# A General Safety Framework for Learning-Based Control in Uncertain Robotic Systems

Jaime F. Fisac, Anayo K. Akametalu, Melanie N. Zeilinger, Shahab Kaynama, Jeremy Gillula, and Claire J. Tomlin

***Abstract*—The proven efficacy of learning-based control schemes strongly motivates their application to robotic systems operating in the physical world. However, guaranteeing correct operation during the learning process is currently an unresolved issue, which is of vital importance in safety-critical systems. We propose a general safety framework based on Hamilton–Jacobi reachability methods that can work in conjunction with an arbitrary learning algorithm. The method exploits approximate knowledge of the system dynamics to guarantee constraint satisfaction while minimally interfering with the learning process. We further introduce a Bayesian mechanism that refines the safety analysis as the system acquires new evidence, reducing initial conservativeness when appropriate while strengthening guarantees through real-time validation. The result is a least-restrictive, safety-preserving control law that intervenes only when the computed safety guarantees require it, or confidence in the computed guarantees decays in light of new observations. We prove theoretical safety guarantees combining probabilistic and worst-case analysis and demonstrate the proposed framework experimentally on a quadrotor vehicle. Even though safety analysis is based on a simple point-mass model, the quadrotor successfully arrives at a suitable controller by policy-gradient reinforcement learning without ever crashing, and safely retracts away from a strong external disturbance introduced during flight.**

Manuscript received June 3, 2017; revised February 13, 2018; accepted August 24, 2018. Date of publication October 16, 2018; date of current version June 26, 2019. This work was supported in part by the NSF CPS project ActionWebs under Grant 0931843, in part by the NSF CPS project FORCES under Grant 1239166, in part by the ONR under the HUNT, SMARTS and Embedded Humans MURIs, and in part by the AFOSR under the CHASE MURI. The work of J. F. Fisac was supported by the "la Caixa" Foundation. The work of A. K. Akametalu was supported by the NSF Bridge to Doctorate program. The work of M. N. Zeilinger was supported by the EU FP7 (FP7/2007–2013) under Grant PIOFGA-2011-301436-COGENT. The authors thank Roy Dong for his helpful insights on measurability in function spaces. *(Jaime F. Fisac and Anayo K. Akametalu contributed equally to this work.)* Recommended by Associate Editor S. S. Saab. *(Corresponding author: Jaime F. Fisac.)*

J. F. Fisac, A. K. Akametalu, and C. J. Tomlin are with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, Berkeley, CA 94720 USA (e-mail: jfisac@eecs.berkeley.edu; kakametalu@eecs.berkeley.edu; tomlin@eecs.berkeley.edu).

M. N. Zeilinger is with the Department of Mechanical and Process Engineering, ETH Zurich, Zürich 8092, Switzerland (e-mail: mzeilinger@ethz.ch).

S. Kaynama is with the Apple Inc., Cupertino, CA 95104 USA (e-mail: skaynama@apple.com).

J. Gillula is with the Electronic Frontier Foundation, San Francisco, CA 94109 USA (e-mail: jeremy@eff.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2018.2876389

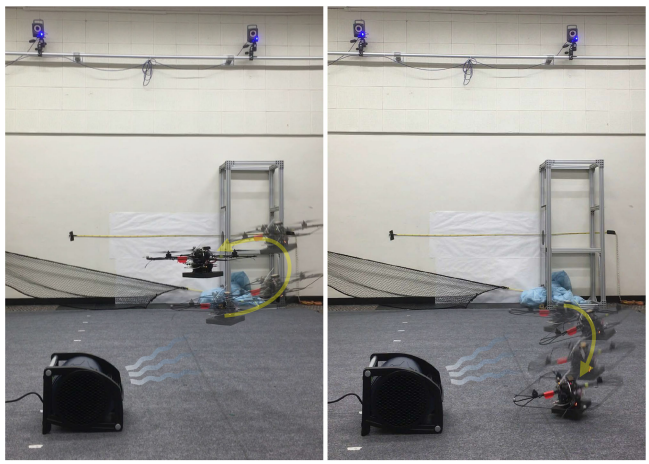

Fig. 1. Hummingbird quadrotor learning a vertical flight policy under the requirement of not colliding. When the fan is turned ON, the system experiences an unmodeled disturbance that it has not previously encountered. This can lead to a ground collision even under robust safety policies (right). The proposed Bayesian validation method detects the inconsistency and prevents the vehicle from entering the uncertain region (left). Video: https://youtu.be/WAAxyeSk2bw.

***Index Terms*—Safety, robot learning, autonomous systems, robust optimal control, Gaussian processes.**

## I. Introduction

LEARNING-BASED methods in control and artificial intelligence are generating a considerable amount of excitement in the research community. The auspicious results of deep reinforcement learning schemes in virtual environments, such as arcade videogames [1] and physics simulators [2], make these techniques extremely attractive for robotics applications, in which complex dynamics and hard-to-model environments limit the effectiveness of purely model-based approaches. However, the difficulty of interpreting the inner workings of many machine learning algorithms (notably in the case of deep neural networks), makes it challenging to make meaningful statements about the behavior of a system during the learning process, especially while the system has not yet converged to a suitable control policy. While this may not be a critical issue in a simulated reality, it can quickly become a limiting factor when attempting to put such an algorithm in control of a system in the physical world (such as the vehicle in Fig. 1),

where certain failures, such as collisions, can result in damage that would severely hinder or even terminate the learning process, in addition to material loss or human injury. We refer to systems in which certain failure states are unacceptable as *safety-critical*.

In the last decade, learning-based control schemes have been successfully demonstrated in robotics applications in which the safety-critical aspects were effectively removed or mitigated, typically by providing a manual fallback mechanism or retrofitting the environment to allow safe failure. In [3] and [4], a trained pilot was able to remotely take over control of the autonomous helicopter at any time; the power slide car maneuvers in [5] were performed on an empty test track; and the aerobatic quadrotor in [6] was enclosed in a safety net. While mostly effective, these *ad hoc* methods tend to come with their own issues (pilot handoffs, for instance, are notoriously prone to result in accidents [7]) and do not generalize well beyond the context of the particular demonstration. It therefore seems imperative to develop principled and provably correct approaches to safety, attuned to the exploration-intense needs of learning-based algorithms that can be built into the autonomous operation of learning robotic systems.

Current efforts in policy transfer learning propose training an initial control policy in simulation and then carrying it over to the physical system [8]. While progress in this direction is likely to reduce overall training time, it does not eliminate the risk of catastrophic system misbehavior. State-of-the-art neural network policies have been shown to be vulnerable to small changes between training and testing conditions [9], which inevitably arise between simulated and real systems. Guaranteeing correct behavior of simulation-trained schemes in the real world thus remains an important unsolved problem.

Providing guarantees about a system's evolution inevitably requires some form of knowledge about the causal mechanisms that govern it. Fortunately, in practice, it is never the case that the designer of a robotic system has no knowledge whatsoever of its dynamics: making use of approximate knowledge is both possible and, we argue, advantageous for safety. Yet, perfect knowledge of the dynamics can hardly if ever be safely assumed either. This motivates searching for points of rapprochement between data-driven and model-based techniques.

We identify following three key properties that we believe any general safe learning framework should satisfy.

1) *High confidence:* The framework should be able to keep the system safe with high probability given the available knowledge about the system and the environment.
2) *Modularity:* The framework should work in conjunction with an arbitrary learning-based control algorithm, without requiring modifications to said algorithm.
3) *Minimal intervention:* The framework should not interfere with the learning process unless deemed strictly necessary to ensure safety, and should return control to the learning algorithm as soon as possible.

We can use these criteria to evaluate the strengths and shortcomings of existing approaches to safety in intelligent systems, and place our work in the context of prior research.

### A. Related Work

Early proposals of safe learning date back to the turn of the century. Lyapunov-based reinforcement learning [10] allowed a learning agent to switch between a number of precomputed "base-level" controllers with desirable safety and performance properties; this enabled solid theoretical guarantees at the expense of substantially constraining the agent's behavior; in a similar spirit, later work has considered constraining policy search to the space of stabilizing controllers [11].

In risk-sensitive reinforcement learning [12], the expected return was heuristically weighted with the probability (risk) of reaching an "error state"; while this allowed for more general learning strategies, no guarantees could be derived from the heuristic effort. Nonetheless, the ideal problem formulation proposed in the paper, to maximize performance subject to some maximum allowable risk, inspired later work (see [13] for a survey) and is very much aligned with our own goals.

More recently, Moldovan and Abbeel [14] proposed an ergodicity-based safe exploration policy for Markov decision processes with uncertain transition measures, which imposed a constraint on the probability, under the current belief, of being able to return to the starting state. While practical online methods for updating the system's belief on the transition dynamics are not discussed, and the toy grid-world demonstrations fall short of capturing the criticality of dynamics in many real-world safety problems, the probabilistic safety analysis is extremely powerful, and our work certainly takes inspiration from it. Recent safe exploration efforts in robotics concurrent with our work use Gaussian processes to model uncertain dynamics, but restrict safety analysis to local stability (i.e., region of attraction) and do not consider state constraints [15], [16].

The robust model-predictive control approach in [17] learns about system dynamics for performance only, while enforcing constraints based on a robust nominal model. The method was successfully demonstrated on problems with nontrivial dynamics, including quadrotor flight. However, using an *a priori* model for safety at best constrains the system's ability to explore, and at worst may fail to keep the real system safe.

To explicitly account for model uncertainty, the safety problem can be studied as a differential game [18], in which the controller must keep the system within the specified state constraints (i.e., away from failure states) inspite of the actions of an adversarial disturbance: the optimal solution to this reachability game, obtainable through Hamilton–Jacobi methods [19], [20], has been used to guarantee safety in a variety of engineering problems [21]–[23]. This robust, worst-case analysis determines a safe region in the state space and a control policy to remain inside it; a related approach involves ensuring invariance through "barrier functions" [24], [25]. A key advantage is that in the interior of this *safe set* one can execute any desired action, as long as the safe control is applied at the boundary: in this sense, the technique yields a *least-restrictive* control law, which naturally lends itself to minimally constrained learning-based control. Initial work exploring this approach was presented in [26] and [27].

The above-mentioned methods are subject, however, to the fundamental limitation of any model-based safety analysis,

namely, the contingency of all guarantees upon the validity of the model. This faces designers with a difficult tradeoff. On the one hand, if, in order to guarantee safety under large or poorly understood uncertainty, they assume conservative bounds on model error, this will reduce the computed safe set, and thereby restrict the freedom of the learning algorithm. If, on the other hand, the assumed bounds fail to fully capture the true evolution of the state, the theoretical guarantees derived from the model may not in fact apply to the real system.

### B. Contribution

In this paper, we propose a novel general safety framework that combines model-based control-theoretical analysis with data-driven Bayesian inference to construct and maintain high-probability guarantees around an arbitrary learning-based control algorithm. Drawing on Hamilton–Jacobi robust optimal control techniques, it defines a *least-restrictive* supervisory control law, which allows the system to freely execute its learning-based policy almost everywhere, but imposes a computed action at states where it is deemed critical for safety. The safety analysis is refined through Bayesian inference in light of newly gathered evidence, both avoiding excessive conservativeness and improving reliability by rapidly imposing the computed safe actions if confidence in model-based guarantees decreases due to unexpected observations.

This paper consolidates the preliminary work presented in [26]–[28], extending the theoretical results to provide a unified treatment of model learning and guarantee validation, and presenting significant novel experimental results. To our knowledge, this is the first work in the area of reachability analysis that reasons online about the validity of computed guarantees and uses a resilient mechanism to continue exploiting them under inaccurate prior assumptions on model error.

Our framework relies on reachability analysis for the model-based safety guarantees, and on Gaussian processes for the online Bayesian analysis. It is important to acknowledge that both of these techniques are computationally intensive and scale poorly with the dimensionality of the underlying continuous spaces, which can generally limit their applicability to complex dynamical systems. However, recent compositional approaches have dramatically increased the tractability of lightly coupled high-dimensional systems [22], [29]–[31], while new analytic solutions entirely overcome the "curse of dimensionality" in some relevant cases [32], [33]. The key contribution of this paper is in the principled methodology for incorporating safety into learning-based systems: we thus focus our examples on problems of low dimensionality, implicitly bypassing the computational issues, and note that our method can readily be used in conjunction with these decomposition techniques to extend its application to more complex systems.

We demonstrate our method on a quadrotor vehicle learning to track a vertical trajectory close to the ground (see Fig. 1), using a policy gradient algorithm [34]. The reliability of our method is evidenced under uninformative policy initializations, inaccurate safe set estimation and strong unmodeled disturbances.

The remainder of the paper is organized as follows: In Section II, we introduce the modeling framework and formally state the safe learning problem. Section III presents the differential game analysis and derives some important properties. The proposed methodology is described in Section IV with the proofs of its fundamental guarantees, as well as a computationally tractable alternative with weaker, but practically useful, properties. Last, in Section V, we present the experimental results.

## II. Problem Formulation

### A. System Model and State-Dependent Uncertainty

The analysis in this paper considers a fully observable system whose underlying dynamics are assumed deterministic, but only *partially* known. This modeling framework can in practice be applied to a wide range of systems for which an approximate dynamic model is available but the exact behavior is hard to model *a priori* (due to manufacturing tolerances, aerodynamic effects, uncertain environments, etc.).

