th Conf. on Decision & Control (CDC)*, Las Vegas, NV, USA, 2016, pp. 4661–4666.

[132] T. Beckers, D. Kulić, and S. Hirche, "Stable gaussian process based tracking control of euler–lagrange systems," *Automatica*, vol. 103, pp. 390–397, 2019.

[133] A. Lederer, J. Umlauft, and S. Hirche, "Uniform error bounds for gaussian process regression with application to safe control," *Advances in Neural Information Processing Sys.*, vol. 32, 2019.

[134] K. P. Wabersich and M. N. Zeilinger, "Nonlinear learning-based model predictive control supporting state and input dependent model uncertainty estimates," *Int. J. Robust and Nonlinear Control*, vol. 31, no. 18, pp. 8897–8915, 2021.

[135] 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," in *Proc. IEEE 60th Conf. on Decision and Control (CDC)*, Austin, TX, USA, 2021, pp. 6762–6769.

[136] Y. S. Shao, C. Chen, S. Kousik, and R. Vasudevan, "Reachability-based trajectory safeguard (rts): A safe and fast reinforcement learning safety layer for continuous control," *IEEE Robotics and Automation Let.*, vol. 6, no. 2, pp. 3663–3670, 2021.

[137] L. C. Evans and P. E. Souganidis, "Differential games and representation formulas for solutions of hamilton-jacobi-isaacs equations," *Indiana University mathematics journal*, vol. 33, no. 5, pp. 773–797, 1984.

[138] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. The MIT Press, 2006.

[139] A. J. Taylor, A. Singletary, Y. Yue, and A. D. Ames, "A control barrier perspective on episodic learning via projection-to-state safety," *IEEE Control Sys. Let.*, vol. 5, no. 3, pp. 1019–1024, 2021.

[140] 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 Sys. Let.*, vol. 6, pp. 908–913, 2022.

[141] S. Muntwiler, K. P. Wabersich, L. Hewing, and M. N. Zeilinger, "Data-driven distributed stochastic model predictive control with closed-loop chance constraint satisfaction," in *Proc. IEEE European Control Conf. (ECC)*, Delft, Netherlands, 2021, pp. 210–215.

[142] D. L. Marruedo, T. Alamo, and E. Camacho, "Input-to-state stable mpc for constrained discrete-time nonlinear systems with bounded additive uncertainties," in *Proc. IEEE 41st Conf. Decision and Control (CDC)*, vol. 4, Las Vegas, NV, USA, 2002, pp. 4619–4624.

[143] J. Köhler, M. A. Müller, and F. Allgöwer, "A novel constraint tightening approach for nonlinear robust model predictive control," in *Proc. IEEE American Control Conf. (ACC)*, Milwaukee, WI, USA, 2018, pp. 728–734.

[144] M. Chen, S. L. Herbert, H. Hu, Y. Pu, J. F. Fisac, S. Bansal, S. Han, and C. J. Tomlin, "Fastrack:a modular framework for real-time motion planning and guaranteed safe tracking," *IEEE Trans. on Automatic Control*, vol. 66, no. 12, pp. 5861–5876, 2021.

[145] A. Singletary, A. Swann, Y. Chen, and A. D. Ames, "Onboard safety guarantees for racing drones: High-speed geofencing with control barrier functions," *IEEE Robotics and Automation Let.*, vol. 7, no. 2, pp. 2897–2904, 2022.

[146] W. Chu and Z. Ghahramani, "Preference learning with gaussian processes," in *Proc. Int. Conf. on Machine Learning (ICML)*, Bonn, Germany, 2005, pp. 137–144.

[147] 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," *Proc. of Machine Learning Research (PMLR)*, vol. 168, pp. 1020–1033, 2022.

[148] S. Ross, G. J. Gordon, and J. A. Bagnell, "No-regret reductions for imitation learning and structured prediction," in *Proc. 13th Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, Sardinia, Italy, 2010, pp. 661–668.