

**Document Version**

Final published version

**Licence**

Dutch Copyright Act (Article 25fa)

**Citation (APA)**

Laurenti, L., & Lahijanjan, M. (2026). A Unifying Perspective for Safety of Stochastic Systems: From Barrier Functions to Finite Abstractions. *IEEE Transactions on Automatic Control*, 71(2), 769-779.  
<https://doi.org/10.1109/TAC.2025.3597569>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

In case the licence states "Dutch Copyright Act (Article 25fa)", this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.  
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

**Sharing and reuse**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

# A Unifying Perspective for Safety of Stochastic Systems: From Barrier Functions to Finite Abstractions

Luca Laurenti , Member, IEEE, and Morteza Lahijanian , Member, IEEE

**Abstract**—Providing safety guarantees for stochastic dynamical systems is a central problem in various fields, including control theory, machine learning, and robotics. Existing methods either employ *stochastic barrier functions* (SBFs) or rely on numerical approaches based on *finite abstractions*. SBFs, analogous to Lyapunov functions, are used to establish (probabilistic) set invariance, whereas abstraction-based approaches approximate the stochastic system with a finite model to compute safety probabilistic bounds. This article presents a *unifying perspective* on these seemingly different approaches. Specifically, we show that both methods can be interpreted as approximations of a *stochastic dynamic programming* problem. This perspective allows us to formally establish the correctness of both techniques, characterize their convergence and optimality properties, and analyze their respective assumptions, advantages, and limitations. Our analysis reveals that, unlike SBFs-based methods, abstraction-based approaches can provide asymptotically optimal safety certificates, albeit at the cost of increased computational effort.

**Index Terms**—Barrier function, finite abstraction, probabilistic safety, robustness, stochastic systems.

## I. INTRODUCTION

IN the age of autonomous systems, control systems have become ubiquitous, playing a pivotal role in *safety-critical* applications. Examples span from autonomous vehicles [1] to medical robots [2], where the consequences of bad decisions are not only costly but can also prove fatal. A common characteristic of these systems is the inherent complexity and nonlinearity of the dynamics that are subject to uncertainty due to physics (e.g., sensor or actuation noise) or algorithms (e.g., black-box controllers or perception). Consequently, the question *how to ensure the safety of stochastic control systems?* has emerged as a central research topic in various disciplines, including control

Received 25 March 2025; accepted 2 August 2025. Date of publication 11 August 2025; date of current version 30 January 2026. This work was supported by NSF under Grant 2039062. Recommended by Associate Editor D. Antunes. (Luca Laurenti and Lahijanian contributed equally to this work.) (Corresponding author: Luca Laurenti.)

Luca Laurenti is with the Delft University of Technology, 2628, CD Delft, The Netherlands (e-mail: L.Laurenti@tudelft.nl).

Morteza Lahijanian is with the University of Colorado Boulder, Boulder, CO 80309 USA (e-mail: morteza.lahijanian@colorado.edu).

Digital Object Identifier 10.1109/TAC.2025.3597569

theory, machine learning, formal methods, and robotics [3], [4], [5].

In stochastic systems, *safety* generalizes the standard notion of *stochastic stability* [5]. It is defined as the probability that the system avoids unsafe behavior, i.e., with high probability, the first exit time from a given safe set is greater than a specific threshold or infinite. However, computing the exact safety probability is generally intractable, even in simple stochastic *linear* systems. As a result, existing formal methods rely on *underapproximations*, with two of the most commonly employed approaches being *stochastic barrier functions* (SBFs) [4], [6] and finite-abstraction methods [7], [8]. Similar to Lyapunov functions for stability, SBFs are energy-like functions that provide lower bounds on the probability of a stochastic system remaining within a safe set. On the other hand, abstraction-based methods abstract the original system into a finite-state stochastic process, typically a variant of a Markov chain (MC), for which efficient algorithms exist to compute safety probabilities [9], [10], [11], [12].

This article presents a *unifying perspective* on safety verification of discrete-time, continuous-space stochastic systems through the lens of *dynamic programming* (DP), bridging the gap between SBFs and abstraction-based methods. While traditionally seen as distinct, with SBFs typically derived from super-martingale conditions [5], we show that both approaches can be understood as approximations of the same DP framework. First, we establish that, for the class of systems considered in this article, the safety probability can be computed via DP, and that there always exists a deterministic Markov policy (strategy) that optimizes this probability. We then show that both SBFs and abstraction-based methods can be seen as approximations of this DP framework. Specifically, for SBFs, we show that existing bounds can be obtained by overapproximating the indicator function of the unsafe set with a barrier function, i.e., a nonnegative function that is at least one in the unsafe set. On the other hand, abstraction-based methods, which convert the system into an (uncertain) Markov process, arise as piecewise-constant over and underapproximations of the defined DP.

Viewing both SBF and abstraction-based methods as approximations of a DP problem has several advantages: 1) it gives a unified treatment of safety for stochastic systems, 2) it allows us to establish formal bounds and guarantees on the precision and correctness of these approaches, and 3) it enables

us to fairly compare these methods and highlight their strengths and weaknesses. Specifically, we show how abstraction-based methods can return tighter bounds on the safety probability compared to SBFs, and this generally comes at the cost of increased computational effort.

In summary, the main contributions of this article are as follows.

- 1) Introducing a unifying perspective on SBF and abstraction-based methods as approximations of a stochastic DP algorithm.
- 2) Providing formal proofs on the convergence of abstraction-based methods to the optimal safety probability and optimal strategies, as well as demonstrating the lack of convergence guarantees for (Martingale-based) SBF methods to the optimal probability and strategies.
- 3) Discussing the pros and cons of each method, supported by theoretical proofs and empirical illustrations.

The rest of this article is organized as follows. In Section II, we introduce the class of systems we consider and formally define probabilistic safety. In Section III, we formally define SBFs. We present abstraction-based methods in Section IV. In Section V, we analyze strengths and weaknesses of each method. Finally, Section VI concludes this article, highlighting important open research questions. We provide all the proofs in Appendix.

## II. PROBABILISTIC SAFETY FOR DISCRETE-TIME STOCHASTIC PROCESSES

Consider a discrete-time controlled stochastic system described by the following difference equation:

$$\mathbf{x}[k+1] = F(\mathbf{x}[k], \mathbf{u}[k], \mathbf{v}[k]) \quad (1)$$

where  $\mathbf{x}[k]$  is the state of the system at time  $k \in \mathbb{N}$  taking values in  $X \subseteq \mathbb{R}^{n_x}$ , and  $\mathbf{u}[k]$  denotes the control or action at time  $k$  taking values in compact set  $U \subset \mathbb{R}^{n_u}$ . For a probability space  $(\Omega, \mathcal{F}, P_v)$ ,  $\mathbf{v}[k] : \Omega \rightarrow V \subseteq \mathbb{R}^{n_v}$  is a random variable with associated probability distribution  $P_v$ , which represents the noise affecting the system at every time step.  $\mathbf{v}[k]$  is assumed to be independent and identically distributed at every time step  $k$ . Furthermore, sets  $X$ ,  $U$ , and  $V$  are all assumed to be appropriately (Borel) measurable sets. Finally,  $F : X \times U \times V \rightarrow X$  is a possibly nonlinear measurable function representing the one-step dynamics of the system.

*Remark 1:* System (1) represents a general model of a nonlinear controlled stochastic system, which includes a large class of stochastic models used in practice. For instance, in the case of additive noise, we have  $F(\mathbf{x}[k], \mathbf{u}[k], \mathbf{v}[k]) = \bar{F}(\mathbf{x}[k], \mathbf{u}[k]) + \mathbf{v}[k]$ . In this example, if  $\bar{F}(\mathbf{x}[k], \mathbf{u}[k])$  is a neural network, then System (1) becomes a stochastic neural network dynamic model [3], [13].

*Definition 1 (Policy):* A *feedback policy* (or strategy)  $\pi = (\pi_0, \pi_1, \dots)$  for System (1) is a sequence of universally measurable stochastic kernels such that, for every  $k \geq 0$ ,  $\pi_k : X^{k+1} \rightarrow \mathcal{P}(U)$ ,<sup>1</sup> where  $\mathcal{P}(U)$  is the set of probability measures over  $U$ . Policy  $\pi$  is called *deterministic* if for each  $k$  and  $(x_0, \dots, x_k)$ ,

<sup>1</sup> Note that we assume that  $\pi_k$  is independent of the value of the previous actions, and only depends on previous states. This is without loss of generality

$\pi_k(x_0, \dots, x_k)$  assigns probability mass one to some  $u \in U$ . In this case, with a slight abuse of notation,  $\pi = (\pi_0, \pi_1, \dots)$  can be considered as the sequence of universally measurable functions  $\pi_k : X^{k+1} \rightarrow U$ . If for each  $k$ ,  $\pi_k$  only depends on  $x_k$ , where  $\pi$  is a *Markov policy*. A policy is *stationary* if, for every  $k_1, k_2 \in \mathbb{N}$ , it holds that  $\pi_{k_1} = \pi_{k_2}$ , in which case, with an abuse of notation, we use  $\pi$  to denote any of these functions. The set of all policies is denoted by  $\Pi$ , while the set of deterministic Markov policies by  $\Pi^{M,D}$ .