We can formalize this as a dynamical system with state $x \in \mathbb{R}^n$, and two inputs, $u \in \mathcal{U} \subset \mathbb{R}^{n_u}$, $d \in \mathcal{D} \subset \mathbb{R}^{n_d}$ (with $\mathcal{U}$ and $\mathcal{D}$ compact) which we will refer to as the *controller* and the *disturbance*

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

In this context, however, $d$ is thought of as a deterministic state-dependent disturbance capturing unmodeled dynamics, given by an unknown Lipschitz function $d : \mathbb{R}^n \to \mathcal{D}$. That is, we could in principle write the unknown dynamics as $F(x, u) = f(x, u, d(x))$. Unlike $F$, $f$ is a known function, with all uncertainty captured by $d(\cdot)$. The flow field $f : \mathbb{R}^n \times \mathcal{U} \times \mathcal{D} \to \mathbb{R}^n$ is assumed uniformly continuous and bounded, as well as Lipschitz in $x$ and $d$ for all $u$: this ensures that the unknown dynamics $F$ are Lipschitz in $x$.

Letting $\mathfrak{U}$ and $\mathfrak{D}$ denote the collections of measurable[1] functions $\boldsymbol{u} : [0, \infty) \to \mathcal{U}$ and $\boldsymbol{d} : [0, \infty) \to \mathcal{D}$, respectively, and allowing the controller and disturbance to choose any such signals, the evolution of the system from any initial state $x$ is determined (see, for example, [35], Ch. 2, Theorems 1.1 and 2.1) by the unique continuous trajectory $\xi : [0, \infty) \to \mathbb{R}^n$ solving

$$\begin{aligned} \dot{\xi}(s) &= f(\xi(s), \boldsymbol{u}(s), \boldsymbol{d}(s)), \text{ a.e. } s \geq 0 \\ \xi(0) &= x. \end{aligned} \tag{2}$$

Note that this is a solution in Carathéodory's *extended sense*, that is, it satisfies the differential equation *almost everywhere* (i.e., except on a subset of Lebesgue measure zero).

Since $d(x)$ is unknown, we attempt to bound it at each state by a compact set $\hat{\mathcal{D}}(x) \subseteq \mathcal{D}$, which is allowed to vary in the state space. In Section III, we present a robust, least-restrictive safety control law that enforces constraint satisfaction subject to $d(x) \in \hat{\mathcal{D}}(x)$. In Section IV, we present a Bayesian approach to find a bound $\hat{\mathcal{D}}(x)$ based on a Gaussian process model of $d(x)$. Our overall approach, therefore, combines robust (worst-case) guarantees with Bayesian (probabilistic) analysis by construct-

[1]A function $f : X \to Y$ between two measurable spaces $(X, \Sigma_X)$ and $(Y, \Sigma_Y)$ is said to be measurable if the preimage of a measurable set in $Y$ is a measurable set in $X$, that is: $\forall V \in \Sigma_Y$, $f^{-1}(V) \in \Sigma_X$, with $\Sigma_X$ and $\Sigma_Y$ $\sigma$-algebras on $X$ and $Y$.

ing the disturbance bound to reflect the local uncertainty around the inferred disturbance function.[2]

Any model-based safety guarantees for the system will require that the bound $\hat{\mathcal{D}}(x)$ correctly captures the unknown part of the dynamics given by $d(x)$, at least at some critical set of states $x$ (discussed in Section III). One key insight in this paper is that the system should take action to ensure safety not only when the model predicts that this action may be necessary, but also when the system detects that the model itself may become unreliable in the near future.

We state here a preliminary result that will be useful later on in the paper, and introduce the notion of local model reliability.

*Proposition 1:* If $d(x) \in \operatorname{int} \hat{\mathcal{D}}(x)$ and the set-valued map $\hat{\mathcal{D}} : \mathbb{R}^n \to 2^{\mathcal{D}}$ is Lipschitz-continuous under the Hausdorff metric,[3] then there exists $\Delta t > 0$ such that all possible trajectories followed by the system starting at $x$ will satisfy $d(\xi(\tau)) \in \hat{\mathcal{D}}(\xi(\tau))$ for all $\tau \in [t, t + \Delta t]$.

*Proof:* Let $L_{\hat{\mathcal{D}}}$ be the Lipschitz (Hausdorff) constant of $\hat{\mathcal{D}}$, $L_d$ the Lipschitz constant of $d$, and $C_f$ a norm bound on the dynamics $f$. We then have that over an arbitrary time interval $[t, t + \Delta t]$, regardless of the control and disturbance signals $\boldsymbol{u}(\cdot)$, $\boldsymbol{d}(\cdot)$, any system trajectory starting at $\xi(t) = x$ satisfies $|\xi(\tau) - x| \leq C_f \Delta t, \forall \tau \in [t, t + \Delta t]$. This implies both $|d(\xi(\tau)) - d(x)| \leq L_d C_f \Delta t$ and $d_H(\hat{\mathcal{D}}(\xi(\tau)), \hat{\mathcal{D}}(x)) \leq L_{\hat{\mathcal{D}}} \mathcal{C}_f \Delta t$. Requiring that the open ball $B(d(x), (L_d + L_{\hat{\mathcal{D}}}) C_f \Delta t)$ be contained in $\hat{\mathcal{D}}(x)$ ensures $d(\xi(\tau)) \in \hat{\mathcal{D}}(\xi(\tau))$. Since $d(x) \in \operatorname{int} \hat{\mathcal{D}}(x)$, there must exist a small enough $\Delta t > 0$ for which this condition is met. ∎

We can further quantify this $\Delta t$ through the signed distance[4] to $\hat{\mathcal{D}}(x)$ at the current $d(x)$, denoted $s_{\hat{\mathcal{D}}(x)}(d(x))$.

*Corollary 1:* If the Lipschitz constants are known, then $d(x) \in \operatorname{int} \hat{\mathcal{D}}(x)$ implies $d(\xi(\tau)) \in \hat{\mathcal{D}}(\xi(\tau))$ for all times $\tau \in [t, t + \Delta t]$, with

$$\Delta t = \frac{-s_{\hat{\mathcal{D}}(x)}\left(d(x)\right)}{(L_d + L_{\hat{\mathcal{D}}}) C_f}.$$

The disturbance bounds $\hat{\mathcal{D}}$ derived in this paper satisfy the hypothesis of Proposition 1 (see Appendix for details), and we thus refer to the condition $d(x) \in \operatorname{int} \hat{\mathcal{D}}(x)$ as the model being *locally reliable* at $x$.

Finally, we assume that the effect of the disturbance on the dynamics is independent of the action applied by the controller

$$\dot{x} = f(x, u, d(x)) = g(x, u) + h(d(x)) \tag{3}$$

with $g : \mathbb{R}^n \times \mathbb{R}^{n_u} \to \mathbb{R}^n, h : \mathbb{R}^{n_p} \to \mathbb{R}^n$, where $g$ and $h$ inherit Lipschitz continuity in their first argument from $f$, and $h$ is injective onto its image. This decoupling assumption, made for ease of exposition, is not strictly necessary, and the theoretical results in this paper can be easily adapted to the coupled case.

Throughout our analysis, we will use the notation $\xi_{x,\hat{\mathcal{D}}}^{\boldsymbol{u},\boldsymbol{d}}(\cdot)$ to denote the state trajectory $t \mapsto x$ corresponding to the initial condition $x \in \mathbb{R}^n$, the control signal $\boldsymbol{u} \in \mathfrak{U}$ and the disturbance signal $\boldsymbol{d} \in \mathfrak{D}$, subjecting the latter to satisfy $\boldsymbol{d}(t) \in \hat{\mathcal{D}}(\xi_{x,\hat{\mathcal{D}}}^{\boldsymbol{u},\boldsymbol{d}}(t))$ for all $t \geq 0$.

### B. State Constraints

A central element in our problem is the *constraint set*, which defines a region $\mathcal{K} \subseteq \mathbb{R}^n$ of the state space, typically the complement of all unacceptable failure states, where the system is required to remain throughout the learning process. This set is assumed closed and time-invariant; no further assumptions (boundedness, connectedness, convexity, etc.) are needed.

From closedness, we can implicitly characterize $\mathcal{K}$ as the zero superlevel set of a Lipschitz *surface function* $l : \mathbb{R}^n \to \mathbb{R}$:

$$x \in \mathcal{K} \iff l(x) \geq 0. \tag{4}$$

This function always exists, since we can simply choose $l(x) = -s_{\mathcal{K}}(x)$, which is Lipschitz continuous by definition.

To express whether a given trajectory *ever* violates the constraints, let the functional $\mathcal{V} : \mathbb{R}^n \times \mathfrak{U} \times \mathfrak{D} \to \mathbb{R}$ assign to each initial state $x$ and input signals $\boldsymbol{u}(\cdot)$, $\boldsymbol{d}(\cdot)$ the lowest value of $l(\cdot)$ achieved by trajectory $\xi_{x,\hat{\mathcal{D}}}^{\boldsymbol{u},\boldsymbol{d}}(\cdot)$ over all times $t \geq 0$:

$$\mathcal{V}(x, \boldsymbol{u}(\cdot), \boldsymbol{d}(\cdot)) := \inf_{t \geq 0} l\left(\xi_{x,\hat{\mathcal{D}}}^{\boldsymbol{u},\boldsymbol{d}}(t)\right). \tag{5}$$

This outcome $\mathcal{V}$ will be strictly smaller than zero if there exists any $t \in [0, \infty)$ at which the trajectory leaves the constraint set, and will be nonnegative if the system remains in the constraint set for all of $t \geq 0$. Denoting $\mathcal{V}^{\boldsymbol{u},\boldsymbol{d}}(x) = \mathcal{V}(x, \boldsymbol{u}(\cdot), \boldsymbol{d}(\cdot))$, the following statement follows from (4) and (5) by construction.

*Proposition 2:* The set of points $x$ from which the system trajectory $\xi_{x,\hat{\mathcal{D}}}^{\boldsymbol{u},\boldsymbol{d}}(\cdot)$ under given inputs $\boldsymbol{u}(\cdot) \in \mathfrak{U}, \boldsymbol{d}(\cdot) \in \mathfrak{D}$ will remain in the constraint set $\mathcal{K}$ at all times $t \geq 0$ is equal to the zero superlevel set of $\mathcal{V}^{\boldsymbol{u},\boldsymbol{d}}(\cdot)$:

$$\{x \in \mathbb{R}^n : \forall t \geq 0,\ \xi_{x,\hat{\mathcal{D}}}^{\boldsymbol{u},\boldsymbol{d}}(t) \in \mathcal{K}\} = \{x \in \mathbb{R}^n : \mathcal{V}^{\boldsymbol{u},\boldsymbol{d}}(\cdot) \geq 0\}.$$

Guaranteeing safe evolution from a given point $x \in \mathbb{R}^n$ given an uncertainty bound $\hat{\mathcal{D}}$ requires determining whether there exists a control input $\boldsymbol{u}(\cdot) \in \mathfrak{U}$ such that, for all disturbance inputs $\boldsymbol{d}(\cdot) \in \mathfrak{D}$ satisfying $\boldsymbol{d}(t) \in \hat{\mathcal{D}}(\xi_{x,\hat{\mathcal{D}}}^{\boldsymbol{u},\boldsymbol{d}}(t))$, the evolution of the system remains in $\mathcal{K}$, or equivalently $\mathcal{V}^{\boldsymbol{u},\boldsymbol{d}}(x) \geq 0$. In Section III, we review how to answer this question using differential game theory and state some important properties of the associated solution.

### C. Objective: Safe Learning

Learning-based control aims to achieve desirable system behavior by autonomously improving a policy $\kappa_l : \mathbb{R}^n \to \mathcal{U}$, typically seeking to optimize an objective function. Safe learning additionally requires that certain constraints $\mathcal{K}$ remain satisfied while searching for such a policy. With full knowledge of the system dynamics $F(x, u) = f(x, u, d(x))$, we would like to

[2] Alternative methods to providing the disturbance bounds (for example, a conservative *a priori* estimate, or a system identification procedure) are possible, provided some basic conditions to ensure that the dynamical system resulting from (2) with the restriction $d \in \hat{\mathcal{D}}(x)$ remains well defined. For the interested reader, sufficient conditions on $\hat{\mathcal{D}}(x)$ are discussed in the Appendix.

[3] The Hausdorff metric (or Hausdorff distance) between any two sets $A$ and $B$ in a metric space $(M, d_M)$ is defined as $d_H(A, B) = \max\{\sup_{a \in A} \inf_{b \in B} d_M(a, b), \sup_{b \in B} \inf_{a \in A} d_M(a, b)\}$.

[4] For any nonempty set $\mathcal{M} \subset \mathbb{R}^m$, the *signed distance function* $s_{\mathcal{M}} : \mathbb{R}^m \to \mathbb{R}$ is defined as $\inf_{y \in \mathcal{M}} |z - y|$ for points $z \in \mathbb{R}^m \setminus \mathcal{M}$ and $-\inf_{y \in \mathbb{R}^m \setminus \mathcal{M}} |z - y|$ for points $z \in \mathcal{M}$, where $|\cdot|$ denotes a norm on $\mathbb{R}^m$.

find a safe control policy $\kappa^* : \mathbb{R}^n \to \mathcal{U}$ producing trajectories $\xi(t) \in \mathcal{K} \quad \forall t \geq 0$, for the largest set of initial states $x = \xi(0)$, then restrict any learned policy so that $\kappa_l(x) = \kappa^*(x)$ wherever required to ensure safety. When $d(x)$ is not known exactly, however, this problem cannot be solved.

