Risk-Aware Robotics: Tail Risk Measures in Planning, Control, and Verification

Prithvi Akella*, Anushri Dixit*, Mohamadreza Ahmadi, Lars Lindemann, Margaret P. Chapman, George J. Pappas, Aaron D. Ames, and Joel W. Burdick

P. Akella (prithvi.akella@gmail.com), A. D. Ames (ames@caltech.edu), and J. W. Burdick (jwb@robotics.caltech.edu) are with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA, USA.
A. Dixit (anushri.dixit@princeton.edu) is with the Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, NJ, USA.
M. Ahmadi (reza.ahmadi@gatik.ai) is with Gatik AI, 161 E Evelyn Ave, Mountain View, CA 94041.
L. Lindemann (llindema@usc.edu is with the Department of Computer Science, University of Southern California, Los Angeles, CA, USA.
M. P. Chapman (mchapman@ece.utoronto.ca) is with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada.
G. J. Pappas (pappasg@seas.upenn.edu) is with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA, USA.
*Both authors contributed equally.

Why is the consideration of risk integral to robotics? Consider the scenario of a rescue robot designed to navigate through a constrained environment to locate and rescue victims. Despite limited visibility and a ground that is potentially difficult to traverse, the robot is required to assess whether to execute a planned route or to seek an alternative route. Delays in finding an alternative route could negatively impact the victim's well-being while opting for the planned route could potentially lead to the robot getting stuck. How should the robot gauge these competing risks? The issue of risk assessment is not new to roboticists and control engineers, and in practical applications, it is usually tackled using heuristics. For instance, the robot constructs a map of the environment using computer vision. As this map is uncertain, one may now attempt to decrease the robot's risk by tightening the area that can safely be traversed. This tightening is often based on past experimental data. However, this experimental data may not accurately represent real-world disaster scenarios. Additionally, an overly conservative tightening might lead the robot to waste time seeking alternate routes, while too little tightening could place the robot at risk of entrapment. Consequently, there is a need for a systematic approach to assessing the risks and rewards associated with different actions amid uncertainty.

The need for a systematic approach to risk assessment has only increased in recent years due to the ubiquity of autonomous systems that alter our day-to-day experiences and their need for safety, e.g., for self-driving vehicles, mobile service robots, and bipedal robots. These systems are expected to function safely in unpredictable environments and interact seamlessly with humans, whose behavior is notably challenging to forecast. To reason about risk in such settings, the fields of systems science and control engineering have a long history and a rich literature on analyzing and designing systems under uncertainty. Existing methodologies can be broadly classified into the three paradigms of worst-case, risk-neutral, and risk-aware approaches as classified in [1]. In the *worst-case paradigm*, a system's ability to remain safe or perform satisfactorily is judged by examining its most severe safety violation or worst performance. This paradigm forms the basis of robust control [2–4] and robust safety analysis [5]. For instance, if a system is supposed to track a planned trajectory, the largest discrepancy between the system's realized trajectory and the planned trajectory can serve as a measure of the system's performance. Contrarily, in the *risk-neutral paradigm*, a system's capacity to stay safe or perform satisfactorily is evaluated on average or probabilistically.[1] This paradigm is often used in stochastic control and reinforcement learning [4, 6, 7] as well as in verification[8]. In the case of trajectory tracking, one would consider the average discrepancy between the system's realized trajectory and the planned trajectory to gauge the system's performance when using a risk-neutral approach. If the system aims to reach a designated target while avoiding an unsafe region, the probability of the system accomplishing these goals can be used to assess the system's performance [9]. However, the worst-case paradigm may result in an overly conservative risk assessment and impractical solutions, while the risk-neutral paradigm can not account for harmful but less likely events. We also note that the developments in the worst-case and the risk-neutral paradigms were often mathematically motivated but not practically driven as one was able to derive analytical solutions to the posed problems. The *risk-aware paradigm*, on the other hand, lies conceptually in between and aims to evaluate a system's performance by giving attention to system outcomes that do not correspond to the worst case or the average. Historically, the risk-aware paradigm has focused on the application of mean-variance approximations, e.g., for controller design [10–12], which is derived from the Markowitz model for evaluating the risk of financial portfolios [13].

A contemporary risk-aware approach that is adopted in this survey to mitigate rare and detrimental outcomes employs the use of *tail risk measures*, a concept borrowed from financial literature [14]. As such, this survey will introduce these measures and explain their relevance in the context of robotic systems. To preface this explanation, however, we will first provide a deeper overview of the aforementioned paradigms, including an emphasis on the limitations of the *worst-case* and *risk-neutral* paradigms, which prompted the rise of *risk-aware* study.

Limitations of the risk-neutral and worst-case paradigms Many systems are affected by uncertainties that can be well-modeled by random noise. For example, uneven terrain can disturb a robot's planned trajectory, a wind gust can destabilize an aerial vehicle, and smoke can interfere with perception during an autonomous rescue mission. However, designing operating rules to optimize a system's performance on average need not yield trustworthy performance in practice. For instance, consider an agricultural robot where the source of uncertainty is tomorrow's precipitation, a difficult quantity to predict. An irrigation protocol that regulates soil moisture on average may be indifferent to variable weather patterns or spatial variability of farmland, which may lead to reduced crop yields or wasted water. In another scenario, imagine a robot autonomously navigating a disaster-stricken area in search of survivors. The robot must be able to assess risks in debris-laden zones while taking into account uncertainties derived from sensor and state estimation inaccuracies. If the robot operates under a worst-case scenario strategy, it may not find a feasible path through the rubble due to its aversion to any risk. On the other hand, a risk-neutral path-planning approach might expose the robot to unsafe, mission-jeopardizing situations. These examples are among the many that underline the limitations of the risk-neutral and worst-case paradigms in the context of robotic applications. Case studies from our previous works will be presented later on to further illustrate these points.