For a given initial condition  $x_0$ , a time horizon  $H \in \mathbb{N}$ , and a policy  $\pi = (\pi_0, \dots, \pi_{H-1})$ ,  $\mathbf{x}[k]$  is a stochastic process with a well-defined probability measure  $P$  generated by the noise distribution  $p_{\mathbf{v}_k}$  [14, Prop. 7.45] such that for  $k \in \{0, \dots, H-1\}$  and measurable sets  $X_0, X_{k+1} \subseteq X$ , it holds that

$$\begin{aligned} P(\mathbf{x}[0] \in X_0 \mid \mathbf{u}[0] = a) &= \mathbb{1}(x_0, X_0) \\ P(\mathbf{x}[k+1] \in X_{k+1} \mid \mathbf{x}[k] = x, \mathbf{u}_k = a) \\ &:= \int_{\Omega} \mathbb{1}(F(x, \mathbf{v}[k](\omega), a), X_{k+1}) P_v(d\omega) \\ &:= T(X_{k+1} \mid x, a) \end{aligned}$$

where

$$\mathbb{1}(x_k, X_k) = \begin{cases} 1, & \text{if } x_k \in X_k \\ 0, & \text{otherwise} \end{cases}$$

is the indicator function. We refer to  $T(X_{k+1} \mid x_k, a)$  as the *stochastic kernel* of System (1), and we assume that for each measurable  $X_j \subseteq X$ ,  $x \in X$ ,  $a \in U$ ,  $T(X_j \mid x, a)$  is Lipschitz continuous in both  $x$  and  $a$ .

### A. Probabilistic Safety

For a given policy  $\pi$  and a time horizon  $H \in \mathbb{N}$ , probabilistic safety is defined as the probability that  $\mathbf{x}[k]$  stays within a measurable safe set  $X_s \subseteq X$  for the next  $H$  time steps, i.e., the first exit time from  $X_s$  is greater than  $H$ .<sup>2</sup>

*Definition 2 (Probabilistic safety):* Given a policy  $\pi$ , safe set  $X_s \subset X$ , time horizon  $H \in \mathbb{N}$ , and initial set of states  $X_0 \subseteq X_s$ , probabilistic safety is defined as

$$\begin{aligned} P_s(X_s, X_0, H \mid \pi) &= \\ &\inf_{x_0 \in X_0} P(\forall k \in [0, H], \mathbf{x}[k] \in X_s \mid \mathbf{x}[0] = x_0, \pi). \end{aligned}$$

Probabilistic safety and its equivalent dual, probabilistic reachability<sup>3</sup> are widely used to certify the safety of dynamical systems [15] and represent a generalization of the notion of invariance that is commonly employed for analysis of deterministic systems [16].

because probabilistic safety, as defined in Definition 2, only depends on the state values.

<sup>2</sup> Depending on the application domain, horizon  $H$  can be the prediction horizon (e.g., in MPC) or long-term time horizon (e.g., in motion planning).

<sup>3</sup> Given a finite-time horizon  $H \in \mathbb{N}$ , policy  $\pi$ , initial point  $x_0$ , and a target set  $X_u$ , probabilistic reachability is defined as  $P_{\text{reach}}(X_u, x_0, H \mid \pi) = P(\exists k \in [0, H], \mathbf{x}[k] \in X_u \mid \mathbf{x}[0] = x_0, \pi)$ . Consequently, we have that  $P_s(X_s, x_0, H \mid \pi) = 1 - P_{\text{reach}}(X \setminus X_s, x_0, H \mid \pi)$ .

*Remark 2:* In Definition 2, we consider only a finite horizon  $H$ , but this is without loss of generality. For an infinite horizon, one of the following two cases occurs.

- 1) There exists a subregion of  $X_s$  from which the system never exits, in which case  $P_s$  corresponds to the probability of eventually reaching this region while remaining safe.
- 2) Otherwise,  $P_s = 0$ , as in the case where  $\mathbf{v}$  is additive and its distribution has unbounded support (e.g., a Gaussian distribution).

In the remainder of this section, we show that, to compute a policy that maximizes probabilistic safety, it is sufficient to restrict to deterministic Markov policies. To that end, we first show that  $P_s$  can be characterized as the solution of a DP problem. In particular, for  $k \in \{H, H-1, \dots, 0\}$  and  $X_u := X \setminus X_s$ , consider value functions  $V_k^* : X \rightarrow [0, 1]$  defined recursively (backwardly in time) as

$$V_H^*(x) = \mathbb{1}(x, X_u) \quad (2)$$

$$V_k^*(x) = \inf_{a \in U} (\mathbb{1}(x, X_u) + \mathbb{1}(x, X_s) \mathbb{E}_{x' \sim T(\cdot | x, a)} [V_{k+1}^*(x')]) \quad (3)$$

where notation  $x' \sim T(\cdot | x, a)$  means that  $x'$  is distributed according to  $T(\cdot | x, a)$ , and  $\mathbb{E}_{x' \sim T(\cdot | x, a)} [V_{k+1}^*(x')]$  is the expectation of value function  $V_{k+1}^*$  with respect to  $T(\cdot | x, a)$ . Intuitively, at each time step  $k$ ,  $V_k^*$  selects the (deterministic) action that minimizes the probability of reaching a state from which the system may reach  $X_u$  in the next  $H - k$  time steps. Consequently, by propagating  $V_k^*$  backward over time, we compute the probability of reaching  $X_u$  in the future. The following theorem guarantees that  $\sup_{\pi \in \Pi} P_s(X_s, x_0, H | \pi)$  is equal to  $1 - V_0^*(x_0)$ .

*Theorem 1:* For an initial state  $x_0 \in X_s$ , it holds that

$$\sup_{\pi \in \Pi} P_s(X_s, x_0, H | \pi) = 1 - V_0^*(x_0).$$

The proof is in the Appendix, where we show that standard results for optimality of deterministic policies commonly derived for cumulative or discounted reward models [14] extend to probabilistic safety. Therefore, a straightforward consequence of Theorem 1 is Corollary 1, which guarantees that deterministic Markov policies are optimal. In fact, note that in obtaining  $V_0^*$  via (2) and (3), we only consider deterministic Markov policies.

*Corollary 1:* Deterministic Markov policies are sufficient for optimal probabilistic safety, i.e, it holds that

$$\sup_{\pi \in \Pi} P_s(X_s, x_0, H | \pi) = \sup_{\pi \in \Pi^{M,D}} P_s(X_s, x_0, H | \pi).$$

Furthermore, for every  $\pi \in \Pi^{M,D}$ , it holds that

$$P_s(X_s, x_0, H | \pi) = 1 - V_0^\pi(x_0)$$

where  $V_0^\pi(x_0)$  is defined recursively as

$$V_H^\pi(x) = \mathbb{1}(x, X_u) \quad (4)$$

$$V_k^\pi(x) = \mathbb{1}(x, X_u) + \mathbb{1}(x, X_s) \mathbb{E}_{x' \sim T(\cdot | x, \pi_k(x))} [V_{k+1}^\pi(x')]. \quad (5)$$

Theorem 1 and Corollary 1 guarantee that in order to synthesize optimal policies, it is enough to consider deterministic Markov policies, and these policies can be computed via DP.

We should stress that without the assumptions made in this article (compactness of  $U$ , continuity of  $T$ , and measurability of the various sets),  $V_k^*$  may not be measurable and the infimum in  $U$  in (3) may not be attained. In that case, the integrals in the expectations in (3) and (5) have to be intended as outer integrals [14]. However, under the assumptions in this article, the expectations in the above DP are well-defined and a universally measurable deterministic Markov optimal policy exists [14], i.e., the inf is attainable for every point in the state space by a universally measurable function. Nevertheless, even if  $V_0^*$  and  $V_0^\pi$  are well-defined, computation of (3) is infeasible in practice due to the need to solve uncountably many optimization problems. Thus, computation for  $P_s$  requires approximations.

In what follows, we consider the two dominant approaches in the literature that (with certified error bounds) compute probabilistic safety and synthesize policies for System (1), namely, SBFs and *abstraction-based methods*. We show that both approaches arise as overapproximations of the value functions  $V_k^\pi$ . In other words, both SBF-based and abstraction-based methods are approximation techniques to solve the same DP in (4) and (5). Such a unified framework provides a basis for a fair comparison of these approaches, which consequently reveals their advantages and disadvantages. We start by considering SBF in Section III and abstraction-based methods in Section IV.

### III. STOCHASTIC BARRIER FUNCTIONS

We begin with the setting, where a deterministic Markov policy  $\pi$  is given, and we aim to compute a (nontrivial) lower bound of  $P_s(X_s, X_0, H | \pi)$ . We first show how one can use SBFs to compute an upper bound on  $V_0^\pi$  (hence, a lower bound on  $P_s$ ) without the need to directly evolve the dynamics of System (1) over time. Then, we focus on the key challenge with SBFs: how to find an SBF that allows one to bound  $P_s$  without leading to overly conservative results. The control synthesis case is considered in Section III-A.

An SBF [6], [17] is simply a continuous almost everywhere function  $B : \mathbb{R}^{n_x} \rightarrow \mathbb{R}_{\geq 0}$  that overapproximates  $V_H^\pi(x)$ , that is, the indicator function for the unsafe set. In particular, we say function  $B$  is a SBF iff

$$\forall x \in X, B(x) \geq 0 \quad \text{and} \quad \forall x \in X_u, B(x) \geq 1. \quad (6)$$

The intuition is that, when  $B$  is propagated backwards over time in a DP fashion, it produces an overapproximation for  $V_k^\pi(x)$ . That is, value functions  $\bar{V}_k^\pi : \mathbb{R}^{n_x} \rightarrow [0, 1]$  with  $k \in \{0, \dots, H\}$  defined recursively as

$$\bar{V}_H^\pi(x) = B(x) \quad (7)$$

$$\bar{V}_k^\pi(x) = \mathbb{1}(x, X_u) B(x) + \mathbb{1}(x, X_s) \mathbb{E}_{x' \sim T(\cdot | x, \pi_k(x))} [\bar{V}_{k+1}^\pi(x')] \quad (8)$$

overapproximate  $V_k^\pi(x)$ , as formalized in the following lemma.

*Lemma 1:* Consider the value functions  $V_k^\pi(x)$  and  $\bar{V}_k$  defined in (4) and (5) and (7) and (8), respectively. For every  $k \in \{0, \dots, H\}$  and every  $x \in X$ , it holds that  $V_k^\pi(x) \leq \bar{V}_k(x)$ .