Instead, given an estimated disturbance set $\hat{\mathcal{D}}(x)$, we can find an inner approximation of the set of safe states by considering all the possible trajectories that can be produced under the bounded uncertainty $d(x) \in \hat{\mathcal{D}}(x)$. Our goal, then, is to find the set of *robustly safe* states $x$ for which there exists a control policy $\kappa^*$ that can keep the closed-loop system evolution in $\mathcal{K}$, and consistently limit $\kappa_l$ to ensure that $\kappa^*$ is applied when necessary. To formally state this, we introduce an important notion from robust control theory.

*Definition 1:* A subset $\mathcal{M} \subset \mathbb{R}^n$ is a *robust controlled invariant set* under uncertain dynamics $\dot{x} = f(x, u, d)$, $d \in \hat{\mathcal{D}}(x)$, if there exists a feedback control policy $\kappa : \mathbb{R}^n \to \mathcal{U}$ such that all possible system trajectories starting at $\xi(0) \in \mathcal{M}$ are guaranteed to satisfy $\xi(t) \in \mathcal{M}$ for all time $t \geq 0$.

Given that trajectories are continuous, the system state can only leave $\mathcal{M}$ by crossing its boundary $\partial\mathcal{M}$. Hence, if $\mathcal{M}$ is closed, applying the feedback policy $\kappa(x)$ for $x \in \partial\mathcal{M}$ is enough to render $\mathcal{M}$ robust controlled invariant, allowing an arbitrary control action to be applied in the interior of $\mathcal{M}$.

*Definition 2:* The *safe set* $\Omega_{\hat{\mathcal{D}}}$ is the maximal robust controlled invariant set under uncertain dynamics $\dot{x} = f(x, u, d)$, $d \in \hat{\mathcal{D}}(x)$, that is contained in the constraint set $\mathcal{K}$.

Success in safe learning therefore seems closely linked to model uncertainty: a tighter bound $\hat{\mathcal{D}}(x)$ on $d(x)$ yields a less conservative safe set $\Omega_{\hat{\mathcal{D}}}$, which in turn reduces the restrictions on the learning process. However, an estimated bound that fails to fully capture $d(x)$ may allow the system to execute control actions resulting in a constraint violation. The disturbance bound should thus be as tight as possible, to allow the system greater freedom in learning, yet wide enough to confidently capture the unknown dynamics, in order to ensure safety.

In the following two sections, we formalize this tradeoff and propose a framework to reason about safety guarantees under uncertainty. Section III poses the safety problem as a differential game between the controller and an adversarial disturbance, presenting a stronger result than commonly used in the reachability safety literature, which exploits the entire value function of the game rather than only its zero level set. Section IV leverages this result to provide a principled approach to global safety under model uncertainty, as well as a fast local alternative that may often be useful in practice.

## III. Safety as a Differential Game

The safety problem can be posed as a two-player zero-sum differential game between the system controller and the disturbance. Intuitively, we are requiring the controller to keep the system from violating the constraints for *all* possible disturbance inputs within a certain family: by conducting a worst-case analysis assuming an optimally adversarial disturbance, we implicitly protect the system against all "suboptimal" disturbances as well. We first introduce relevant background on differential games, and then present new enabling insights.

### A. Background: HJI Equation and Safe Set

To obtain a *safe set* and an associated *safety policy*, we formulate a game whose outcome is given by the functional $\mathcal{V}(x, \boldsymbol{u}(\cdot), \boldsymbol{d}(\cdot))$ introduced in (5), negative for those trajectories $\xi^{\boldsymbol{u},\boldsymbol{d}}_{x,\hat{\mathcal{D}}}(\cdot)$ that at some point violate the constraints $\mathcal{K}$.

In the robust safety problem, the controller seeks to maximize the outcome of the game, while the disturbance tries to minimize it: that is, the disturbance is trying to drive the system out of the constraint set, and the controller wants to prevent it from succeeding. Following [36], we define the set of *nonanticipative strategies* for the disturbance containing the functionals $\mathcal{B} = \{\boldsymbol{\beta} : \mathfrak{U} \to \mathfrak{D} \mid \forall t \geq 0 \quad \forall \boldsymbol{u}(\cdot), \hat{\boldsymbol{u}}(\cdot) \in \mathfrak{U}, (\boldsymbol{u}(\tau) = \hat{\boldsymbol{u}}(\tau) \text{ a.e.} \tau \geq 0) \Rightarrow (\boldsymbol{\beta}[\boldsymbol{u}](\tau) = \boldsymbol{\beta}[\hat{\boldsymbol{u}}](\tau) \text{ a.e.} \tau \geq 0)\}$. Since the disturbance and the control inputs are decoupled in the system dynamics, Isaacs' minimax condition holds[5] and the *value* of the game is well defined as follows:

$$V(x) := \inf_{\boldsymbol{\beta} \in \mathcal{B}} \sup_{\boldsymbol{u} \in \mathfrak{U}} \mathcal{V}(x, \boldsymbol{u}(\cdot), \boldsymbol{\beta}[\boldsymbol{u}](\cdot)). \tag{6}$$

Under this information structure, we draw on the (infinite-horizon) discriminating kernel concept from viability theory.

*Definition 3:* A point $x \in \mathcal{K}$ is in $\mathcal{K}$'s *discriminating kernel* $Disc_{\hat{\mathcal{D}}}(\mathcal{K})$ if the system trajectory $\xi^{\boldsymbol{u},\boldsymbol{d}}_{x,\hat{\mathcal{D}}}$ starting at $x$, with both players acting optimally, remains in $\mathcal{K}$ for all time $t \geq 0$:

$$\text{Disc}_{\hat{\mathcal{D}}}(\mathcal{K}) := \{x \in \mathbb{R}^n : \forall \boldsymbol{\beta}(\cdot) \in \mathcal{B}, \exists \boldsymbol{u}(\cdot) \in \mathfrak{U} \quad \forall t \geq 0, \xi^{\boldsymbol{u},\boldsymbol{\beta}[\boldsymbol{u}]}_{x,\hat{\mathcal{D}}}(t) \in \mathcal{K}\}.$$

The following classical result follows from Proposition 2.

*Proposition 3:* The discriminating kernel of the constraint set $\mathcal{K}$ is the zero superlevel set of the value function $V$:

$$\text{Disc}_{\hat{\mathcal{D}}}(\mathcal{K}) = \{x \in \mathbb{R}^n : V(x) \geq 0\}.$$

Further, from Definitions 2 and 3, it can be seen that the discriminating kernel $\text{Disc}_{\hat{\mathcal{D}}}(\mathcal{K})$ is identical to the safe set $\Omega_{\hat{\mathcal{D}}}$.

It has been shown that the value function for *minimum payoff* games of the form presented in Section III-A (i.e., games in which the payoff is the minimum of a state function over time) can be characterized as the unique viscosity solution to a variational inequality involving an appropriate Hamiltonian [37], [38]; an alternative formulation involves a modified partial differential equation [18]. In a finite-horizon setting, with the game taking place over the compact time interval $[0, T]$, the value function $V(x, t)$ can be computed by solving the Hamilton–Jacobi–Isaacs (HJI) variational inequality

$$0 = \min \left\{ l(x) - V(x,t), \frac{\partial V}{\partial t}(x,t) + \max_{u \in \mathcal{U}} \min_{d \in \hat{\mathcal{D}}(x)} \frac{\partial V}{\partial x}(x,t) f(x,u,d) \right\} \tag{7a}$$

$$V(x, T) = l(x). \tag{7b}$$

As long as there exists a nonempty safe set in the problem, $V(x, t)$ becomes independent of $t$ inside of this set as $T \to \infty$.

[5]This means that we could have alternatively let the controller use nonanticipative strategies, without affecting the solution of the game.

We accordingly drop the dependence on $t$ and recover $V(x)$ as defined in (6), which we refer to as the *safety function*.

*Definition 4:* The *optimal safe policy* $\kappa^*(\cdot)$ is the solution to the optimization[6]

$$\kappa^*(x) = \arg\max_{u\in\mathcal{U}} \min_{d\in\hat{\mathcal{D}}(x)} \frac{\partial V}{\partial x}(x) f(x,u,d).$$

Policy $\kappa^*(x)$ attempts to drive the system to the safest possible state always assuming an adversarial disturbance. If the disturbance bound $\hat{\mathcal{D}}(x)$ is correct everywhere, then one can allow the system to execute any desired control while in the interior of $\Omega_{\hat{\mathcal{D}}}$, as long as the safety-preserving action $\kappa^*(x)$ is taken whenever the state reaches the boundary $\partial\Omega_{\hat{\mathcal{D}}}$; the system is then guaranteed to remain inside $\Omega_{\hat{\mathcal{D}}}$ for all time. This least-restrictive control law can be used in conjunction with an arbitrary learning-based control policy $\kappa_l(x)$ (which may be repeatedly updated by the corresponding learning algorithm), to produce a *safe learning policy*

$$\kappa(x) = \begin{cases} \kappa_l(x), & \text{if } V(x) > 0 \\ \kappa^*(x), & \text{otherwise.} \end{cases} \tag{8}$$

Rather than imposing the optimal safe action $\kappa^*(x)$, it would have, in principle, been sufficient to project the desired $\kappa_l(x)$ onto the set of control inputs that guarantee nonnegative local evolution of $V$ for all $d \in \hat{\mathcal{D}}(x)$. However, $\kappa^*(x)$ results in the greatest predicted increase in value, which is desirable under model uncertainty, as we will see in Section IV-C.

### B. Invariance Properties of Level Sets

Traditionally, the implicit hypothesis made to guarantee safety using a least-restrictive law in the form of (8) has been correctness of the estimated disturbance bound $\hat{\mathcal{D}}$ everywhere in the state space, (i.e., $d(x) \in \hat{\mathcal{D}}(x)\ \forall x \in \mathbb{R}^n$), or at least everywhere in the constraint set $\mathcal{K}$ [18], [27]. We will now argue that the necessary hypothesis for safety is in fact much less stringent, by proving an important result that we will use in the following section to strengthen the proposed safety framework and retain safety guarantees under partially incorrect models.

*Proposition 4:* Any nonnegative superlevel set of $V(x)$ is a robust controlled invariant set with respect to $d \in \hat{\mathcal{D}}(x)$.

*Proof:* By Lipschitz continuity of $f$ and $l$, we have that $V$ is Lipschitz continuous [36] and hence, by Rademacher's theorem, almost everywhere differentiable. The convergence of $V(x,t)$ to $V(x)$ as $T \to \infty$ implies that at the limit $\frac{\partial V}{\partial t}(x,t) = 0$. Therefore, given any $\alpha \geq 0$, for any point $x \in \{x \mid V(x) \geq \alpha\}$ there must exist a control action $u^*$ such that $\forall d \in \hat{\mathcal{D}}(x)$, $\frac{\partial V}{\partial x}(x) f(x,u^*,d) \geq 0$; otherwise the right-hand side of (7a) would be strictly negative for $T \to \infty$, contradicting convergence. Then, the value of $V$ from any such state $x$ can always be kept from decreasing, so $\{x|V(x) \geq \alpha\}$ is a robust controlled invariant set with respect to $d \in \hat{\mathcal{D}}(x)$. ■

[6]While in general the solution need not be unique, we can always choose one element of the $\arg\max$ set arbitrarily. Therefore, we will assume for simplicity a policy $\kappa^* : \mathbb{R}^n \to \mathcal{U}$ uniquely mapping states to control inputs.

*Proposition 5:* Consider two disturbance sets $\mathcal{D}_1(x)$ and $\mathcal{D}_2(x)$, and a closed set $\mathcal{M} \subset \mathbb{R}^n$ that is robustly controlled invariant under $\mathcal{D}_1(x)$. If $\mathcal{D}_2(x) \subseteq \mathcal{D}_1(x)\ \forall x \in \partial\mathcal{M}$, then $\mathcal{M}$ is robustly controlled invariant also under $\mathcal{D}_2(x)$.

*Proof:* Consider an arbitrary trajectory $\xi^{\boldsymbol{u},\boldsymbol{d}}_{x_0,\mathcal{D}_2}$ under the disturbance set $\mathcal{D}_2(x)$, starting at $x_0 \in \mathcal{M}$, such that for some $\tau < \infty, \xi(\tau) \notin \mathcal{M}$. Since trajectories are continuous, there must then exist $s \in [t_0, \tau]$ such that $\xi^{\boldsymbol{u},\boldsymbol{d}}_{x_0,\mathcal{D}_2}(s) \in \partial\mathcal{M}$. On the other hand, because $\mathcal{M}$ is robustly controlled invariant under $\mathcal{D}_1(x)$, we know that $\exists \kappa : \mathbb{R}^n \to \mathcal{U}$ such that no possible disturbance $d \in \mathcal{D}_1(x)$ can drive the system out of $\mathcal{M}$. Since $\mathcal{D}_2(x) \subseteq \mathcal{D}_1(x)\ \forall x \in \partial\mathcal{M}$, the same control policy $\kappa^*(x)$ on the boundary guarantees that no disturbance $d \in \mathcal{D}_2(x) \subseteq \mathcal{D}_1(x)$ can drive the system out of $\mathcal{M}$. Hence, for $\xi^{\boldsymbol{u},\boldsymbol{d}}_{x_0,\mathcal{D}_2}$, switching to policy $\kappa$ at time $s$ guarantees that the system will remain in $\mathcal{M}$. Therefore, $\mathcal{M}$ is a robust controlled invariant set under $\mathcal{D}_2(x)$. ■

*Corollary 2:* Let $\mathcal{Q}_\alpha = \{x \in \mathbb{R}^n : V(x) = \alpha\}$ with $\alpha \geq 0$ be any nonnegative level set of the safety function $V$, computed for some disturbance set $\hat{\mathcal{D}}(x)$. If $d(x) \in \hat{\mathcal{D}}(x)\ \forall x \in \mathcal{Q}_\alpha$, then the superlevel set $\{x \in \mathbb{R}^n : V(x) \geq \alpha\}$ is an invariant set under the computed safe control policy $\kappa^*(x)$.

This corollary, which follows from Propositions 4 and 5 by considering the singleton $\{d(x)\}$, is an important result that will be at the core of our data-driven safety enhancement. It provides a sufficient condition for safety, but unlike the standard HJI solution, it does not readily prescribe a least-restrictive control law to exploit it: how should one determine what candidate $\alpha \geq 0$ to choose, or whether a valid $\mathcal{Q}_\alpha$ exists at all? Deciding when the safe controller should intervene and what guarantees are possible is nontrivial and requires additional analysis.

The next section proposes a Bayesian approach enabling the safety controller to reason about its confidence in the model-based guarantees described in this section. If this confidence reaches a prescribed minimum value in light of the observed data, the controller can intervene early to ensure that safety will be maintained with high probability.

## IV. Bayesian Safety Analysis

### A. Learning-Based Safe Learning

As we have seen, robust optimal control and dynamic game theory provide powerful analytical tools to study the safety of a dynamical model. However, it is important to realize that the applicability of any theoretically derived guarantee to the real system is contingent upon the validity of the underlying modeling assumptions; in the formulation considered here, this amounts to the state disturbance function $d(x)$ being captured by the bound $\hat{\mathcal{D}}(x)$ on at least a certain subset of the state space. The system designer therefore faces an inevitable tradeoff between risk and conservativeness, due to the impossibility of accounting for every aspect of the real system in a tractable model.

In many cases, choosing a parametric model *a priori* forces one to become overly conservative in order to ensure that the system behavior will be adequately captured: this results in a large bound $\hat{\mathcal{D}}(x)$ on the disturbance, which typically leads to

a small safe set $\Omega_{\hat{\mathcal{D}}}$, limiting the learning agent's ability to explore and perform the assigned tasks. In other cases, insufficient caution in the definition of the model can lead to an estimated disturbance set $\hat{\mathcal{D}}(x)$ that fails to contain the actual model error $d(x)$, and therefore the computed safe set $\Omega_{\hat{\mathcal{D}}}$ may not in fact be controlled invariant in practice, which can end all safety guarantees.

In order to avoid excessive conservativeness and keep theoretical guarantees valid, it is imperative to have both a principled method to refine the system model based on acquired measurements and a reliable mechanism to detect and react to model discrepancies with the real-system's behavior; both of these elements are necessarily data-driven. We, thus, arrive at what is perhaps the most important insight in this paper: *the relation between safety and learning is reciprocal*. Not only is safety a key requirement for learning in autonomous systems: learning about the real-system's behavior is itself indispensable to provide practical safety guarantees.

In the remainder of this section, we propose a method for reasoning about the uncertain system dynamics, using Gaussian processes to regularly update the model used for safety analysis, and introduce a Bayesian approach for online validation of model-based guarantees *in between* updates. We then define an adaptive safety control strategy based on this real-time validation, which leverages the theoretical results from Hamilton–Jacobi analysis to provide stronger guarantees for safe learning under possible model inaccuracies.

### B. Gaussian Process

To estimate the disturbance function $d(x)$ over the state space, we model it as being drawn from a Gaussian process. Gaussian processes are a powerful abstraction that extends multivariate Gaussian regression to the infinite-dimensional space of functions, allowing Bayesian inference based on (possibly noisy) observations of a function's value at finitely many points.[7]

A Gaussian process is a random process or field defined by a mean function $\mu : \mathbb{R}^n \to \mathbb{R}$ and a positive semidefinite covariance kernel function $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$. We will treat each component $d^j, j \in \{1, \dots, n_d\}$, of the disturbance function as an independent Gaussian process

$$d^j(x) \sim \mathcal{GP}(\mu^j(x), k^j(x, x')). \tag{9}$$

A defining characteristic of a Gaussian process is that the marginal probability distribution of the function value at any finite number of points is a multivariate Gaussian. This will allow us to obtain the disturbance bound $\hat{\mathcal{D}}(x)$ as a Cartesian product of confidence intervals for the components of $d(x)$ at each state $x$, choosing the bound to capture a desired degree of confidence.[8]

Gaussian processes allow incorporating new observations in a nonparametric Bayesian setting. First, assume a prior Gaussian process distribution over the $j$th component of $d(\cdot)$, with mean $\mu^j(\cdot)$ and covariance kernel $k^j(\cdot, \cdot)$. The class of the prior mean function and covariance kernel function is chosen to capture the characteristics of the model (linearity, periodicity, etc.), and is associated to a set of hyperparameters $\theta_p$. These are typically set to maximize the marginal likelihood of an available set of training data, or possibly to reflect some prior belief about the system.

Next, consider $N$ measurements $\hat{\mathbf{d}}^j = [\hat{d}_1^j, \dots, \hat{d}_N^j]$, observed with independent Gaussian noise $\epsilon_i^j \sim \mathcal{N}(0, (\sigma_n^j)^2)$ at the points $X = [x_1, \dots, x_N]$, i.e., $\hat{d}_i^j = d^j(x_i) + \epsilon_i^j$. Combined with the prior distribution (9), this new evidence induces a Gaussian process posterior; in particular, the value of $d^j$ at finitely many points $X_*$ is distributed as a multivariate normal

$$\mathbb{E}[d^j(X_*) \mid \hat{\mathbf{d}}^j, X] = \mu^j(X_*) + K^j(X_*, X)(K^j(X, X) + (\sigma_n^j)^2 I)^{-1}(\hat{\mathbf{d}}^j - \mu^j(X)) \tag{10a}$$

$$\operatorname{cov}[d^j(X_*) \mid X] = K^j(X_*, X_*) - K^j(X_*, X)(K^j(X, X) + (\sigma_n^j)^2 I)^{-1} K^j(X, X_*) \tag{10b}$$

where $d_i^j(X) = d^j(x_i)$, $\mu_i^j(X) = \mu^j(x_i)$, and for any $X, X'$ the matrix $K^j(X, X')$ is defined component-wise as $K_{ik}^j(X, X') = k^j(x_i, x_k')$. Note that whenever a new batch of data $X$ is obtained the hyperparameters of the kernel function are refitted, so the variance implicitly depends on the measurements $d^j$. If a single query point is considered, i.e., $X_* = \{x_*\}$, the marginalized Gaussian process posterior becomes a univariate normal distribution quantifying both the expected value of the disturbance function, $\bar{d^j}(x_*)$, and the uncertainty of this estimate, $(\sigma^j(x_*))^2$,

$$\bar{d^j}(x_*) = \mathbb{E}[d^j(x_*) | \hat{\mathbf{d}}^j, X] \tag{11a}$$

$$(\sigma^j(x_*))^2 = \operatorname{cov}[d^j(x_*) | X]. \tag{11b}$$

We can use the Bayesian machinery of Gaussian process regression to compute a *likely* bound $\hat{\mathcal{D}}(x)$ on the disturbance function $d(x)$ based on the history of commanded inputs $u_i$ and state measurements $x_i$, $i \in \{1, \dots, N\}$. To this effect, we assume that a method for approximately measuring the state derivatives is available (e.g., by numerical differentiation), and denote each of these measurements by $\hat{f}_i$. Based on (3), we can obtain measurements of $d(x_i)$ from the residuals between the observed dynamics and the model's prediction

$$\hat{d}(x_i) = h^{-1}\left(\hat{f}_i - g(x_i, u_i)\right). \tag{12}$$

The residuals $\hat{\mathbf{d}} = [\hat{d}(x_1), \dots, \hat{d}(x_N)]$ are processed through (10) to infer the marginal distribution of $d(x_*)$ for an arbitrary point $x_*$, specified by the expected value $\bar{d^j}(x_*)$ and the standard deviation $\sigma^j(x_*)$ of each component of the disturbance. This distribution can be used to construct a disturbance set $\hat{\mathcal{D}}(x_*) \subseteq \mathcal{D}$ at any point $x_*$; in practice, this will be done at finitely many points $x_{\mathfrak{i}}$ on a grid, and used in the numerical reachability computation to obtain the safety function and the safe control policy.

[7] We give here an overview of Gaussian process regression and direct the interested reader to [39] for a more comprehensive introduction.

[8] By assuming independence of disturbance components, we are effectively over-approximating the confidence ellipsoid in $\mathbb{R}^{n_d}$ by its minimal containing box; a less conservative analysis could compute $\hat{\mathcal{D}}(x)$ using a vector-valued Gaussian process model, at the expense of heavier computation.

We now introduce the design parameter $p$ as the desired marginal probability that the disturbance function $d(x)$ will belong to the bound $\hat{\mathcal{D}}(x)$ at each point $x$; typically, $p$ should be chosen to be close to 1. The set $\hat{\mathcal{D}}(x)$ is then chosen for each $x$ as follows. Let $z = \sqrt{2}\,\mathrm{erf}^{-1}(p^{1/n_d})$, where $\mathrm{erf}(\cdot)$ denotes the Gauss error function; that is, define $z$ so that the probability that a sample from a standard normal distribution $\mathcal{N}(0,1)$ lies within $[-z, z]$ is $p^{1/n_d}$. We construct $\hat{\mathcal{D}}(x)$ by taking a Cartesian product of confidence intervals

$$\hat{\mathcal{D}}(x) = \prod_{j=1}^{n_d} \left[\bar{d}^j(x) - z\sigma^j(x), \bar{d}^j(x) + z\sigma^j(x)\right]. \tag{13}$$

Since each component $d^j(x)$ is given by an independent Gaussian $\mathcal{N}(\bar{d}^j(x), \sigma^j(x))$, the probability of $d(x)$ lying within the above hyperrectangle is by construction $(p^{1/n_d})^{n_d} = p$.

*Remark 1:* It is commonplace to use Gaussian distributions to capture beliefs on variables that are otherwise known to be bounded. While one might object that the unbounded support of (11) contradicts our problem formulation (in which the disturbance $d$ took values from some compact set $\mathcal{D} \subset \mathbb{R}^{n_d}$), the hyperrectangle $\hat{\mathcal{D}}(x)$ in (13) is always a compact set. Note that the theoretical input set $\mathcal{D}$ is never needed in practice, so it can always be assumed to contain $\hat{\mathcal{D}}(x)$ for all $x$.

Under Lipschitz continuous prior means $\mu^j$ and covariance kernels $k^j$, the disturbance bound (13) varies (Hausdorff) Lipschitz-continuously in $x$, satisfying the hypotheses of Proposition 1. This is formalized and proved in the Appendix.

The safety analysis described in Section III can be carried out by solving the Hamilton–Jacobi equation (7) for $\hat{\mathcal{D}}(x)$ given by (13), which—based on the information available at the time of computation—will be a correct disturbance bound at any single state $x$ with probability $p$. As the system goes on to gather new information, however, the posterior probability for $d(x) \in \hat{\mathcal{D}}(x)$ will change at each $x$ (and will typically no longer equal $p$). More generally, we have the following result.

*Proposition 6:* Let $q$ be the probability that $d(x) \in \hat{\mathcal{D}}(x)$ for some state $x$ with $V(x) \geq 0$. Then, the probability that $L_f^* V(x) := D_x V(x) \cdot f(x, \kappa^*(x), d(x)) \geq 0$ is at least $q$.

*Proof:* Omitting $x$ for conciseness, we have: $P(L_f^* V \geq 0) = P(L_f^* V \geq 0 | d \in \hat{\mathcal{D}}) P(d \in \hat{\mathcal{D}}) + P(L_f^* V \geq 0 | d \notin \hat{\mathcal{D}}) P(d \notin \hat{\mathcal{D}})$.

By Corollary 2, the first term evaluates to $1 \cdot q$; the second term is nonnegative (and will typically be positive, since not all values of $d \notin \hat{\mathcal{D}}$ will be unfavorable for safety, and there may be some for which the input $\kappa^*(x)$ leads the system to locally increase $V$). ■

Based on this result, we can begin to reason about the guarantees of the reachability analysis applied to the real system in a Bayesian framework, inherited from the Gaussian process model.

### C. Online Safety Guarantee Validation

In order to ensure safety under the possibility of model mismatch with the real system, it may become necessary to intervene not only on the boundary of the computed safe set, but also whenever the observed evolution of the system indicates that the model-based safety guarantees may lose validity. Indeed, failure to take a safe action in time may lead to complete loss of guarantees if the system enters a region of the state space where the model is consistently incorrect.

While the estimated bound $\hat{\mathcal{D}}$ (Section IV-B) and the associated safety guarantees (Section III) should certainly be recomputed as frequently as possible in light of new evidence, this process can typically take seconds or minutes, and in some cases may even require offline computation. This motivates the need to augment model-based guarantees through an online data-driven mechanism to quickly adapt to new incoming information even as new, improved guarantees are computed.

Bayesian analysis allows the system to update its belief on the disturbance function as new observations are obtained. This in turn can be used to provide a probabilistic guarantee on the validity of the safety results obtained from the robust dynamical model generated from the older observations. In the remainder of this section, we will discuss how to update the belief on the disturbance function, and then provide two different theoretical criteria for safety intervention. The first criterion provides global probabilistic guarantees, but has computational challenges associated to its practical implementation. The alternative method only provides a local guarantee, but can more easily be applied in real time.