Note that a sufficient condition for the inequality in Lemma 1 to become a strict inequality is that  $\exists x \in X_s$  such that  $B(x) > 0$  and  $T(\cdot | x, \pi_k(x))$  has unbounded support for every  $x$ . In fact, this guarantees that for any  $x \in X_s$ ,  $\mathbb{E}_{x' \sim T(\cdot | x, \pi_k(x))}[B(x')] > \mathbb{E}_{x' \sim T(\cdot | x, \pi_k(x))}[\mathbb{1}(x', X_u)]$ .

To obtain  $\bar{V}_k^\pi$ , one needs to compute the expectation in (8) for uncountably many  $x$ , which is generally intractable. To overcome this problem, Martingale-based SBFs, define constant  $\beta \geq 0$  as

$$\beta \geq \sup_{x \in X_s, k \in \{0, \dots, H-1\}} (\mathbb{E}_{x' \sim T(\cdot | x, \pi_k(x))}[B(x')] - B(x)). \quad (9)$$

That is,  $\beta$  bounds how much the probability of reaching  $X_u$  can grow in a single time step. Then, by rearranging terms in (8), we obtain

$$V_k^\pi(x) \leq \bar{V}_k^\pi(x) \leq (H - k)\beta + B(x). \quad (10)$$

This leads to the following theorem.

*Theorem 2 (See [17]):* Let  $B : X \rightarrow \mathbb{R}_{\geq 0}$  be a function that satisfies (6), and consider the bounds  $\eta = \sup_{x \in X_0} B(x)$  and  $\beta$  in (9). Then, it holds that

$$\inf_{x_0 \in X_0} P_s(X_s, x_0, H | \pi) \geq 1 - (\eta + \beta H).$$

Theorem 2 ensures that once the constants  $\eta$  and  $\beta$  are determined, one can establish a lower bound on  $P_s(X_s, x_0, H | \pi)$  for all initial states  $x_0 \in X_0$ . As a result, Theorem 2 reduces solving the DP in (3) to quantifying the constants  $\eta$  and  $\beta$ .

*Remark 3:* From (7) and (8), we can observe that the closer  $B(x)$  is to the indicator function, the closer  $\bar{V}_k^\pi$  gets to  $V_k^\pi$ ; and thus, the tighter the bound computation becomes for  $P_s$ . This may lead the reader to wonder why we do not simply assume  $B(x) = \mathbb{1}(x, X_u)$ . To clarify this, note that, in the derivation of Theorem 2, another source of conservatism comes from the choice of  $\beta$ . In fact,  $\beta$  is the supremum expected change over all  $x \in X_s$ . Hence, setting  $B(x) = \mathbb{1}(x, X_u)$  may lead to overly conservative results, e.g., cases where there are only few regions from which the probability that the system transition to the unsafe set is not negligible. This is illustrated in Fig. 1, where an indicator barrier function is compared against an SBF synthesized using sum-of-square (SoS) optimization as proposed in [17]. This example illustrates that how the choice of  $B$  can have a large impact on  $\beta$  and  $\eta$ , and consequently, on the resulting bounds on  $P_s$ .

Following Remark 3, it is clear that the key challenge in the SBF approach is finding a valid  $B$  that avoids excessively conservative results while also allowing for the efficient computation of  $\eta$  and  $\beta$ . Let  $B \subset \{f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}_{\geq 0}\}$  be a class of nonnegative functions, e.g., exponential or SoS. Then, the problem of searching for a valid barrier can be formulated as the solution of the following optimization problem:

$$\max_{B \in \mathcal{B}} 1 - (\eta + \beta H)$$

subject to:

$$\inf_{x \in X_u} B(x) \geq 1$$

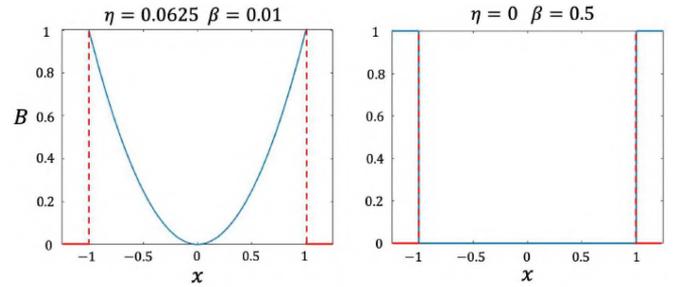


Fig. 1. For a stochastic system  $\mathbf{x}[k+1] = \mathbf{x}[k] + \mathbf{v}$ , where  $\mathbf{v} \sim \mathcal{N}(0, 0.1)$  is an independent additive zero-mean Gaussian noise with standard deviation 0.1. We consider  $X_s = [-1, 1]$  and  $X_u = [-0.25, 0.25]$ . Two different SBFs are shown: (left) a 4th degree polynomial as synthesized via SoS optimization for a time horizon  $H = 10$  using the approach in [17], and (right) the indicator function on the unsafe set for which  $\beta = \int_{(\infty, -1] \cup [1, \infty)} \mathcal{N}(x | 1, 0.1) dx$ . We obtain a lower bound on  $P_s$  of 0.8275 for SoS (left) and 0 for the indicator function (right).

$$\eta = \sup_{x \in X_0} B(x)$$

$$\beta = \sup_{x \in X_s} (\mathbb{E}_{x' \sim T(x' | x, \pi(x))}[B(x')] - B(x)). \quad (11)$$

In the case where  $\mathcal{B}$  is the class of exponential or SoS functions and dynamics function  $F$  is polynomial in  $\mathbf{x}$  and linear in  $\mathbf{v}$ , the above optimization problem can be reformulated as a convex optimization problem [17], [18]. However, in the more general setting, the optimization problem in (11) is nonconvex, requiring relaxations that typically lead to partitioning of  $X_s$ , or ad-hoc methods to solve it efficiently [13], [19], [20].

*Remark 4:* The solution to the optimization problem in (11) is also a solution to the DP in (7) and (8); hence, the obtained solution is an overapproximation of the solution for the unifying DP in (4) and (5). This simple formulation and its computational tractability are primarily enabled by the conditions on  $\eta$  defined in Theorem 2 and  $\beta$  in (9). However, these same  $\eta$  and  $\beta$  conditions are also the main sources of conservatism in lower bounding  $P_s$  in SBF methods.

*Remark 5:* Note that Theorem 2 only requires  $B$  to be almost surely continuous. That is,  $B$  can be discontinuous in a set of (Lebesgue) measure 0. Consequently,  $B$  can be taken to be a piecewise continuous functions. In practice, this can often be advantageous especially in the cases where  $F$  is nonlinear [21].

## A. Control Synthesis With SBFs

We now study how to find a policy  $\pi$  that maximizes  $P_s$  using SBFs. In particular, consider a stationary policy (feedback controller)  $\pi(\cdot | \theta) : X \rightarrow U$  parameterized in some parameters  $\theta \in \mathbb{R}^{n_\theta}$ . Then, control synthesis with SBFs can be performed by modifying the optimization problem in (11) to the following:

$$\max_{B \in \mathcal{B}, \theta \in \mathbb{R}^{n_\theta}} 1 - (\eta + \beta H)$$

subject to:

$$\inf_{x \in X_u} B(x) \geq 1$$

$$\eta = \sup_{x \in X_0} B(x)$$

$$\beta = \sup_{x \in X_s} (\mathbb{E}_{x' \sim T(x'|x, \pi(x|\theta))} [B(x')] - B(x)). \quad (12)$$

This optimization problem aims to simultaneously synthesize a barrier function  $B$  and a stationary policy (feedback controller)  $\pi$ . Unfortunately, because the expectation in the  $\beta$  (third) constraint depends on both  $\pi$  and  $B$ , the resulting optimization problem is generally nonconvex [22], [23]. To address this problem, recent approaches employ iterative methods [6], [8], [13]. They generally proceed by first finding a  $B$  for a fixed  $\pi$  and then updating  $\pi$  to maximize the lower bound on  $P_s$ , i.e.,  $1 - (\eta + \beta H)$ , for the fixed  $B$ . Other recent methods include employing machine learning algorithms [24].

As discussed above, there are two main sources of conservatism in bounding  $P_s$  using SBFs: 1) the choice of the barrier, and 2) the  $\beta$  term in (9), which is obtained as a uniform bound over the safe set. Both of these sources of errors can be mitigated by abstraction-based approaches, usually at the price of increased computational effort.

#### IV. DISCRETE ABSTRACTION

Another class of well-established approaches to compute  $P_s$  and a policy  $\pi$  that maximizes  $P_s$  is based on finite abstraction [8]. These approaches aim to numerically solve the continuous DP in (4) and (5) by discretizing  $X$ , while accounting for the error induced by discretization. We again start with the case where a deterministic Markov policy  $\pi$  is given, and consider the control synthesis case in Section IV-B.

Abstraction-based methods first partition the safe set  $X_s$  into  $n_p$  sets  $X_1, \dots, X_{n_p}$  and treat  $X_u$  as another discrete region (i.e.,  $X_{n_p+1} = X_u$ ), resulting in a total of  $n_p + 1$  regions. For  $k \in \{0, \dots, H-1\}$  and a given policy  $\pi$ , one can then define piecewise-constant functions

$$\gamma_k^\pi(x) = \begin{cases} \gamma_{k,1}^\pi, & \text{if } x \in X_1 \\ \vdots & \vdots \\ \gamma_{k,n_p}^\pi, & \text{if } x \in X_{n_p} \\ \gamma_{k,n_p+1}^\pi, & \text{otherwise} \end{cases}$$

recursively, for  $i \in \{1, \dots, n_p + 1\}$ , as<sup>4</sup>

$$\gamma_H^\pi(x) = \gamma_{H,i}^\pi := \max_{\bar{x} \in X_i} \mathbb{1}(\bar{x}, X_u) \quad (13)$$

$$\gamma_k^\pi(x) = \gamma_{k,i}^\pi := \max_{\bar{x} \in X_i} \left( \mathbb{1}(\bar{x}, X_u) + \mathbb{1}(\bar{x}, X_s) \sum_{j=1}^{n_p+1} \gamma_{k+1,j}^\pi T(X_j | \bar{x}, \pi_k(\bar{x})) \right). \quad (14)$$

Intuitively, each  $\gamma_{k,i}^\pi$  provides an upper bound on the worst-case probability that, starting from any point in  $X_i$  at time  $k$ , System (1) will reach the unsafe region before time  $H$ . Consequently,

<sup>4</sup> Note that under the assumption that  $\pi$  and  $T$  are locally continuous in  $X_i$ , we can replace the supremum with the maximum in the DP in (13) and (14). In fact, a continuous function on a compact set attains its maximum and minimum on this set.

due to the maximum operator in (13) and (14),  $\gamma_k^\pi(x)$  overapproximates  $V_k^\pi(x)$ , which is the probability of reaching the unsafe set starting from  $x$  in the next  $H - k$  time steps. This leads to the following theorem.

*Theorem 3:* Let  $\gamma_k^\pi$  be defined as in (13) and (14). Then, it holds that

$$P_s(X_s, x_0, H | \pi) \geq 1 - \gamma_0^\pi(x_0).$$

*Proof:* Because of Theorem 1 it is enough to show that for each  $k \in \{0, \dots, H\}$  and  $x \in X$ , it holds that  $V_k(x) \leq \gamma_k^\pi(x)$ . The case  $x \in X_i$  for  $X_i \cap X_u \neq \emptyset$  is trivially verified. Consequently, in what follows, we assume  $x \in X_i \subseteq X_s$ . The proof is by induction. Base case is  $k = H - 1$ , in this case, we have

$$\begin{aligned} V_{H-1}^\pi(x) &= T(X_u | x, \pi_{H-1}(x)) \\ &\leq \max_{x \in X_i} T(X_u | x, \pi_{H-1}(x)) \\ &\leq \max_{x \in X_i} \sum_{j=1}^{n_p+1} \gamma_{H,j}^\pi T(X_j | x, \pi_{H-1}(x)) \\ &= \gamma_{H-1}^\pi(x). \end{aligned}$$

For the induction case, assume  $x \in X_s$ . Then, under the assumption that  $V_{k+1}^\pi(x) \leq \gamma_{k+1}^\pi(x)$  (induction hypothesis), it follows that:

$$\begin{aligned} V_k^\pi(x) &= \mathbb{E}_{x' \sim T(\cdot | x, \pi_k(x))} [V_{k+1}^\pi(x')] \\ &\leq \mathbb{E}_{x' \sim T(\cdot | x, \pi_k(x))} [\gamma_{k+1}^\pi(x')] \\ &= \sum_{j=1}^{n_p+1} \gamma_{k+1,j}^\pi T(X_j | x, \pi_k(x)) \\ &= \gamma_k^\pi(x). \end{aligned}$$

□

*Remark 6:* Note that if, in (13) and (14), we replace the maximum operators with the minimum, then the solution of the resulting program returns a lower bound of  $V_0^\pi$  and consequently an upper bound on  $P_s$ .

Since  $\gamma_0^\pi$  is a piecewise-constant function and  $\gamma_{0,i}^\pi$  is the unsafety probability bound for region  $X_i$ , (13) and (14) can be viewed as a DP for a discrete abstraction of System (1). In that view, the abstraction is a (time-inhomogenous) MC  $\mathcal{M} = (Q, P_{\mathcal{M}})$ , where  $Q = \{q_1, \dots, q_{n_p+1}\}$  is the state space such that  $q_i$  represents region  $X_i$ , and  $P_{\mathcal{M}} : Q \times Q \times \mathbb{N}_{<H} \rightarrow [0, 1]$  is the transition probability function such that  $P_{\mathcal{M}}(q_i, q_j, k)$  is the probability of transitioning from partition  $X_i$  to  $X_j$  at time step  $k \in \mathbb{N}_{<H}$ , i.e.,

$$P_{\mathcal{M}}(q_i, q_j, k) = \begin{cases} T(X_j | x^*, \pi(x^*)), & \text{if } i \neq n_p + 1 \\ 1, & \text{if } i, j = n_p + 1 \\ 0, & \text{otherwise} \end{cases}$$

where  $x^* = \arg \max_{x \in X_i} \sum_{j=1}^{n_p+1} \gamma_{k+1,j}^\pi T(X_j | x, \pi_k(x))$ .

Computation of  $\gamma_k^\pi$  reduces to recursively solving the following maximization problem:

$$\max_{x \in X_i} \sum_{j=1}^{n_p+1} \gamma_{k+1,j}^\pi T(X_j | x, \pi(x)). \quad (15)$$

That is, we need to find the input point that maximizes a linear combination of the transition probabilities for System (1). Since a positive weighted combination of convex (concave) functions is still convex (concave), (15) can be solved exactly in the case  $T$  is either concave or convex in  $x$ . Nevertheless, in the more general case, obtaining exact solutions of this optimization problem becomes infeasible. In what follows, we consider a relaxation of (15) into a linear programming problem that leads to a well-known model, called interval MCs (IMCs) or decision processes (IMDPs), which are a class of uncertain Markov processes where the exact transition probabilities between states are unknown but known to lie in some independent intervals [25].

### A. Abstraction to Interval Markov Processes

To relax the maximization problem in (15), we note that for each  $x \in X_i$ ,  $T(\cdot | x, \pi(x))$  is a discrete probability distribution over regions  $X_1, \dots, X_{n_p}, X_u$ . Consequently, (15) weighs term  $\gamma_{k+1,j}^\pi$  with the probability of transitioning to any of the  $n_p + 1$  partitions of  $X$ . This observation leads to Proposition 1 below, where (15) is relaxed to a linear programming problem.

*Proposition 1 (See [25]):* For partition  $X_i \subseteq X_s$ , let  $\bar{\gamma}_{k,i}^\pi$  be the solution of the following linear programming problem:

$$\bar{\gamma}_{k,i}^\pi = \max_{t \in [0,1]^{n_p+1}} \sum_{j=1}^{n_p+1} \gamma_{k+1,j}^\pi t_j$$

subject to:

$$t_j \in \left[ \min_{x \in X_i} T(X_j | x, \pi_k(x)), \max_{x \in X_i} T(X_j | x, \pi_k(x)) \right]$$

$$\sum_{j=1}^{n_p+1} t_j = 1$$

where  $t_j$  is the  $j$ th component of vector  $t$ . Then, it holds that  $\bar{\gamma}_{k,i}^\pi \geq \gamma_{k,i}^\pi = \max_{x \in X_i} \sum_{j=1}^{n_p+1} \gamma_{k+1,j}^\pi T(X_j | x, \pi(x))$ .

In Proposition 1, we model the likelihood of transitioning between each pair of states as intervals of probabilities, where the intervals are composed of all the valid transition probabilities for each point in the starting partition, i.e., each  $t$  in the linear program in Proposition 1 represents a valid probability. In this view, the introduced relaxation is in fact a relaxation of the MC abstraction  $\mathcal{M}$  of System (1) to an IMC [25], [26]  $\mathcal{I} = (Q, \check{P}_T, \hat{P}_T)$ , where  $Q$  is the same as in MC  $\mathcal{M}$ , and  $\check{P}_T, \hat{P}_T : Q \times Q \rightarrow [0,1]$  are, respectively, the lower and upper bounds of the transition probability between each pair of states such that

$$\check{P}(q_i, q_j) = \begin{cases} \min_{x \in X_i} T(X_j | x, \pi_k(x)), & \text{if } i \neq n_p + 1 \\ 1, & \text{if } i, j = n_p + 1 \\ 0, & \text{otherwise} \end{cases}$$

$$\hat{P}(q_i, q_j) = \begin{cases} \max_{x \in X_i} T(X_j | x, \pi_k(x)), & \text{if } i \neq n_p + 1 \\ 1, & \text{if } i, j = n_p + 1 \\ 0, & \text{otherwise.} \end{cases}$$

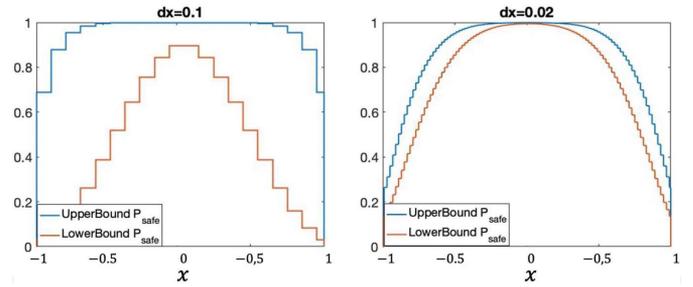


Fig. 2. We consider the same setting as in Fig. 1 and for each state in the safe set, we plot upper and lower bounds of the probability of remaining within the safe set  $[-1, 1]$  for ten time steps starting from that state using two IMC abstractions: one obtained by discretizing the safe set uniformly with discretization step  $dx = 0.1$  and the other with  $dx = 0.02$ . For the initial set  $X_0 = [-0.25, 0.25]$ , we obtain a lower bound on  $P_s$  of 0.756 for the case  $dx = 0.1$  (left) and 0.975 for  $dx = 0.02$  (right).

Consequently, it follows that, in solving the linear program (LP) in Proposition 1, we select the more conservative feasible distribution<sup>5</sup> with respect to probabilistic safety.