Let us denote $X_{\text{old}}$ and $\hat{\mathbf{d}}^{\mathbf{j}}_{\text{old}}$ as the evidence used in computing the disturbance set $\hat{\mathcal{D}}(x)$, and $X_{\text{new}}$ and $\hat{\mathbf{d}}^{\mathbf{j}}_{\text{new}}$ as the evidence acquired online after the disturbance set is computed. Conditioned on the old evidence, the function $d^j(x)$ is normally distributed with mean and variance given by (10) with $X = X_{\text{old}}$ and $\hat{\mathbf{d}}^{\mathbf{j}} = \hat{\mathbf{d}}^{\mathbf{j}}_{\text{old}}$, and the disturbance set is given by (13). If we also condition on the new evidence and keep the hyperparameters fixed, then the mean and variance are updated by modifying (10) with $X = [X_{\text{old}}, X_{\text{new}}]$ and $\hat{\mathbf{d}}^{\mathbf{j}} = [\hat{\mathbf{d}}^{\mathbf{j}}_{\text{old}}, \hat{\mathbf{d}}^{\mathbf{j}}_{\text{new}}]$.

*Remark 2:* Performing the update requires inverting $K^j([X_{\text{old}}, X_{\text{new}}], [X_{\text{old}}, X_{\text{new}}])$. This can be done efficiently employing standard techniques: since $K^j(X_{\text{old}}, X_{\text{old}})$ has already been inverted (in order to compute the disturbance bound $\hat{\mathcal{D}}$), all that is needed is inverting the Schur complement of $K^j(X_{\text{old}}, X_{\text{old}})$ in $K^j([X_{\text{old}}, X_{\text{new}}], [X_{\text{old}}, X_{\text{new}}])$, which has the same size as $K^j(X_{\text{new}}, X_{\text{new}})$.

Ideally, we would incorporate $X_{\text{new}}$ and $\hat{\mathbf{d}}^{\mathbf{j}}_{\text{new}}$ to relearn the Gaussian process hyperparameters as quickly as new measurements come in: otherwise, new measured disturbance values $\hat{\mathbf{d}}^{\mathbf{j}}_{\text{new}}$ will only affect the posterior mean, with the variance depending exclusively on where the measurements were made ($X_{\text{new}}$). However, performing this update online is computationally prohibitive. Instead, we update the hyperparameters every time a new estimated bound $\hat{\mathcal{D}}$ is produced for safety analysis, keeping them fixed in between. In practice, the set $X_{\text{old}}$ will be much larger than $X_{\text{new}}$, so the estimated hyperparameters would not be expected to change significantly.

*Remark 3:* In settings where conditions are slowly time-varying, it may be desirable to give recently observed data more weight than older observations. This can naturally be encoded by the Gaussian process by appending time as an additional dimension in $X$: points that are distant in time would then be more weakly correlated, analogous to space.

Based on the new Gaussian distribution, we can reason about the posterior confidence in the safety guarantees produced by our original safety analysis, which relied on the prior Gaussian distribution resulting from measurements $\hat{\mathbf{d}}^{\mathbf{j}}_{\text{old}}$ at states $X_{\text{old}}$.

***1) Global Bayesian Safety Analysis:*** The strongest result available for guaranteeing safety under the present framework is Corollary 2, which allows the system to exploit any superzero level set $\mathcal{Q}_\alpha$ ($\alpha \geq 0$) of the safety function $V$ throughout which the model is locally correct; all that is needed is for such a $\mathcal{Q}_\alpha$ to exist for $\alpha \in [0, V(x)]$ given the current state $x$.

It is possible to devise a safety policy to fully exploit the sufficient condition in Corollary 2 in a Bayesian setting: if the posterior probability that the corollary's hypotheses will hold drops to some arbitrary *global confidence threshold* $\gamma_0$, the safe controller can override the learning agent. With probability $\gamma_0$, the corollary will still apply, in which case the system is guaranteed to remain safe for all time; even if Corollary 2 does not apply at this time (which could happen with probability $1-\gamma_0$), it is still possible that the disturbance $d(x)$ will not consistently take adversarial values that force the computed safety function $V(x)$ to decrease, in which case the system may still evolve safely. Therefore, this policy guarantees a lower bound on the probability of maintaining safety for all time.

In order to apply this safety criterion, the system needs to maintain a Bayesian posterior of the sufficient condition in Corollary 2. We refer to this posterior probability as the *global safety confidence* $\gamma(x; X, \hat{\mathbf{d}}^{\mathbf{j}})$, or $\gamma(x)$ for conciseness

$$\gamma(x; X, \hat{\mathbf{d}}^{\mathbf{j}}) := P(\exists \alpha \in [0, V(x)] \\ \forall x \in \mathcal{Q}_\alpha : d(x) \in \hat{\mathcal{D}}(x) | X, \hat{\mathbf{d}}^{\mathbf{j}}). \quad (14)$$

Based on this, we propose the least-restrictive control law

$$\kappa(x) = \begin{cases} \kappa_l(x), & \text{if } (\gamma(x) > \gamma_0) \wedge (V(x) > 0) \\ \kappa^*(x), & \text{otherwise} \end{cases} \quad (15)$$

so the system applies any action it desires if the global safety confidence is above the threshold, but applies the safe controller once this is no longer the case.

Note that if confidence in the safety guarantees is restored after applying the safety action the learning algorithm will be allowed to resume control of the system. This can happen by multiple mechanisms: moving to a region with higher $V(x)$ will tend to increase the probability that *some* lower level set may satisfy the hypotheses of Corollary 2; moving to a region with less inconsistency between expected and observed dynamics will typically lead to higher posterior belief that *nearby* level sets will satisfy the hypotheses of Corollary 2; and generally acquiring new data may, in some cases, increase the posterior confidence that Corollary 2 may apply.

Computing the joint probability that the bound $\hat{\mathcal{D}}(x)$ captures the Gaussian process $d(x)$ *everywhere* on a level set $\mathcal{Q}_\alpha$ is not possible, since the set of functions $d(x)$ satisfying this condition is bounded on uncountably many dimensions, and thus not measurable in function space. Similarly, evaluating the joint probability for a continuum of level sets $\mathcal{Q}_\alpha$ for $\alpha \in [0, V(x)]$ is not feasible. Instead, exploiting the Lipschitz assumption on $d(x)$, we can obtain the sought probability $\gamma(x)$ from a marginal distribution over a sufficiently dense set of sample points on each $\mathcal{Q}_\alpha$ and a sufficiently dense collection of level sets between 0 and $V(x)$.

We can then use numerical methods [40] to compute the multivariate normal cumulative distribution function and estimate the marginal probability (using compact logic notation)

$$\gamma(x) \approx P\left(\bigvee_{s=1}^{S} \bigwedge_{i=1}^{I} d(x_{s,i}) \in \hat{\mathcal{D}}(x_{s,i})\right) \quad (16)$$

over $S$ level sets $0 = \alpha_0 < \cdots < \alpha_S = V(x)$ and $I$ sample points from each level set $Q_{\alpha_s}$. As the density of samples increases with larger $S$ and $I$, the marginal probability (16) asymptotically approaches the sought probability (14). Unfortunately, however, current numerical methods can only efficiently approximate these probabilities for multivariate Gaussians of about 25 dimensions [40], which drastically limits the number of sample points ($S \times I \approx 25$) that the marginal probability can be evaluated over, making it difficult to obtain a useful estimate. In view of this, a viable approach may be to bound (14) below as follows:

$$\gamma(x) \geq \underline{\gamma}(x) := \max_{\alpha \in [0, V(x)]} P(\forall x \in \mathcal{Q}_\alpha : d(x) \in \hat{\mathcal{D}}(x)) \quad (17)$$

and approximately compute this as

$$\underline{\gamma}(x) \approx \max_{s \in \{1, \ldots, S\}} P\left(\bigwedge_{i=1}^{I} d(x_{s,i}) \in \hat{\mathcal{D}}(x_{s,i})\right) \quad (18)$$

with the advantage that a separate multivariate Gaussian evaluation can be done now for each level set ($I \approx 25$). Computing this approximate probability as the system explores its state space provides a decision mechanism to guarantee safe operation of the system with a desired degree of confidence, which the system designer or operator can adjust through the $\gamma_0$ parameter.

***2) Local Bayesian Safety Analysis:*** Evaluating the expression in (18) is still computationally intensive, which can limit the practicality of this method for real-time validation of safety guarantees in some applications, such as mobile robots relying on onboard processing. An alternative is to replace the global safety analysis with a local criterion that offers much faster computation traded off with a weaker safety guarantee.

Instead of relying on Corollary 2, this lighter method exploits Propositions 1 and 6. The system is allowed to explore the computed safe set freely as long as the probability of the estimated model $\hat{\mathcal{D}}$ being *locally reliable* remains above a certain threshold $\lambda_0$; if this threshold is reached, the safe controller intervenes, and the system is guaranteed to locally maintain or increase the computed safety value $V(x)$ with probability no less than $\lambda_0$. While this local guarantee does not ensure safety globally, it does constitute a useful heuristic effort to prevent the system from entering unexplored and potentially unsafe regions of the state space. Further, although the method is not explicitly tracking the hypotheses of Corollary 2, the local result becomes a global guarantee if these hypotheses do indeed hold.

We define the *local safety confidence* $\lambda(x; X, \hat{\mathbf{d}}^{\mathbf{j}})$, more concisely $\lambda(x)$, as the posterior probability that $d(x)$ will be contained in $\hat{\mathcal{D}}(x)$ at the current state $x$, given all observations made until now

$$\lambda(x; X, \hat{\mathbf{d}}^{\mathbf{j}}) := P(d(x) \in \hat{\mathcal{D}}(x) \mid X, \hat{\mathbf{d}}^{j}). \tag{19}$$

We then have the following local safety certificate.

*Proposition 7:* Let the disturbance $d(\cdot)$ be distributed component-wise as $n_d$ independent Gaussian processes (9). The safety policy $\kappa^*(\cdot)$ is guaranteed to locally maintain or increase the system's computed safety $V(\cdot)$ with probability greater than or equal to the local safety confidence $\lambda(x)$.

*Proof:* The proof follows directly from Propositions 1 and 6, and the definition of $\lambda(x)$, noting that the boundary of $\hat{\mathcal{D}}(x)$ has zero Lebesgue measure and thus under any Gaussian distribution $P(d \in \operatorname{int} \hat{\mathcal{D}}(x) | d \in \hat{\mathcal{D}}(x)) = 1$. ■

A *local confidence threshold* $\lambda_0 \in (0, p)$ can be established such that whenever $\lambda(x) < \lambda_0$ the model is considered insufficiently reliable (reachability guarantees may fail locally with probability greater than $1 - \lambda_0$), and the safety control is applied. The proposed safety control strategy is, therefore, as follows:

$$\kappa(x) = \begin{cases} \kappa_l(x), & \text{if } (\lambda(x) > \lambda_0) \wedge (V(x) > 0) \\ \kappa^*(x), & \text{otherwise.} \end{cases} \tag{20}$$

Similarly to (15), under this control law, if confidence on the local reliability of the model is restored after applying the safe action and making new observations, the system will be allowed to resume its learning process, as long as it is in the interior of the computed safe set.

After generating a new Gaussian process model and defining $\hat{\mathcal{D}}(x)$ as described in Section IV-B, the prior probability with which the disturbance function $d(x)$ belongs to the set $\hat{\mathcal{D}}(x)$ is by design $p$ everywhere in the state space. As the system evolves, more evidence is gathered in the form of measurements of the disturbance along the system trajectory, so that the belief that $d(x) \in \hat{\mathcal{D}}(x)$ is updated for each $x$. In particular, in the Gaussian process model, this additional evidence amounts to augmenting the covariance matrix $K^j$ in (10) with additional data points and reevaluating the mean and variance of the posterior distribution of $d(x)$. Based on the new Gaussian distribution, $\lambda(x; X, \hat{\mathbf{d}}^{\mathbf{j}})$ can readily be evaluated for each $x$ as follows:

$$\lambda(x) = \prod_{j=1}^{n_d} \frac{1}{2} \left[ \operatorname{erf}\left( \frac{d_+^j(x) - m^j(x)}{s^j(x)\sqrt{2}} \right) - \operatorname{erf}\left( \frac{d_-^j(x) - m^j(x)}{s^j(x)\sqrt{2}} \right) \right] \tag{21}$$

with $d_+^j(x) = \bar{d}^j(x) + z\sigma^j(x)$, $d_-^j(x) = \bar{d}^j(x) - z\sigma^j(x)$, $m^j(x) = \mathbb{E}[d^j(x)|X, \hat{\mathbf{d}}^j]$, $s^j(x) = \sqrt{\operatorname{var}(d^j(x)|X)}$; recall that $z$ was defined to yield the desired probability mass $p$ in $\hat{\mathcal{D}}(x)$ at the time of safety computation, as per (13).

Parameters $p$ and $\lambda_0$ (or, in its case, $\gamma_0$) allow the system designer to choose the degree of conservativeness in the system: while $p$ regulates the amount of uncertainty accounted for by the robust model-based safety computation, $\lambda_0$ ($\gamma_0$) determines the acceptable degradation in the resulting certificate's posterior confidence before a safety intervention is initiated. A value of $p$ close to 1 will lead to a large, high-confidence $\hat{\mathcal{D}}(x)$ throughout the state space, but this analysis may result in a small or even empty safe set; on the other hand, if $p$ is low, $\hat{\mathcal{D}}(x)$ will be smaller and the computed safe set will be larger, but guarantees are more likely to be deemed unreliable (as per $\lambda_0$ or $\gamma_0$) in light of later observations.