*Remark 7:* The optimization problem in Proposition 1 can be solved particularly efficiently due to the specific structure of the linear program. In particular, as shown in [25], one can simply order states based on the value of  $\gamma_{k+1}^\pi$  and then assign upper or lower bounds based on the ordering and on the fact that  $\sum_{j=1}^{n_p+1} t_j = 1$ . However, note that to solve the DP in (13) and (14), the resulting linear program problem needs to be solved  $H$  times for each of the  $n_p + 1$  states in  $Q$ .

*Remark 8:* Abstraction-based methods are numerical methods. As a consequence, their precision is dependent on the discretization of the state space. An example is illustrated in Fig. 2, where we show how for a finer partition, the precision of approach increases (a proof of convergence to  $P_s$  is given in Section IV). How to optimally discretize the safe set, while limiting the explosion of the partition size, is still an active area of research, see, e.g., [3] and [27]. We should also stress that the approach presented in this Section, while general, can sometimes be made tighter and less computationally expensive by adding to the optimization problem in Proposition 1 additional constraints derived from additional assumptions on System (1). For instance, this is the case if  $T$  has the product form, as shown in [28].

*Remark 9:* Note that an alternative approach for approximately solving (14) would be to associate a representative point  $x_i \in X_i$  to each partition and then consider the following approximation:

$$\max_{x \in X_i} \sum_{j=1}^{n_p+1} \gamma_{k+1,j}^\pi T(X_j | x, \pi_k(x))$$

$$\approx \sum_{j=1}^{n_p+1} \gamma_{k+1,j}^\pi T(X_j | x_i, \pi_k(x_i)). \quad (16)$$

The resulting abstraction leads to an MC with the same state space  $Q$ , and the transition probabilities between each pair of

<sup>5</sup> A feasible distribution  $t$  is a distribution that satisfies constraints of Proposition 1.

states are computed using representative points in the starting region, as shown in (16). Such numerical approaches have been widely studied in the literature [8], [29], [30], including also for continuous-time systems [31], and approaches to quantify the resulting error have been developed [30], [32], [33]. Compared to the IMDP or IMC approaches described before, these approaches have the advantage of not requiring the solution of any optimization problem over  $x$ . However, similar to SBF (see Theorem 2), the resulting error generally grows linearly with time [8], [30] and is commonly more conservative compared to abstractions to IMC (or IMDP) [11].

## B. Control Synthesis With IMDPs

For control policy computation, the set of controls or actions  $U$  should be included in the abstraction model. Hence, the abstraction becomes an IMDP  $\mathcal{I} = (Q, U, \check{P}, \hat{P})$ , where the transition probability bounds  $\check{P}, \hat{P} : Q \times U \times Q \rightarrow [0, 1]$  are now also functions of  $U$ , i.e., given  $q_i, q_j \in Q$  and  $a \in U$ ,  $\check{P}(q_i, a, q_j)$  and  $\hat{P}(q_i, a, q_j)$  are the lower and upper bounds transition probability from  $q_i$  to  $q_j$  under action  $a$ , respectively. An optimal policy via IMDP abstractions for region  $X_i \subseteq X_s$  can then be computed by solving the following value iteration, which combines Proposition 1 with (13) and (14), and where we iteratively seek for the action that minimizes the probability that  $\mathbf{x}_k$  enters the unsafe region  $X_u$

$$\bar{\gamma}_{H,i}^* = \max_{x \in X_i} \mathbb{1}(x, X_u) \quad (17)$$

$$\bar{\gamma}_{k,i}^* = \min_{a \in U} \max_{t \in T_a^i} \sum_{j=1}^{n_p+1} \bar{\gamma}_{k+1,j}^* t_j \quad (18)$$

where for  $i \leq n_p$ , we define

$$T_a^i = \left\{ t \in [0, 1]^{n_p+1} \mid t_j \in \left[ \inf_{x \in X_i} T(X_j \mid x, a) \right. \right.$$

$$\left. \left. \sup_{x \in X_i} T(X_j \mid x, a) \right], \sum_{j=1}^{n_p+1} t_j = 1 \right\}$$

and

$$T_a^{n_p+1} = \{t \in [0, 1]^{n_p+1} \mid t_j = 0 \text{ for } j \leq n_p, t_{n_p+1} = 1\}.$$

Note that in the inner maximization problem in (18), we seek a feasible distribution that maximizes the problem posed in Proposition 1.

Consequently, synthesizing a policy according to (17) and (18) boils down to recursively solving the following min-max optimization problem for each partition  $X_i \subseteq X_s$

$$\min_{a \in U} \max_{t \in T_a^i} \sum_{j=1}^{n_p+1} \bar{\gamma}_{k+1,j}^* t_j. \quad (19)$$

If  $U$  is discrete, then we simply need to apply Proposition 1  $|U|$  times, one for each action, and then take the action that minimizes the expression. If  $U$  is uncountable, then in the case where  $T$  is a convex or concave function of  $a$ , a solution to (19) can be found efficiently via convex optimization [34].

For the more general cases, where one cannot rely on convex optimization, a suboptimal policy can be found via heuristics or by discretizing  $U$  [12], but how to compute optimal policies for the general case still remains an open problem.

The following theorem, whose proof can be found in the Appendix, guarantees that the solution of the DP in (17) and (18) returns a lower bound on probabilistic safety for System (1) and that the resulting policy converges to the optimal policy in the limit of a fine enough partition.

*Theorem 4:* Let  $\bar{\gamma}_k^*$  be as defined in (17) and (18). Then, for any  $x \in X$ , it holds that  $\bar{\gamma}_0^*(x) \geq V_0^*(x)$ . Furthermore, assume that  $X_s$  is a bounded hyperrectangle, and that it is discretized uniformly. Then, for every  $x \in X$ , in the limit of  $n_p \rightarrow \infty$ , we have that  $\bar{\gamma}_0^*(x)$  converges uniformly to  $V_0^*(x)$ .

*Remark 10:* Note that in the proof of Theorem 4, reported in the Appendix, we characterize the convergence rate of (17) and (18) to the optimal policy and to  $P_s$ . Nevertheless, the resulting convergence rate is generally conservative and to bound the error of abstraction-based methods, a much less conservative approach is as follows. For a given policy  $\pi$ , possibly obtained by solving (17) and (18), solve (13) and (14), to get a lower bound of  $P_s(X_s, x, H \mid \pi)$ . Then, an upper bound can be similarly obtained by solving the following DP:

$$\gamma_H^{\pi,L}(x) = \gamma_{H,i}^{\pi,L} := \min_{x \in X_i} \mathbb{1}(x, X_u) \quad (20)$$

$$\gamma_k^{\pi,L}(x) = \gamma_{k,i}^{\pi,L} := \min_{t \in T_{\pi_k(x)}^i} \sum_{j=1}^{n_p+1} \bar{\gamma}_{k+1,j}^* t_j \quad (21)$$

where  $T_{\pi_k(x)}^i$  is defined as in (18). Intuitively, in the above problem, we iteratively seek a feasible distribution that minimizes the probability of reaching the unsafe region. Thus,  $1 - \gamma_k^{\pi,L}(x)$  returns an upper bound on  $1 - P_s(X_s, x, H \mid \pi)$ , as illustrated in Fig. 2.

## V. ABSTRACTIONS VERSUS BARRIERS: PROS AND CONS

Sections III and IV provide a unifying perspective on SBF-based and abstraction-based methods under the framework of DP. Specifically, these sections demonstrate that these methods are different approximation techniques for solving the same DP problem described in (2) and (3) [or equivalently in (4) and (5)]. Hence, we can now fairly compare the pros and cons of the two approaches. Namely, we focus on theoretical analysis with respect to the following properties: soundness, optimality, accuracy, computational effort, and scalability.

- 1) *Soundness:* As we show in Theorems 2 and 4, both approaches are guaranteed to return a valid lower bound on  $P_s$ . Hence, both methods are sound.
- 2) *Optimality guarantees:* As the  $\beta$  term in Theorem 2 is obtained by the supremum expected change for all points in  $X_s$ , the policy synthesized via SBF-based approaches is necessarily suboptimal with a few exceptions (e.g., when  $\beta = 0$ ). For this reason, there are no convergence guarantees of the safety certificate to  $P_s$  for SBFs, in general. In contrast, for IMDP-based approaches, as proved in Theorem 4, for a fine enough partition, both the policy

TABLE I

COMPARISON OF LOWER BOUNDS ON  $P_s$  OBTAINED VIA VARIOUS SBF-BASED AND ABSTRACTION-BASED METHODS FOR THE SYSTEM IN FIG. 1:  $\mathbf{x}[k+1] = \mathbf{x}[k] + \mathbf{v}$ , WITH  $\mathbf{v} \sim \mathcal{N}(0, 0.1)$ , SAFE SET  $X_s = [-1, 1]$ , INITIAL SET  $X_0 = [-0.25, 0.25]$ , AND  $H = 10$

Method	$P_s$
SBF (SoS degree 2)	0.001
SBF (SoS degree 4)	0.8275
SBF (SoS degree 8)	0.8379
SBF (SoS degree 12)	0.8386
SBF (SoS degree 16)	0.8386
SBF (neural network two hidden layers with 32 neurons each)	0.802
SBF (indicator function)	0
IMC ( $dx = 0.1$ )	0.751
IMC ( $dx = 0.02$ )	0.975
True probabilistic safety	<b>0.9884</b>

For the IMC methods,  $dx$  is the width of the cells of a uniform grid that partitions  $X_s$ . For SBF, in the case of sos we use the approach presented in [17] with various degrees of the SoS, polynomial as reported in the table and with lagrangian multipliers of degree 4. In the case of neural network SBF, we used the approach developed in [19] and Rectified Linear Unit (ReLU) activation functions. For the reference, true  $P_s$  is also reported.

and the certificate of safety converge to the optimal values for System (1).

- 3) *Accuracy*: As SBF-based approaches have no guarantees of convergence to the true  $P_s$  (see discussion above), IMDP-based methods can generally provide less conservative bounds. However, as shown in Theorem 4, this comes at the price of a fine enough partition. To illustrate this point, Table I compares the bounds of  $P_s$  obtained for the system in Fig. 1 by using the state-of-the-art SBF methods and IMC abstractions of various sizes. It is possible to observe that even barriers parameterized as neural networks or SoS polynomials of high degrees admit bounds that are far from the true probability even for this simple 1-D linear example. This is in contrast with abstraction-based methods that converge to the true probability in the limit of arbitrarily large grid.
- 4) *Computational effort*: IMDP abstraction methods always require to partition the state space and to solve a DP over the resulting discretized space. Consequently, IMDP approaches necessitate to solve a number of linear programs that depends linearly on both the time horizon and number of states in the partition, which itself is exponential in the dimension of the state space of System (1). While, as explained in Remark 7, each of these problems can be solved efficiently and they are highly parallelizable, SBF-based approaches only require to solve a single (in general nonconvex) optimization process. Furthermore, in the case of linear or polynomial dynamics, the resulting optimization problem can be often solved without any partitioning by employing existing tools from convex optimization.
- 5) *Scalability*: In terms of scalability with respect to the dimensionality of  $X$ , abstraction-based methods face the state-space explosion problem since the size of discretization grows exponentially with dimensionality. Even if compositional approaches and methods based on model order reduction have been developed to alleviate this problem [8], [35], still the issue remains. Similarly, as

dimensionality grows, existing SBF methods face significant increase in complexity, depending on the parameterization of SBF. For instance, SoS-based approaches [17] experience an exponential blow-up in the number of basis functions, e.g., number of monomials, leading to exponential growth in the optimization parameters. Alternative approaches that parameterize an SBF as a neural network [19], [20], [24] also experience exponential complexity in the dimension of  $X$  to prove that a neural network is a valid SBF and to compute  $\eta$  and  $\beta$ , as defined in Theorem 2. Hence, scaling to high-dimensional systems remains a key challenge in safety verification of stochastic systems.

A greatly simplified summary of the discussion above is presented in Table II. Note that the table reflects the theoretical analysis rather than empirical results. Also note that the computational effort comparison is based on the number of optimization problems that each method is required to solve. As explained above, this is not necessarily equivalent to time complexity of the algorithms.

From the above discussion, it becomes clear how abstraction-based methods and SBFs are complementary approaches, with a tradeoff between computational demands and accuracy and/or flexibility in performing policy synthesis. Consequently, the choice of the method should depend on the particular application.

We should also mention that often one is interested in properties beyond safety and reachability, such as temporal logic properties. While this problem has been well studied for abstraction-based methods [10], where it has been shown to follow from the results for safety and reachability, only few works have recently considered this problem for SBFs [23].

## VI. CONCLUSION AND FUTURE DIRECTIONS

Safety analysis of stochastic systems is a major problem in today's world, especially as autonomous cyber-physical systems become increasingly integral. Existing methods for such analysis are based on either SBFs or discrete abstraction. Our analysis shows that both of these methods are approximations of the same stochastic DP problem. This perspective unifies these approaches and allows for a fair comparison, revealing that the methods are complementary, each with its own strengths and weaknesses. Consequently, the choice of approach should depend on the specific application under consideration.

Our demonstration of the effectiveness of these methods in solving the safety DP problem raises several open research questions. The first question arises from the connection established between SBFs and abstraction-based methods and the fact that they are complementary. Specifically, given this knowledge, is it possible to devise new approaches that integrate their strengths, mitigate their weaknesses, and improve the scalability of existing methods? The second question stems from the observation that only a few approaches for controller synthesis algorithms exist, and they are generally limited to discrete or convex settings [17], [23], [34]. Therefore, there is a need for more research to tackle the control synthesis problem with better and more general algorithms. Another interesting question is related to

TABLE II  
SUMMARY OF PROS AND CONS OF STOCHASTIC BARRIER FUNCTION AND DISCRETE ABSTRACTION METHODS

Approach	Soundness	Optimality	Accuracy	Computational effort*	Nonlinear dynamics
Stochastic Barrier Function	+	−	−	+	+
Discrete Abstraction	+	+	+	−	+

\*Computational effort is with respect the number of required optimization problems.

what this article does not consider. That is, in this article, we only consider systems with uncertainty rooted in the noise in the dynamics of the system (i.e., aleatoric uncertainty); however, often, the noise characteristics or the dynamics of the system are themselves uncertain. This setting has recently attracted interest from the control and machine learning communities, with approaches based on abstractions to robust MDPs [36] and scenario-based optimization [21], [37]. However, how to provide guarantees of safety and synthesize controllers for this more general setting, especially in the case where the noise is not additive, is still an open question.

This work lays a theoretical foundation for computing safety guarantees for stochastic systems. By establishing a common ground for the development of both SBFs and abstraction-based methods, we hope to stimulate new research at the intersection of these frameworks. We believe these methods may eventually enable the attainment of safety guarantees for real-world control systems.

## APPENDIX PROOFS

### A. Proof of Theorem 1

Define  $P_u(X_s, x_0, H | \pi) := 1 - P_s(X_s, x_0, H | \pi)$ . Then, to prove the result it is enough to show that

$$\inf_{\pi \in \Pi} P_u(X_s, x_0, H | \pi) = V_0^*(x_0).$$

In the proof, we are going to require the following lemma, which, for  $\pi \in \Pi$  allows us to represent  $P_u(X_s, x_0, H | \pi)$  via DP for history-dependent, possibly random, policies.

*Lemma 2:* For any  $\pi = (\pi_0, \dots, \pi_{H-1}) \in \Pi$  and  $k \in \{0, \dots, H\}$ , let  $\bar{V}_k^\pi : X^{k+1} \rightarrow [0, 1]$  be defined as

$$\bar{V}_H^\pi(x_0, \dots, x_H) = \mathbb{1}(x_H, X_u)$$

$$\bar{V}_k^\pi(x_0, \dots, x_k) = \mathbb{1}(x_k, X_u) +$$

$$\mathbb{1}(x_k, X_s) \mathbb{E}_{x' \sim T(\cdot | x_k, \pi_k(x_0, \dots, x_k))} [\bar{V}_{k+1}^\pi(x_0, \dots, x_k, x')].$$

Then, it holds that

$$P_u(X_s, x_0, H | \pi) = \bar{V}_0^\pi(x_0).$$

*Proof:* The proof is by induction over the time index  $k$ . Define

$$P_u^{[x_0, \dots, x_k]}(X_s, [k, H] | \pi)$$

$$= P[\exists k' \in [k, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | \mathbf{x}_0 = x_0, \dots, \mathbf{x}_k = x_k, \pi].$$

Then, base case is for  $k = H$ , that is  $P_u^{[x_0, \dots, x_H]}(X_s, [H, H] | \pi) = \bar{V}_H^\pi(x_0, \dots, x_H)$ , which follows trivially. Then, assuming that

$$P_u^{[x_0, \dots, x_{k+1}]}(X_s, [k+1, H] | \pi) = \bar{V}_{k+1}^\pi(x_0, \dots, x_{k+1})$$

it holds that

$$\begin{aligned} & P_u^{[x_0, \dots, x_k]}(X_s, [k, H] | \pi) \\ &= P[\exists k' \in [k, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | \mathbf{x}_0 = x_0, \dots, \mathbf{x}_k = x_k, \pi] \\ &= P[\mathbf{x}_k \in X_u \vee \exists k' \in [k+1, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | x_0 = x_0, \dots, \mathbf{x}_k = x_k, \pi] \\ &= P[\mathbf{x}_k \in X_u | \mathbf{x}_k = x_k] \\ &\quad + P[\exists k' \in [k+1, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | x_0 = x_0, \dots, x_k = x_k, \pi] \\ &\quad - P[\mathbf{x}_k \in X_u \wedge \exists k' \in [k+1, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | \mathbf{x}_0 = x_0, \dots, \mathbf{x}_k = x_k, \pi] \\ &= \mathbb{1}(x_k, X_u) \\ &\quad + P[\exists k' \in [k+1, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | \mathbf{x}_0 = x_0, \dots, \mathbf{x}_k = x_k, \pi] \\ &\quad - \mathbb{1}(x_k, X_u) \\ &= P[\exists k' \in [k+1, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | \mathbf{x}_0 = x_0, \dots, \mathbf{x}_k = x_k, \pi] \\ &= \mathbb{1}(x_k, X_u) + (1 - \mathbb{1}(x_k, X_u)) \\ &\quad P[\exists k' \in [k+1, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | \mathbf{x}_0 = x_0, \dots, \mathbf{x}_k = x_k, \pi] \\ &= \mathbb{1}(x_k, X_u) + \mathbb{1}(x_k, X_s) P[\exists k' \in [k+1, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | \mathbf{x}_0 = x_0, \dots, \mathbf{x}_k = x_k, \pi] \\ &= \mathbb{1}(x_k, X_u) + \mathbb{1}(x_k, X_s) \int_X P[\exists k' \in [k+1, H] \text{ s.t. } \mathbf{x}_{k'} \in X_u | x_0 = x_0, \dots, x_k = x_k, \mathbf{x}_{k+1} = x', \pi] P[\mathbf{x}_{k+1} = x' | \mathbf{x}_0 = x_0, \dots, \mathbf{x}_k = x_k, \pi] dx' \\ &= \mathbb{1}(x_k, X_u) + \mathbb{1}(x_k, X_s) \int_X \bar{V}_{k+1}^\pi(x_0, \dots, x_k, x') \\ &\quad P[x_{k+1} = x'] \\ &= \bar{V}_k^\pi(x_0, \dots, x_k). \end{aligned}$$