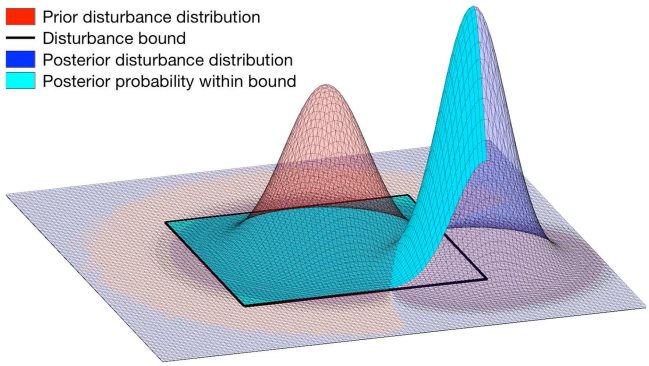


Fig. 2. Evolution of the probability distribution of the disturbance $d(x)$ at a particular state $x$. The prior distribution is used to compute the bound $\hat{\mathcal{D}}(x)$ using confidence intervals, such that it contains a specified probability mass $p$. As more data are obtained, the distribution may shift, leading to a different posterior probability mass contained within $\hat{\mathcal{D}}(x)$.

In the case of local safety analysis, immediately after computing a new model $\hat{\mathcal{D}}$, $\lambda(x)$ is by construction equal to $p$ everywhere in the state space. As more measurements are obtained, the posterior distribution over the disturbance changes, as illustrated in Fig. 2, which can result in $\lambda(x)$ locally increasing or decreasing. If $\lambda_0$ is chosen to be close to $p$, it is likely that the safety override will take place under minor deviations with respect to the model's prediction; as $\lambda_0$ becomes lower, however, the probability that the disturbance will violate the modeling assumptions before the safety controller intervenes increases. This reflects the fundamental tradeoff between risk and conservativeness in safety-critical decision making under uncertainty. The proposed framework, therefore, allows the system designer to adjust the degree of conservativeness according to the needs and characteristics of the system at hand.

## V. Experimental Results

We demonstrate our framework on a practical application with an autonomous quadrotor helicopter learning a flight controller in different scenarios. Our method is tested on the Stanford–Berkeley Testbed of Autonomous Rotorcraft for Multi-Agent Control (STARMAC), using Ascending Technologies Pelican and Hummingbird quadrotors (see Fig. 1). The system receives full state feedback from a VICON motion capture system. For the purpose of this series of experiments, the vehicle's dynamics are approximately decoupled through an onboard controller responsible for providing lateral stability around hover and vertical flight; our framework is then used to learn the feedback gains for a hybrid vertical flight controller. The learning and safety

controllers were implemented and executed in MATLAB, on a Lenovo Thinkpad with an Intel i7 processor that communicated wirelessly with the vehicle's 1.99 GHz Quadcore Intel Atom processor. This was all done using the Indigo version of the Robot Operating System (ROS) framework. Reachability computations are executed using the Level Set Toolbox [19], employing the Lax–Friedrich approximation for the numerical Hamiltonian; a weighted essentially nonoscillatory scheme for spatial derivatives; and a third-order total variation diminishing Runge–Kutta scheme for the time derivative [20], [41]. Once the safety function and safety policy have been computed, they are stored as lookup tables that can be quickly consulted in constant time.

The purpose of the results presented here is not to advance the state of the art of quadrotor flight control or reinforcement learning techniques, but to illustrate how the proposed method can allow safe execution of an arbitrary learning-based controller without requiring any particular convergence rate guarantees. To fully demonstrate the reliability of our safe learning framework, in our first setup, we let the vehicle begin its online learning in midair starting with a completely untrained controller. The general functioning of the framework can be observed in the second flight experiment, in which the vehicle starts with a conservative model and iteratively computes empirical estimates of the disturbance, gradually expanding its computed safe set while avoiding overreliance on poor predictions. Finally, we include an experiment in which an unexpected disturbance is introduced into the system. The vehicle reacts by immediately applying the safe action dictated by its local safety policy and retracting from the perturbed region, successfully maintaining safety throughout its trajectory. We show how the absence of online guarantee validation in the same scenario can result in loss of safety.

We use an affine dynamical model of quadrotor vertical flight, with state equations

$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= k_T u + g + k_0 + d(x)\end{aligned} \tag{22}$$

where $x_1$ is the vehicle's altitude, $x_2$ is its vertical velocity, and $u \in [0, 1]$ is the normalized motor thrust command. The gravitational acceleration is $g = -9.8$ m/s$^2$. The parameters of the affine model $k_T$ and $k_0$ are determined for the Pelican and the Hummingbird vehicles through a simple on-the-ground experimental procedure—a scale is used to measure the normal force reduction for different values of $u$. The state constraint $\mathcal{K} = \{x : 0 \text{ m} \leq x_1 \leq 2.8 \text{ m}\}$ encodes the position of the floor and the ceiling, which must be avoided. Finally, $d$ is an unknown, state-dependent scalar disturbance term representing unmodeled forces in the system. We learn $d(x)$ using a Gaussian process model, and generate $\hat{\mathcal{D}}(x)$ as the marginal 95% confidence interval at each $x$. We implement *local* Bayesian guarantee validation, conservatively approximating (21) by assuming $s^j(x) := \sqrt{\text{var}(d^j(x)|X)} \approx \sqrt{\text{var}(d^j(x)|X_{\text{old}})}$, that is, neglecting the (favorable but often small) reduction in uncertainty due to $X_{\text{new}}$. This was done for ease of prototyping.

As the learning-based controller, we choose an easily implementable policy gradient reinforcement learning algorithm [34], which learns the weights for a linear mapping from state features to control commands. Following [27], we define different features for positive and negative velocities and position errors, since the (unmodeled) rotor dynamics may be different in ascending and descending flight. This can be seen as the policy gradient algorithm learning the feedback gains for a hybrid proportional-integral-derivative controller.

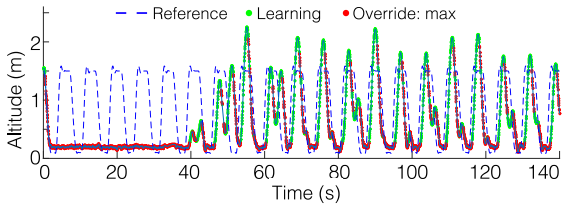


Fig. 3. Vehicle altitude and reference trajectory over time. Initial feedback gains are set to zero. When the learning controller (green/light) lets the vehicle drop, the safety control (red/dark) takes over preventing a collision. Within a few seconds, the learned feedback gains allow rough trajectory tracking and are subsequently tuned as the vehicle attempts to minimize error.

### A. From Fall to Flight

To demonstrate the strength of Hamilton–Jacobi-based guarantees for safely performing learning-based control on a physical system, we first require a Pelican quadrotor to learn an effective vertical trajectory tracking controller with an arbitrarily poor initialization. To do this, the policy gradient algorithm is initialized with all feature weights set to 0. The precomputed safety controller (numerically obtained using [19]) is based on a conservative uncertainty bound of $\pm 1.5$ m/s$^2$ everywhere in the state space; no new bounds are learned during this experiment. The reference trajectory requires the quadrotor to aggressively alternate between hovering at two altitudes, one of them (1.5 m) near the center of the room, the other (0.1 m) close to the floor.

This first experiment illustrates the interplay between the *learning controller* and the *safety policy*. The *iterative safety re-computation* and *Bayesian guarantee validation* components of the framework are not active here. Consistently, this demonstration uses (8) as the least-restrictive safety policy.

The experiment, shown in Fig. 3, is initialized with the vehicle in midair. Since all feature weights are initially set to zero, the vehicle's initial action is to enter free fall. However, as the quadrotor is accelerated by gravity toward the floor, the boundary of the computed safe set is reached, triggering the intervention of the safety controller, which automatically overrides the learning controller and commands the maximum available thrust to the motors ($u = 1$), causing the vehicle to decelerate and hover at a small distance from the ground. For the next few seconds, there is some chattering near the boundary of the safe set, and the policy gradient algorithm has some occasions to attempt to control the vehicle when it is momentarily pushed into the interior of the safe set. Initially, it has little success, which leads the safety controller to continually intervene to prevent the quadrotor from colliding with the floor; this has the undesirable effect of slowing down the learning process, since observations under this interference are uninformative about the behavior of

the vehicle when actually executing the commands produced by the learning controller (which is an "on-policy" algorithm). However, at approximately $t = 40$ s, the learning controller is able to make the vehicle ascend toward its tracking reference, retaining control of the vehicle for a longer span of time and accelerating the learning process. By $t = 60$ s, the quadrotor is approximately tracking the reference, with the safety controller only intervening during the aggressive descent phase of the repeated trajectory, to ensure (under its conservative model) that there is no risk of a ground collision. The controller continues to learn in subsequent iterations, overall improving its tracking accuracy.

The remarkable result in this experiment is not in the quality of the learned tracking controller after only a few seconds of active exploration (a merit that corresponds to the reinforcement learning method [34]), but the system's ability to achieve competent performance at its task from an extremely poor initial policy while remaining safe at all times.

### B. When in Doubt

In the second experiment, we demonstrate the iterative updating of the safe set and safety policy using observations of the system dynamics gathered over time, as well as the online validation of the resulting guarantees. All components of the framework are active during the test, namely *learning controller*, *safety policy*, *iterative safety re-computation*, and *Bayesian guarantee validation*, with the main focus being on the latter two.

Here, the Pelican quadrotor attempts to safely track the same reference trajectory, while using the gathered information about the system's evolution to refine its notion of safety. In this case, the policy gradient learning algorithm is initialized to a hand-tuned set of parameter values. The initial dynamic model available to the safety algorithm is identical to the one used in the previous experiment, with a uniform uncertainty bound of $\pm 1.5$ m/s$^2$. However, the system is now allowed to update this bound, throughout the state space, based on the disturbance posterior computed by a Gaussian process model.

To learn the disturbance function, the system starts with a Gaussian process prior over $d(\cdot)$ defined by a zero mean function and a squared exponential covariance function

$$k(x, x') = \sigma_f^2 \exp\left(\frac{(x - x')^T \mathcal{L}^{-1}(x - x')}{2}\right) \qquad (23)$$

where $\mathcal{L}$ is a diagonal matrix, with $\mathcal{L}_i$ as the $i$th diagonal element, and $\theta_p = [\sigma_f^2, \sigma_n^2, \mathcal{L}_1, \mathcal{L}_2]$ are the hyperparameters, $\sigma_f^2$ being the signal variance, $\sigma_n^2$ the measurement noise variance, and the $\mathcal{L}_i$ the squared exponential's characteristic length for position and velocity, respectively. The hyperparameters are chosen to maximize the marginal likelihood of the training data set, and are recomputed for each new batch of data when a new disturbance model $\hat{\mathcal{D}}(x)$ is generated for safety analysis. Finally, the chosen prior mean and covariance kernel classes are both Lipschitz continuous, ensuring that all required technical conditions for the theoretical results hold (proofs are presented in the Appendix).

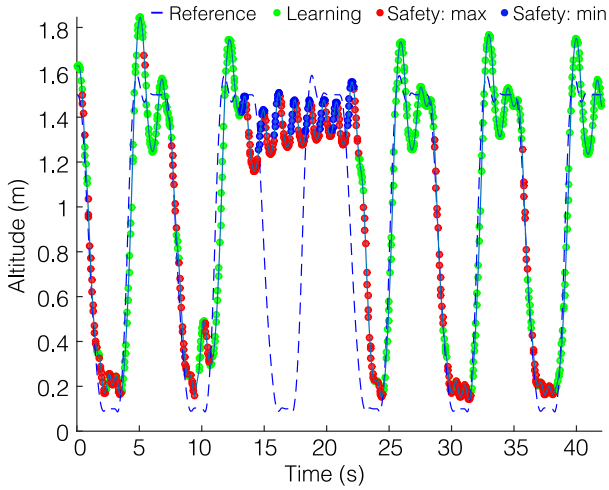


Fig. 4. Vehicle altitude and reference trajectory over time. After flying with an initial conservative model, the vehicle computes a first Gaussian process model of the disturbance with only a few data points, resulting in an insufficiently accurate bound. The safety policy detects the low confidence and refuses to follow the reference to low altitudes. Once a more accurate disturbance bound is computed, tracking is resumed, with a less restrictive safe set than the original one.

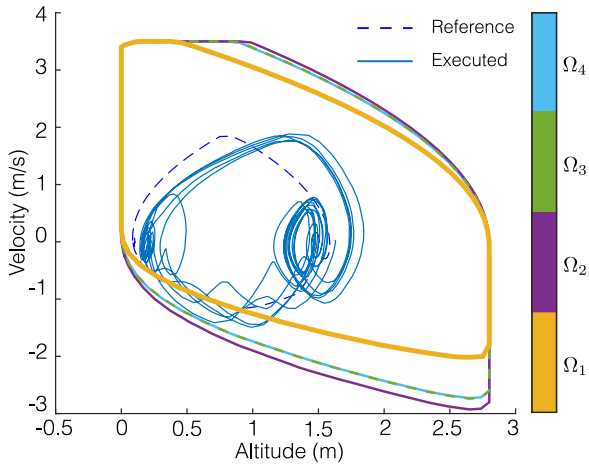


Fig. 5. Safe sets computed online by the safety algorithm as it gathers data and successively updates its Gaussian process disturbance model. The vehicle's trajectory eventually leaves the initial, conservative $\Omega_1$, but remains in the converged safe set ($\Omega_4$) at all times, even *before* this set is computed. While the intermediate set $\Omega_2$ would have been overly permissive, this is remedied by the intervention of the safety controller as soon as the model is observed to behave poorly.