□

We can now prove the main statement. As  $\Pi^{M,D} \subset \Pi$ , it holds that

$$\inf_{\pi \in \Pi} P_u(X_s, X_0, H | \pi) \leq \inf_{\pi \in \Pi^{M,D}} P_u(X_s, X_0, H | \pi).$$

Consequently, the proof is concluded if we can show that it also holds that  $\inf_{\pi \in \Pi} P_u(X_s, X_0, H | \pi) \geq \inf_{\pi \in \Pi^{M,D}} P_u(X_s, X_0, H | \pi)$ . This can be done by induction over the time index  $k \in \{0, H\}$ . The base case is  $k = H$ , which follows trivially as  $\forall x_0 \in X$ , it holds that for any  $\pi \in \Pi$

$$\bar{V}_H^\pi(x_0, \dots, x_H) = V_H^*(x_H) = \mathbb{1}(x_H, X_u)$$

which is independent of the action and of the previous states. Now assume that  $\forall (x_0, \dots, x_{k+1}) \in X^{k+1}$ ,  $\inf_{\pi \in \Pi} V_{k+1}^\pi(x_0, \dots, x_{k+1}) \geq V_{k+1}^*(x_{k+1})$ , then it holds that

$$\begin{aligned} & \inf_{\pi \in \Pi} \bar{V}_k^\pi(x_0, \dots, x_k) \\ &= \inf_{\pi \in \Pi} \mathbb{1}(x_k, X_u) \\ & \quad + \mathbb{1}(x_k, X_s) \mathbb{E}_{x' \sim T(\cdot | x_k, \pi_k(x_0, \dots, x_k))} [\bar{V}_{k+1}^\pi(x_0, \dots, x_k, x')] \\ & \geq \mathbb{1}(x_k, X_u) + \\ & \quad \mathbb{1}(x_k, X_s) \inf_{\pi \in \Pi} \mathbb{E}_{x' \sim T(\cdot | x_k, \pi_k(x_0, \dots, x_k))} [V_{k+1}^*(x')] \\ &= \mathbb{1}(x_k, X_u) + \mathbb{1}(x_k, X_s) \inf_{a \in U} \mathbb{E}_{x' \sim T(\cdot | x_k, a)} [V_{k+1}^*(x')] \\ &= V_k^*(x_k). \end{aligned}$$

## B. Proof of Theorem 4

We recall that  $\bar{\gamma}_k^*$  is the solution of the following DP problem:

$$\begin{aligned} \bar{\gamma}_H^*(x) &= \bar{\gamma}_{H,i}^* := \max_{x \in X_i} \mathbb{1}(x, X_u) \\ \bar{\gamma}_k^*(x) &= \bar{\gamma}_{k,i}^* := \min_{a \in U} \max_{t \in T_a^i} \left( \mathbb{1}(x, X_u) \right. \\ & \quad \left. + \mathbb{1}(x, X_s) \sum_{j=1}^{n_p+1} \bar{\gamma}_{k+1,j}^* t_j \right) \end{aligned}$$

where for  $i \leq n_p$

$$T_a^i = \left\{ t \in [0, 1]^{n_p+1} \mid t_j \in \left[ \inf_{x \in X_i} T(X_j | x, a) \right. \right. \\ \left. \left. \sup_{x \in X_i} T(X_j | x, a) \right], \sum_{j=1}^{n_p+1} t_j = 1 \right\}$$

and for  $i = n_p + 1$ , we have  $T_a^{n_p+1} = \{t \in [0, 1]^{n_p+1} \mid t_j = 0 \text{ for } j \leq n_p \wedge t_{n_p+1} = 1\}$ . Then, we first have to show that for any  $k \in \{0, \dots, H\}$  and  $x \in X$  it holds that  $\bar{\gamma}_0^*(x) \geq V_0^*(x)$ . This can be easily proved by contradiction. In fact, assume that this statement is not true, then there must exist a strategy  $\pi$  such that for  $x \in X$  and  $k \in \{0, \dots, H\}$  it holds that  $\bar{\gamma}_k^\pi(x) \leq V_k^\pi(x)$ . However, this contradicts Theorem 3 and Proposition 1, thus concluding the proof that  $\bar{\gamma}_0^*(x) \geq V_0^*(x)$ . A consequence of this result is that for any  $\bar{x} \in X_i$ , if we consider  $a^* \in \arg \min_{a \in U} \mathbb{E}_{x' \sim T(\cdot | \bar{x}, a)} [V_{k+1}^*(x')]$ , it holds that

$$\begin{aligned} 0 & \leq \min_{a \in U} \max_{x \in X_i} \mathbb{E}_{x' \sim T(\cdot | x, a)} [\bar{\gamma}_{k+1}^*(x')] - \mathbb{E}_{x' \sim T(\cdot | \bar{x}, a^*)} [V_{k+1}^*(x')] \\ & \leq \max_{x \in X_i} \mathbb{E}_{x' \sim T(\cdot | x, a^*)} [\bar{\gamma}_{k+1}^*(x')] - \mathbb{E}_{x' \sim T(\cdot | \bar{x}, a^*)} [V_{k+1}^*(x')]. \end{aligned}$$

We will use this result to prove that for any  $\epsilon > 0$  there exists a partition of  $X$  in  $n_p + 1$  regions such that  $\forall x \in X$ ,  $\bar{\gamma}_0^*(x) - V_0^*(x) \leq \epsilon$ . That is  $\bar{\gamma}_0^*(x)$  converges uniformly to  $V_0^*(x)$ , thus concluding the proof.

Without any loss of generality, assume that  $X_s$  is uniformly partitioned in  $n_p$  hypercubes with the edge of size  $\delta$ . Further, consider an additional set in the partition  $X_{n_p+1} = X_u$ . Then, the proof is as follows. We first show that for any  $\bar{\epsilon} > 0$  such that  $\max_{x \in X} (\bar{\gamma}_{k+1}^*(x) - V_{k+1}^*(x)) \leq \bar{\epsilon}$ , then for all  $x \in X$  it must hold that  $\bar{\gamma}_k^*(x) - V_k^*(x) \leq \bar{\epsilon} + L\delta$ , where  $L$  is such that for any  $x, x' \in X_s$ , and  $X_j \subseteq X$

$$L|x - x'|_\infty \geq \max_{a \in U} |T(X_j | x, a) - T(X_j | x', a)|$$

which is guaranteed to exist because of the Lipschitz continuity of  $T(X_j | x, a^*)$  w.r.t.  $x$ . Then, we derive a bound for  $\bar{\epsilon}$  by considering the case  $k = H - 1$ . We start with the first part of the proof.

As mentioned, we start by assuming that  $\forall x \in X$ ,  $\bar{\gamma}_{k+1}^*(x) - V_{k+1}^*(x) \leq \bar{\epsilon}$ . Under this assumption, for  $x \in X_i$  with  $i \leq n_p$  select  $a^* \in \arg \min_{a \in U} \mathbb{E}_{x' \sim t(x|a)} [V_{k+1}^*(x')]$ , then the following holds:

$$\begin{aligned} & \bar{\gamma}_k^*(x) - V_k^*(x) \\ & \leq \max_{t \in T_{a^*}^i} \sum_{j=1}^{n_p+1} \bar{\gamma}_{k+1,j}^* t_j - \mathbb{E}_{x' \sim T(\cdot | \bar{x}, a^*)} [V_{k+1}^*(x')] \\ & \leq \max_{t \in T_{a^*}^i} \sum_{j=1}^{n_p+1} \bar{\gamma}_{k+1,j}^* t_j - \sum_{j=1}^{n_p+1} \min_{x' \in X_j} V_{k+1}^*(x') T(X_j | x, a^*) \\ & \leq \max_{t \in T_{a^*}^i} \sum_{j=1}^{n_p+1} \left( \min_{x' \in X_j} V_{k+1}^*(x') + \bar{\epsilon} \right) t_j \\ & \quad - \sum_{j=1}^{n_p+1} \min_{x' \in X_j} V_{k+1}^*(x') T(X_j | x, a^*) \\ & \leq \bar{\epsilon} + \max_{t \in T_{a^*}^i} \sum_{j=1}^{n_p+1} \min_{x' \in X_j} V_{k+1}^*(x') (t_j - T(X_j | x, a^*)) \\ & \leq \bar{\epsilon} + \max_{t \in T_{a^*}^i} \max_{j \in \{1, \dots, n_p+1\}} |t_j - T(X_j | x, a^*)| \\ & \leq \bar{\epsilon} + L\delta \end{aligned}$$

where last two inequalities hold because  $\sum_{j=1}^{n_p+1} \min_{x' \in X_j} V_{k+1}^*(x') \leq \min_{x' \in X_j} \sum_{j=1}^{n_p+1} V_{k+1}^*(x') \leq 1$  and because  $t$  is a feasible distribution satisfying the constraints of Proposition 1. Note that the case  $x \in X_{n_p+1} = X_u$  the inequality follows trivially as in that case  $\bar{\gamma}_k^*(x) = \mathbb{1}(x, X_u) = V_k^*(x)$ . What is left to do in order to complete the proof is to obtain a bound of  $\bar{\epsilon}$  as a function of  $\delta$ . We can do that by noticing that

$$\begin{aligned} & \bar{\gamma}_{H-1}^*(x) - V_{H-1}^*(x) = \\ & \max_{t \in T_{a^*}^i} (t_{n_p+1} - T(X_u | x, a^*)) \leq L\delta. \end{aligned}$$

Consequently, to guarantee an error smaller than  $\epsilon > 0$ , we can take  $\delta \leq \frac{\epsilon}{HL}$ , which results in  $n_p \geq \left(\frac{HL}{\epsilon}\right)^n + 1$ , where  $l = \max_{x \in X} |x - x'|_\infty$ .

## REFERENCES

- [1] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annu. Rev. Control, Robot., Auton. Syst.* vol. 1, pp. 187–210, 2018.
- [2] P. E. Dupont et al., "A decade retrospective of medical robotics research from 2010 to 2020," *Sci. Robot.*, vol. 6, no. 60, 2021, Art. no. eabi8017.
- [3] S. Adams, M. Lahijanian, and L. Laurenti, "Formal control synthesis for stochastic neural network dynamic models," *IEEE Contr. Syst. Lett.*, vol. 6, pp. 2858–2863, 2022.
- [4] A. Clark, "Control barrier functions for stochastic systems," *Automatica*, vol. 130, 2021, Art. no. 109688.
- [5] H. J. Kushner, "Stochastic stability and control," New York, NY, USA: Academic Press, 1967.
- [6] S. Prajna, A. Jadbabaie, and G. J. Pappas, "A framework for worst-case and stochastic safety verification using barrier certificates," *IEEE Trans. Autom. Control*, vol. 52, no. 8, pp. 1415–1428, Aug. 2007.
- [7] H. J. Kushner and D. S. Clark, *Stochastic Approximation Methods for Constrained and Unconstrained Systems*, vol. 26. Berlin, Germany: Springer, 2012.
- [8] A. Lavaei, S. Soudjani, A. Abate, and M. Zamani, "Automated verification and synthesis of stochastic hybrid systems: A survey," *Automatica*, vol. 146, 2022, Art. no. 110617.
- [9] M. Lahijanian, S. B. Andersson, and C. Belta, "Approximate Markovian abstractions for linear stochastic systems," in *Proc. IEEE Conf. Decis. Control*, Maui, HI, USA, 2012, pp. 5966–5971.
- [10] M. Lahijanian, S. B. Andersson, and C. Belta, "Formal verification and synthesis for discrete-time stochastic systems," *IEEE Trans. Autom. Control*, vol. 60, no. 8, pp. 2031–2045, Aug. 2015.
- [11] N. Cauchi, L. Laurenti, M. Lahijanian, A. Abate, M. Kwiatkowska, and L. Cardelli, "Efficiency through uncertainty: Scalable formal synthesis for stochastic hybrid systems," in *Proc. 22nd ACM Int. Conf. Hybrid Syst.: Comput. Control*, Montreal, QC, Canada, 2019, pp. 240–251.
- [12] M. Dutreix, J. Huh, and S. Coogan, "Abstraction-based synthesis for stochastic systems with omega-regular objectives," *Nonlinear Analysis: Hybrid Syst.*, vol. 45, 2022, Art. no. 101204.
- [13] R. Mazouz, K. Muvvala, A. Ratheesh L. Babu Laurenti, and M. Lahijanian, "Safety guarantees for neural network dynamic systems via stochastic barrier functions," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 9672–9686.
- [14] D. Bertsekas and S. E. Shreve, *Stochastic Optimal Control: The Discrete-Time Case*, vol. 5. Belmont, MA, USA: Athena Sci., 1996.
- [15] A. Abate, M. Prandini, J. Lygeros, and S. Sastry, "Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems," *Automatica*, vol. 44, no. 11, pp. 2724–2734, 2008.
- [16] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, Aug. 2017.
- [17] C. Santoyo, M. Dutreix, and S. Coogan, "A barrier function approach to finite-time stochastic system verification and control," *Automatica*, vol. 125, 2021, Art. no. 109439.
- [18] J. Steinhart and R. Tedrake, "Finite-time regional verification of stochastic non-linear systems," *Int. J. Robot. Res.*, vol. 31, no. 7, pp. 901–923, 2012.
- [19] F. B. Mathiesen, S. C. Calvert, and L. Laurenti, "Safety certification for stochastic systems via neural barrier functions," *IEEE Contr. Syst. Lett.*, vol. 7, pp. 973–978, 2022.
- [20] A. Abate, M. Giacobbe, and D. Roy, "Learning probabilistic termination proofs," in *Proc. Comput. Aided Verification: 33 rd Int. Conf., CAV 2021*, Virtual Event, 2021, pp. 3–26.
- [21] F. B. Mathiesen, L. Romao, S. C. Calvert, A. Abate, and L. Laurenti, "Inner approximations of stochastic programs for data-driven stochastic barrier function design," in *Proc. 62nd IEEE Conf. Decis. Control*, 2023, pp. 3073–3080.
- [22] A. Agrawal and K. Sreenath, "Discrete control barrier functions for safety-critical control of discrete systems with application to bipedal robot navigation," in *Proc. Robot.: Sci. Syst.*, 2017, pp. 1–10.
- [23] P. Jagtap, S. Soudjani, and M. Zamani, "Formal synthesis of stochastic systems via control barrier certificates," *IEEE Trans. Autom. Control*, vol. 66, no. 7, pp. 3097–3110, Jul. 2021.
- [24] D. Vzikelic, M. Lechner, T. A. Henzinger, and K. Chatterjee, "Learning control policies for stochastic systems with reach-avoid guarantees," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 11926–11935.
- [25] R. Givan, S. Leach, and T. Dean, "Bounded-parameter markov decision processes," *Artif. Intell.*, vol. 122, no. 1/2, pp. 71–109, 2000.
- [26] A. Nilim and L. El Ghaoui, "Robust control of Markov decision processes with uncertain transition matrices," *Operations Res.*, vol. 53, no. 5, pp. 780–798, 2005.
- [27] M. Dutreix and S. Coogan, "Efficient verification for stochastic mixed monotone systems," in *Proc. ACM/IEEE 9th Int. Conf. Cyber-Phys. Syst.*, 2018, pp. 150–161.
- [28] F. B. Mathiesen, S. Haesaert, and L. Laurenti, "Scalable control synthesis for stochastic systems via structural IMDP abstractions," in *Proc. 28th ACM Int. Conf. Hybrid Syst.: Comput. Control*, 2025, pp. 1–12.
- [29] D. Bertsekas, "Convergence of discretization procedures in dynamic programming," *IEEE Trans. Autom. Control*, vol. 20, no. 3, pp. 415–419, Jun. 1975.
- [30] A. Abate, J.-P. Katoen, J. Lygeros, and M. Prandini, "Approximate model checking of stochastic hybrid systems," *Eur. J. Control*, vol. 16, no. 6, pp. 624–641, 2010.
- [31] H. J. K. Kushner, H. J. Kushner, P. G. Dupuis, and P. Dupuis, *Numerical Methods for Stochastic Control Problems in Continuous Time*, vol. 24. Berlin, Germany: Springer, 2001.
- [32] A. Abate, A. D'Innocenzo, and M. D. Di Benedetto, "Approximate abstractions of stochastic hybrid systems," *IEEE Trans. Autom. Control*, vol. 56, no. 11, pp. 2688–2694, Nov. 2011.
- [33] Z. Esmaeil, S. Soudjani, C. Gevaerts, and A. Abate, "FAUST 2: Formal abstractions of uncountable-state stochastic processes," in *Proc. 21st Int. Conf. Tools Algorithms Construction Anal. Syst.*, 2015, pp. 272–286.
- [34] G. Delimpaltadakis, M. Lahijanian, M. Mazo Jr, and L. Laurenti, "Interval Markov decision processes with continuous action-spaces," in *Proc. 26th ACM Int. Conf. Hybrid Syst.: Comput. Control*, 2023, pp. 1–10.
- [35] A. Lavaei, S. Soudjani, E. Frazzoli, and M. Zamani, "Constructing MDP abstractions using data with formal guarantees," *IEEE Contr. Syst. Lett.*, vol. 7, pp. 460–465, 2022.
- [36] I. Gracia, D. Boskos, L. Laurenti, and M. Mazo Jr, "Distributionally robust strategy synthesis for switched stochastic systems," in *Proc. 26th ACM Int. Conf. Hybrid Syst.: Comput. Control*, 2023, pp. 1–10.
- [37] T. Badings, L. Romao, A. Abate, and N. Jansen, "Probabilities are not enough: Formal controller synthesis for stochastic dynamical models with epistemic uncertainty," in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, no. 12, 2023, pp. 14701–14710.



**Luca Laurenti** (Member, IEEE) received the Ph.D. degree in formal methods from the Department of Computer Science, University of Oxford, Oxford, U.K., in 2018.

He is currently an Assistant Professor with the Delft Center for Systems and Control, TU Delft, Delft, The Netherlands, and Co-Director of the HERALD Delft AI Lab. He was a Member of the Trinity College, University of Oxford. He has a background in stochastic systems, control theory, formal methods, and artificial intelligence.

His research focuses on developing data-driven systems provably robust to interactions with a dynamic and uncertain world.



**Morteza Lahijanian** (Member, IEEE) received the B.S. degree in bioengineering from the University of California, Berkeley, Berkeley, CA, USA, in 2005, the M.S. degree in mechanical engineering (control theory and robotics) from Boston University, in 2009, and the Ph.D. degree in mechanical engineering (control theory and robotics) from Boston University, Boston, MA, USA, in 2013.

He is currently an Associate Professor with the Aerospace Engineering Sciences Department, an Affiliated Faculty with the Computer Science department and Robotics Program, and the Director of the Assured, Reliable, and Interactive Autonomous (ARIA) Systems Group, University of Colorado Boulder, Boulder, CO, USA. He was a Postdoctoral Scholar with Department of Computer Science, Rice University, Houston, TX, USA. Prior to joining CU Boulder, he was a Research Scientist with the Department of Computer Science, University of Oxford, Oxford, USA. His research interests include the areas of control theory, stochastic hybrid systems, formal methods, machine learning, and game theory with applications in robotics, particularly, motion planning, strategy synthesis, model checking, and human-robot interaction.

Dr. Lahijanian was the recipient of the Ella Mae Lawrence R. Quarles Physical Science Achievement Award, Jack White Engineering Physics Award, Outstanding Junior Faculty Award, and NSF GK-12 Fellowship.