The expressions (10) and (11) give the marginal Gaussian process posterior on $d(x^*)$ for a query point $x^*$. To numerically compute the safe set, the system first evaluates (13) to obtain the disturbance bound $\hat{\mathcal{D}}(\mathbf{x_i})$ at every point $\mathbf{x_i}$ on a state-space grid, as the 95% confidence interval ($p = 0.95$) of the Gaussian process posterior over $d(x)$; next, it performs the robust safety analysis by numerically solving the HJI equation (7) on this grid (using [19]) and obtaining the safety function $V(x)$.

The trajectory followed by the quadrotor in this experiment is shown in Fig. 4. The vehicle starts off with an *a priori* conservative global bound on $d(x)$ and computes an initial conservative safe set $\Omega_1$ (see Fig. 5). It then attempts to track the reference

trajectory avoiding the unsafe regions by transitioning to the safe control $u^*(x)$ on $\partial\Omega_1$. The disturbance is measured and monitored online during this test, under the local safety confidence criterion, and found to be locally consistent with the initial conservative bound. After collecting 10 s of data, a new disturbance bound $\hat{\mathcal{D}}(x)$ is constructed using the corresponding Gaussian process posterior, from which a second safety function $V_2(x)$ and safe set $\Omega_2$ are computed via Hamilton–Jacobi reachability analysis. This process takes roughly 2 s, and at approximately $t = 12$ s the new safety guarantees and policy are substituted in.

The Pelican continues its flight under the results of this new safety analysis: however, shortly after, the vehicle measures values of $d$ that consistently approach the boundary of $\hat{\mathcal{D}}(x)$, and reacts by applying the safe control policy and locally climbing the computed safety function. This confidence-based intervention takes place several times during the test run, as the vehicle measures disturbances that lower its confidence in the local model bounds, effectively preventing the vehicle from approaching the ground.

After a few seconds, a new Gaussian process posterior is computed based on the first 20 s of flight data, resulting in an estimated safe set $\Omega_3$, an intermediate result between the initial conservative $\Omega_1$ and the overly permissive $\Omega_2$ (see Fig. 5). The learning algorithm is then allowed to resume tracking under this new safety analysis, and no further safety overrides take place due to loss of safety confidence.

This experiment demonstrates the algorithm's ability to safely refine its notion of safety as more data become available, without requiring the process to consist in a series of strictly conservative underapproximations.

### C. Gone With the Wind

In this last experimental result, we display the efficacy of online safety guarantee validation in handling alterations in operating conditions unforeseen by the system designer. All components of the framework are active, except for the *iterative safety re-computation*, which is not used in this case.

This experiment is performed using the lighter Hummingbird quadrotor, which is more agile than the Pelican but also more susceptible to wind. We initialize the disturbance set to a conservative range of $\pm 2$ m/s$^2$, which amply captures the error in the double-integrator model for vertical flight. The vehicle tracks a slow sinusoidal trajectory using policy gradient [34] to improve upon the manually initialized controller parameters. At approximately $t = 45$ s an unmodeled disturbance is introduced by activating a fan aimed laterally at the quadrotor. The fan is positioned on the ground and angled slightly upward, so that its effect increases as the quadrotor flies closer to the ground. The presence of the airflow causes the attitude and lateral position controllers to use additional control authority to stabilize the quadrotor, which couples into the vertical dynamics as an unmodeled force.

The experiment is performed with and without the *Bayesian guarantee validation* component, with resulting trajectories shown in Fig. 6. Without validation, the quadrotor violates the constraints, repeatedly striking the ground. With validation, the fan's airflow is quickly detected as a discrepancy with the model near the floor, and the safety controller override is triggered. The vehicle avoids entering the affected region for the remainder of the flight. Although only the local confidence method is used, providing a strictly local safety guarantee, the safe controller succeeds in maintaining safety throughout the experiment. This provides strong evidence suggesting that, beyond its theoretical guarantees, the local Bayesian analysis also constitutes an effective best-effort approach to safety in more general conditions, given limited computational resources and available knowledge about the system.

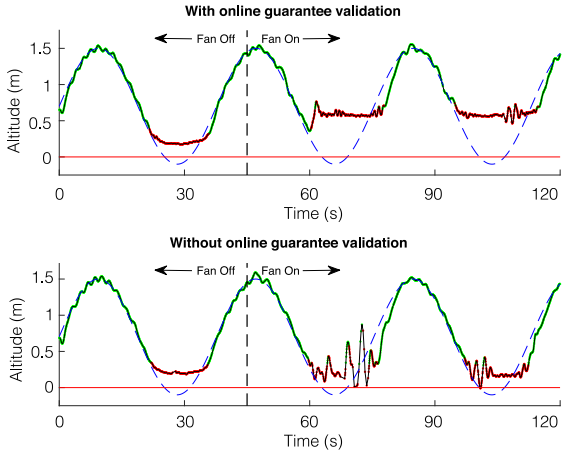


Fig. 6. Vehicle altitude and reference trajectory over time, shown with and without online model validation. After the fan is turned ON, the vehicle checking local model reliability detects the inconsistency and overrides the learning controller, avoiding the region with unmodeled dynamics; the vehicle without model validation enters this region and collides with the ground multiple times. The behavior is repeated when the reference trajectory enters the perturbed region a second time.

## VI. Conclusion

We have introduced a safe learning framework that combines robust reachability guarantees from control theory with Bayesian analysis based on empirical observations. This results in a minimally restrictive supervisory controller that can allow an arbitrary learning algorithm to safely explore its state and strategy spaces. As more data are gathered online, the framework allows the system to probabilistically reason about the validity of its robust model-based safety guarantees in light of the latest empirical evidence.

We firmly believe that providing strong and practically useful safety guarantees for systems that navigate unstructured environments requires a rapprochement between model-based and data-driven techniques, often regarded as a dichotomy by both theoreticians and practitioners. With this paper, we intend to provide mathematical arguments and empirical evidence of the potential that the two approaches hold when used in conjunction.

Scaling up safety certificates as intelligent systems achieve increasing complexity poses an important open-research problem. Our prediction is that, as autonomous systems interact increasingly closely with human beings, the graceful interplay of safety and learning, combining theoretical guarantees with empirical grounding, will become central to their success.

## APPENDIX

Restricting one of the control inputs $d$ to a *state-dependent* bound $\hat{\mathcal{D}}(x)$ introduces questions as to whether a unique Carathéodory solution to (2) continues to exist. The basic existence and uniqueness theorems [35] assume fixed control sets. If the variation in the control sets can instead be expressed through the dynamic equation itself without breaking the continuity conditions, then it is easy to extend the classical result to at least a class of space-dependent control sets. We introduce two technical assumptions, which give sufficient conditions for the existence and uniqueness of a solution to the dynamical equations, and prove that any disturbance set $\hat{\mathcal{D}}(x)$ obtained from a Gaussian process model with Lipschitz prior mean and covariance kernel satisfies these assumptions.

*Assumption 1:* For all $x \in \mathbb{R}^n$, $\hat{\mathcal{D}}(x)$ is a closed deformation retract of $\mathcal{D}$, that is, there exists a continuous map $H_x : \mathcal{D} \times [0,1] \to \hat{\mathcal{D}}(x)$ such that for every $d \in \mathcal{D}$ and $\hat{d} \in \hat{\mathcal{D}}(x)$, $H_x(d,0) = d, H_x(d,1) \in \hat{\mathcal{D}}(x), H_x(\hat{d},1) = \hat{d}$.

*Assumption 2:* Let $r : \mathbb{R}^n \times \mathcal{D} \to \mathcal{D}$ be such that $r(x,d) = H_x(d,1)$ as defined previously. Then, $r$ is Lipschitz continuous in $x$, and uniformly continuous in $d$.

Intuitively, the first assumption means that $\mathcal{D}$ can be continuously deformed into $\hat{\mathcal{D}}(x)$ for any $x$, while the second prevents the disturbance bound $\hat{\mathcal{D}}(x)$ from changing abruptly as one moves in the state space. The retraction map $r$ allows us to absorb the state-dependence of the disturbance bound into the system dynamics, enabling us to use the standard analysis for differential games, which considers measurable time signals drawn from fixed input sets. This is formalized in the following result.

*Proposition 8:* The saturated system dynamics $\tilde{f}_{\hat{\mathcal{D}}}(x,u,d) := f(x,u,r(x,d))$ are bounded and uniformly continuous in all variables, and Lipschitz in $x$.

*Proof:* Boundedness and uniform continuity of $\tilde{f}_{\hat{\mathcal{D}}}$ in $u$ are trivially inherited from $f$; we therefore focus on $d$ and $x$.

First, since $r$ is uniformly continuous in $d$, and $f$ is Lipschitz (hence uniformly continuous) in its third argument, we have by composition that $\tilde{f}_{\hat{\mathcal{D}}}$ is uniformly continuous in $d$.

Lipschitz continuity in $x$ is less immediate due to its appearance in both the first and third arguments of $f$. Again by composition, Lipschitz continuity of $r$ in $x$ implies that $f(y,u,r(\cdot,d))$ is also Lipschitz for all $y \in \mathbb{R}^n$, $u \in \mathcal{U}$ and $d \in \mathcal{D}$. Letting $L_r$ be the Lipschitz constant of $r$ and $L_x$ be the Lipschitz constant of $f$ in its first argument, we have, for any $x, \tilde{x} \in \mathbb{R}^n$:

$$\begin{aligned}
&|f(x,u,r(x,d)) - f(\tilde{x},u,r(\tilde{x},d))| \\
&\leq |f(x,u,r(x,d)) - f(\tilde{x},u,r(x,d))| \\
&\quad + |f(\tilde{x},u,r(x,d)) - f(\tilde{x},u,r(\tilde{x},d))| \\
&\leq (L_x + L_d L_r)|x - \tilde{x}|.
\end{aligned}$$

Thus, $\tilde{f}_{\hat{\mathcal{D}}}(\cdot,u,d) = f(\cdot,u,r(\cdot,d))$ is indeed Lipschitz in $x$. ■

*Corollary 3:* The dynamical system given by (2) with $d$ constrained to lie in a state-dependent set $\hat{\mathcal{D}}(x)$ satisfying Assumptions 1 and 2 has a unique continuous solution in the extended (Carathéodory) sense.

The above-mentioned result is important in that it guarantees existence and uniqueness of system trajectories for any state-dependent disturbance bound $\hat{\mathcal{D}}(\cdot)$ that satisfies Assumptions 1 and 2. Moreover, the above-mentioned construction allows us to transform the system dynamics $f(x,u,d)$ with $d \in \hat{\mathcal{D}}(x)$ into the standard form with fixed input sets (i.e., $\tilde{f}_{\hat{\mathcal{D}}}(x,u,d)$ with $u \in \mathcal{U}$, $d \in \mathcal{D}$), so that all results from the differential games literature can readily be applied to our formulation.

Let us now see that the posterior mean and standard deviation of the components of $d(x)$ are Lipschitz continuous functions of the state $x$ under our Gaussian process framework.

*Proposition 9:* Let the prior mean function $\mu^j$ be Lipschitz continuous, and the covariance kernel $k^j$ be jointly Lipschitz continuous, for the $j$th component of the disturbance function $d(x)$. Then, the posterior mean $\bar{d}^j(x)$ and standard deviation $\sigma^j(x)$, as given by (10) and (11) are Lipschitz continuous in $x$.

*Proof:* The result follows from applying the hypotheses to (10) and (11). Note that the standard deviation $\sigma^j(x)$ is the square root of the variance in (11); the square root function is Lipschitz everywhere except at 0, and Bayesian inference under nondegenerate prior and likelihood never results in 0 posterior variance. Thus, $\sigma^j(\cdot)$ is also Lipschitz. ■

The following proposition relates the state-dependent bound $\hat{\mathcal{D}}(x)$ obtained from Gaussian process regression to Assumptions 1 and 2, ensuring that the dynamical system (2) is well defined, and therefore the associated dynamic game can be solved using the methods presented in Section III.

*Proposition 10:* Let the prior mean function $\mu^j$ be Lipschitz continuous, and the covariance kernel $k^j$ be jointly Lipschitz continuous in its two variables, for all components $j$ of the disturbance function $d(x)$. Then, the disturbance bound $\hat{\mathcal{D}}(x)$, as defined in (13), satisfies Assumptions 1 and 2.

*Proof:* Assumption 1 holds independently of the Lipschitz condition. The bound $\hat{\mathcal{D}}(x)$ given by (13) is a compact convex set in $\mathcal{D}$. As a result, the retraction map $\pi_x : \mathcal{D} \to \hat{\mathcal{D}}(x)$ assigning every $d \in \mathcal{D}$ its (unique) nearest point in $\hat{\mathcal{D}}(x)$ is a Lipschitz continuous function (with Lipschitz constant equal to 1); of course with $\pi_x(\hat{d}) = \hat{d}$ for all $\hat{d} \in \hat{\mathcal{D}}(x)$. Then, the function $H_x(d,t) := (1-t)d + t\pi_x(d)$ is continuous by composition and further satisfies $H_x(d,0) = d, H_x(d,1) \in \hat{\mathcal{D}}(x), H_x(\hat{d},1) = \hat{d}$ for all $d \in \mathcal{D}$ and $\hat{d} \in \hat{\mathcal{D}}(x)$.

Assumption 2 can be shown to hold by noting that the extrema of each of the intervals in (13) are affine in $\bar{d}^j(x)$ and $\sigma^j(x)$, which are Lipschitz continuous in $x$ by Proposition 9. This implies that the position of all vertices of the hyperrectangle in (13) varies as a Lipschitz continuous function of $x$, and so does, as a result, the nearest point in $\hat{\mathcal{D}}(x)$ to any fixed $d \in \mathcal{D}$. The map $r(x,d) := \pi_x(d)$ is hence Lipschitz continuous in $x$. Finally, since $\pi_x$ is Lipschitz continuous in $d$, $r$ is also uniformly continuous in $d$. ■

Finally, we can show that, under the same Lipschitz assumptions on the Gaussian process prior, the disturbance bound $\hat{\mathcal{D}}(x)$ is Lipschitz continuous under the Hausdorff distance, which we required in Proposition 1.

*Proposition 11:* Let the prior mean function $\mu^j$ be Lipschitz continuous, and the covariance kernel $k^j$ be jointly Lipschitz

continuous in its two variables, for all components $j$ of the disturbance function $d(x)$. Then, the set-valued map $\hat{\mathcal{D}}$ is Lipschitz continuous under the Hausdorff distance.

*Proof:* Since the disturbance set $\hat{\mathcal{D}}(x)$ given by (13) is a hyperrectangle, the Hausdorff distance between the disturbance sets $\hat{\mathcal{D}}(x_1)$ and $\hat{\mathcal{D}}(x_2)$ is upper-bounded by the maximum distance between any pair of corners

$$\delta(x_1, x_2) := d_H\left(\hat{\mathcal{D}}(x_1), \hat{\mathcal{D}}(x_2)\right) \leq \max_i \max_k |c_i - c_k|$$

with $c_i, c_k$ used to enumerate all corners of each of the two hyperrectangles. For simplicity of exposition, we use the equivalence of all norms in $\mathbb{R}^{n_d}$ to upper-bound the above, arbitrary norm in $\mathbb{R}^{n_d}$, by the infinity norm, up to a constant factor $m$, which in combination with (13) gives

$$\delta(x_1, x_2) \leq m \cdot \max_j (|\bar{d^j}(x_1) - \bar{d^j}(x_2)| + |z\sigma^j(x_1) - z\sigma^j(x_2)|).$$

Now, by Proposition 9, $\bar{d^j}(x)$ and $\sigma^j(x)$ are Lipschitz continuous in $x$; let their respective constants be $L_\mu^j$ and $L_\sigma^j$. We then have

$$d_H\left(\hat{\mathcal{D}}(x_1), \hat{\mathcal{D}}(x_2)\right) \leq m \cdot \max_j \left(L_\mu^j + zL_\sigma^j\right) |x_1 - x_2|$$

which proves Hausdorff Lipschitz continuity of the set-valued map $\hat{\mathcal{D}}$, with a Lipschitz constant $L_{\hat{\mathcal{D}}}$ upper-bounded by $m \cdot \max_j L_\mu^j + zL_\sigma^j$. ■

## REFERENCES


[1] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, 2015.

[2] J. Schulman, S. Levine, M. Jordan, and P. Abbeel, "Trust region policy optimization," in *Proc. Int. Conf. Mach. Learn.*, 2015.

[3] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng, "An application of reinforcement learning to aerobatic helicopter flight," *in Proc. Adv. Neural Inf. Process. Syst.*, 2007.

[4] A. Coates, P. Abbeel, and A. Y. Ng, "Learning for control from multiple demonstrations," in *Proc. Int. Conf. Mach. Learn.*, 2008.

[5] J. Z. Kolter, C. Plagemann, D. T. Jackson, A. Y. Ng, and S. Thrun, "A probabilistic approach to mixed open-loop and closed-loop control, with application to extreme autonomous driving," in *Proc. Int. Conf. Robot. Automat.*, 2010.

[6] S. Lupashin, A. Schöllig, M. Sherback, and R. D'Andrea, "A simple learning strategy for high-speed quadrocopter multi-flips," in *Proc. Int. Conf. Robot. Automat.*, 2010.

[7] A. Hobbs, "Unmanned aircraft systems," in *Human Factors in Aviation*, 2nd ed., E. Salas and D. Maurino, Eds. Amsterdam, The Netherlands: Elsevier, 2010.

[8] P. Christiano *et al.*, "Transfer from simulation to real world through learning deep inverse dynamics model," 2016.

[9] S. H. Huang, N. Papernot, I. Goodfellow, Y. Duan, and P. Abbeel, "Adversarial attacks on neural network policies," in *Proc. Int. Conf. Learn. Rep.*, 2017.

[10] T. J. Perkins and A. G. Barto, "Lyapunov design for safe reinforcement learning," *J. Mach. Learn. Res.*, vol. 3, 2003.

[11] J. W. Roberts, I. R. Manchester, and R. Tedrake, "Feedback controller parameterizations for reinforcement learning," in *Proc. Symp. Adaptive Dyn. Program. Reinforcement Learn.*, 2011.

[12] P. Geibel and F. Wysotzki, "Risk-sensitive reinforcement learning applied to control under constraints," *J. Artif. Intell. Res.*, vol. 24, 2005.

[13] J. García and F. Fernández, "A comprehensive survey on safe reinforcement learning," *J. Mach. Learn. Res.*, vol. 16, 2015.

[14] T. M. Moldovan and P. Abbeel, "Safe exploration in Markov decision processes," in *Proc. Int. Conf. Mach. Learn.*, 2012.

[15] 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. Conf. Decis. Control*, 2016.

[16] F. Berkenkamp, M. Turchetta, A. P. Schoellig, and A. Krause, "Safe model-based reinforcement learning with stability guarantees," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.

[17] A. Aswani, H. Gonzalez, S. S. Sastry, and C. Tomlin, "Provably safe and robust learning-based model predictive control," *Automatica*, vol. 49, no. 5, 2013.

[18] 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. Autom. Control*, vol. 50, no. 7, 2005.

[19] I. Mitchell and J. Templeton, "A toolbox of Hamilton–Jacobi solvers for analysis of nondeterministic continuous and hybrid systems," in *Proc. Int. Workshop Hybrid Syst.: Comput. Control*, 2005.

[20] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, vol. 153. Berlin, Germany: Springer Science & Business Media, 2003.

[21] J. Ding, J. Gillula, H. Huang, M. P. Vitus, W. Zhang, and C. J. Tomlin, "Toward reachability-based controller design for hybrid systems in robotics," in *Proc. Int. Symp. Artif. Intell., Robot. Automat. Space*, 2011.

[22] M. Chen, J. F. Fisac, S. Sastry, and C. J. Tomlin, "Safe sequential path planning of multi-vehicle systems via double-obstacle Hamilton–Jacobi–Isaacs variational inequality," in *Proc. Eur. Control Conf.*, 2015.

[23] S. L. Herbert, M. Chen, S. Han, S. Bansal, J. F. Fisac, and C. J. Tomlin, "FaSTrack: A modular framework for fast and guaranteed safe motion planning," in *Proc. Conf. Decis. Control*, 2017.

[24] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *Proc. Int. Workshop Hybrid Syst.: Comput. Control*, 2004.

[25] C. Sloth, G. J. Pappas, and R. Wisniewski, "Compositional safety analysis using barrier certificates," in *Proc. Int. Conf. Hybrid Syst.: Comput. Control*, 2012.

[26] J. H. Gillula and C. J. Tomlin, "Guaranteed safe online learning via reachability: Tracking a ground target using a quadrotor," in *Proc. Int. Conf. Robot. Automat.*, 2012.

[27] J. H. Gillula and C. J. Tomlin, "Reducing conservativeness in safety guarantees by learning disturbances online: Iterated guaranteed safe online learning," in *Proc. Conf.Robot.: Sci. Syst.*, 2012.

[28] 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. Conf. Decis. Control*, 2014.

[29] S. Kaynama and M. Oishi, "A modified Riccati transformation for decentralized computation of the viability kernel under LTI dynamics," *IEEE Trans. Autom. Control*, vol. 58, no. 11, 2013.

[30] S. Kaynama, I. M. Mitchell, M. Oishi, and G. A. Dumont, "Scalable safety-preserving robust control synthesis for continuous-time linear systems," *IEEE Trans. Autom. Control*, vol. 60, no. 11, 2015.

[31] M. Chen, S. Herbert, and C. J. Tomlin, "Fast reachable set approximations via state decoupling disturbances," in *Proc. Conf. Decis. Control*, 2016.

[32] J. Darbon and S. Osher, "Algorithms for overcoming the curse of dimensionality for certain Hamilton–Jacobi equations arising in control theory and elsewhere," *Res. Math. Sci.*, vol. 3, no. 1, 2016.

[33] M. R. Kirchner, R. Mar, G. Hewer, J. Darbon, S. Osher, and Y. T. Chow, "Time-optimal collaborative guidance using the generalized Hopf formula," *IEEE Control Syst. Lett.*, vol. 2, no. 2, 2018.

[34] J. Z. Kolter and A. Y. Ng, "Policy search via the signed derivative," *Robot.: Sci. Syst.*, 2009.

[35] E. A. Coddington and N. Levinson, *Theory of Ordinary Differential Equations*. New York, NY, USA: McGraw-Hill, 1955.

[36] L. C. Evans and P. E. Souganidis, "Differential games and representation formulas for solutions of Hamilton–Jacobi–Isaacs equations," *Indiana Univ. Math. J.*, vol. 33, no. 5, 1984.

[37] E. Barron, "Differential games with maximum cost," *Nonlinear Anal.: Theory, Methods Appl.*, vol. 14, 1990.

[38] J. F. Fisac, M. Chen, C. J. Tomlin, and S. S. Sastry, "Reach-avoid problems with time-varying dynamics, targets and constraints," in *Proc. 18th Int. Conf. Hybrid Syst. Comput. Control*. ACM Press, 2015.

[39] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[40] A. Genz, "Numerical computation of multivariate normal probabilities," *J. Comput. Graphical Statist.*, vol. 1, no. 2, 1992.

[41] C.-W. Shu and S. Osher, "Efficient implementation of essentially non-oscillatory shock-capturing schemes," *J. Comput. Phys.*, vol. 77, no. 2, 1988.

**Jaime F. Fisac** received the Diploma degree in electrical engineering from the Universidad Politécnica de Madrid, Madrid, Spain, in 2012, the M.Sc. degree in aeronautics from Cranfield University, Cranfield, U.K., in 2013, and the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, Berkeley CA, USA, in 2019.

Dr. Fisac is the recipient of the "la Caixa" Foundation Fellowship (2013–2015). His research interests include control theory and artificial intelligence, with a focus on safety in human centered autonomous systems.

**Anayo K. Akametalu** received the B.S. degree in electrical engineering from the University of California, Santa Barbara, Santa Barbara, CA, USA, in 2012, and the M.S. and Ph.D. degrees in electrical engineering and computer sciences from the University of California, Berkeley, Berkeley, CA, USA, in 2014 and 2018, respectively.

He is an Autonomy Engineer at Uber Advanced Technologies Group, San Francisco, CA, USA. His research was funded through the National Science Foundation Bridge to Doctorate Fellowship, UC Berkeley Chancellor's Fellowship, and GEM Fellowship.

**Melanie N. Zeilinger** received the Diploma degree in engineering cybernetics from the University of Stuttgart, Stuttgart, Germany, in 2006, and the Ph.D. degree in electrical engineering from ETH Zurich, Zurich, Switzerland, in 2011.

She is an Assistant Professor with the Department of Mechanical and Process Engineering, ETH Zurich, Zurich, Switzerland. She was a Marie Curie Fellow and Postdoctoral Researcher with the Max Planck Institute for Intelligent Systems, Germany (until 2015) and with UC Berkeley, CA, USA (2012–2014), and a Postdoctoral Researcher with EPFL, Switzerland (2011–2012). Her research interests include distributed control and optimization, as well as safe learning-based control, with applications to human-in-the-loop systems.

**Shahab Kaynama** received the M.Sc. degree in advanced control and systems engineering from the University of Manchester, Manchester, U.K., in 2006, and the Ph.D. degree in electrical and computer engineering from the University of British Columbia, Vancouver, BC, Canada, in 2012.

He is a Senior Algorithms Architect with Apple Inc., Cupertino, CA, USA. Between 2014 and 2017, he was with Clearpath Robotics as an Autonomy Team Lead for Navigation and Controls, where he helped launch its two OTTO flagship products. He was a Postdoctoral Research Scholar with UC Berkeley (since 2012) and with the University of British Columbia (since 2014).

**Jeremy Gillula** received the B.S. degree in computer science from the California Institute of Technology, Pasadena, CA, USA, in 2006, and the M.S. and Ph.D. degrees in computer science from the Stanford University, Stanford, CA, USA, in 2011 and 2013, respectively.

After spending 8 months as a Postdoctoral Researcher with UC Berkeley, he became a Staff Technologist with the Electronic Frontier Foundation, a San Francisco-based nonprofit organization dedicated to defending civil liberties in the digital world.

**Claire J. Tomlin** received the B.A.Sc. degree in electrical engineering from the University of Waterloo, Waterloo, Canada, in 1992, the M.S. degree in electrical engineering from Imperial College, University of London, London, U.K., in 1993, and the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, Berkeley, CA, USA, in 1998.

She is the Charles A. Desoer Professor of Engineering in electrical engineering and computer sciences at the University of California, Berkeley, Berkeley, CA, USA. She was an Assistant, an Associate, and a Full Professor in aeronautics and astronautics with Stanford University, Stanford, CA, USA, from 1998 to 2007, and in 2005 joined Berkeley. Her research interests include the area of control theory and hybrid systems, with applications to air traffic management, UAV systems, energy, robotics, and systems biology.

Prof. Tomlin is a MacArthur Foundation Fellow (2006) and and in 2010 held the Tage Erlander Professorship of the Swedish Research Council with KTH Royal Institute of Technology, Stockholm, Sweden.