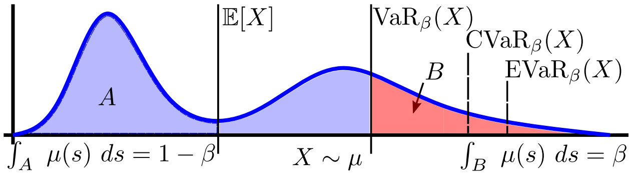

FIGURE 1: Visualization of common tail risk measures. The tail-risk measures referenced in the article are shown above, applied to the random variable X with distribution function μ. Figure adapted from [15].

The risk-aware paradigm In comparison to the risk-neutral and worst-case paradigms, the risk-aware approach has received relatively little attention within the robotics and autonomous systems fields. However, the need for this approach became already evident in previously mentioned examples of autonomous farming and search-and-rescue operations. For instance, in autonomous farming, a risk-aware irrigation strategy can account for spatial variability in soil moisture and potential future variations in temperature and precipitation. This would create a balance between maximizing crop production and conserving water. In the context of autonomous search-and-rescue, noisy state estimations and sensor data could be filtered through a tail-risk perspective to generate paths that are sufficiently distanced from obstacles yet not overly conservative, allowing for the completion of the rescue mission within a predetermined time frame. Traditional risk assessments have been based on a mean-variance

[1] Note that the probability of an undesirable event can be expressed as the expected value over an indicator function, making probabilistic reasoning conceptually similar to average reasoning.

trade-off or the probability of a harmful event, such as machine failure or constraint violation. However, variance measures the variability of a random variable with respect to the average in any direction, thus limiting a system's ability to predict potential harm. To illustrate, in the search-and-rescue scenario, using variance to evaluate the robot's mission completion time wouldn't be fitting. Exceeding the average mission completion time poses a risk of deadline violation and should be avoided, while faster mission completion does not present a problem. On the other hand, the probability of constraint violation provides a rather rudimentary concept of risk, neglecting details about the violation like its magnitude or severity. For instance, rather than simply estimating the probability of the rescue robot getting stuck, it would be more useful to estimate the "clearance" along the robot's path before navigating through it. Moreover, it might be beneficial for the robot to initially adopt a conservative approach to avoid mission-ending scenarios and gradually become more risk-neutral towards the end of the search in order to maximize the number of successful rescues. These considerations underscore the need for risk-aware planning and control methodologies that offer intuitive and systematic ways of dynamically altering the robot's risk perception. This has sparked recent research in robotics that employs tail risk measures, which will be further discussed.

Tail Risk Measures Succinctly, risk measures are functions over scalar-valued random variables designed to identify characteristics of interest of the random variable in question [16]. By *tail risk measures*, we imply that the risk measures of interest assess the upper right tail of the distribution of the random variable. Typically, the random variable indicates a cost so that tail risk measures capture the risk of incurring a high cost. Figure 1 depicts a few examples of such tail risk measures, namely, Value-at-Risk, Conditional-Value-at-Risk, and Entropic-Value-at-Risk. The Value-at-Risk at level $\beta \in (0, 1)$ corresponds to the cutoff value for which a fraction β of the outcomes of the random variable lies to the right of this cutoff value. A more formal description of these risk measures will follow in the Section "Tail Risk Measures: Definitions and Notation." We focus on these measures though as they provide a systematic way of assessing the rare and unsafe (costly) outcomes that must be limited during planning, control, and verification. For example, consider a robot traversing through an environment with uncertain information about the obstacles' positions due to sensor noise. In this case, we could define a cost random variable by negating the minimum distance to all obstacles. As such, more positive, *c.f.* more costly outcomes, would correspond to more unsafe behavior as the system is not maintaining the required distance to the measured obstacle. Then, as we have uncertain knowledge of the obstacle's location, we aim to minimize the tail risk incurred by this random cost in an effort to realize safe, risk-aware control actions. This intuitive approach to risk-aware decision selection can be applied to several facets of an autonomy stack, *i.e.* planning, control, and verification, as has recently been done in both the controls and robotics communities.

Organization As such, this survey serves as an introduction to risk-aware planning, control, and verification in robotics, employing tail risk measures - an emerging field in the literature. Our focus areas include:

» A summary of risk measure theory with an emphasis on tail risk measures in the section "Tail Risk Measures: Definitions and Notation". We highlight how these measures offer a consistent and intuitive means to adjust the system's risk aversion levels.

» A discussion on the key principles of risk-aware planning and control, an introduction of the algorithms, and multiple presentations of real-world case studies such as planning and control in subterranean environments. These are covered in the sections "Risk-Aware Planning" and "Risk-Aware Control".

» A review of temporal logics as a mathematical formalism for articulating complex robotic system specifications in the "Verification" section. Additionally, we explore why it is important to consider the tail risk of system trajectories evaluated against these specifications.

» An introduction of a tail-risk method for safety-critical controller verification in the "Verification" section. We demonstrate this risk-aware verification approach's ability to effectively identify potential mission issues while validating the probabilistic verification statements made within the described framework.

» We end the survey with open problems and future research directions.

Related Work Despite the crucial need for systematic risk evaluation in robotics applications, recent surveys do not emphasize risk-aware planning, control, and verification [17–19]. For instance, Schwarting, Alonso-Mora, and Rus examine planning and decision-making methods for autonomous driving [17], Karpas and Magazzeni discuss strategies that enable robots to automatically combine smaller tasks to achieve broader goals [18], and Brunke et al. outline the interplay between control theory and reinforcement learning for safety in robotic applications [19]. However, these works do not focus on risk measures. Moreover, Hobbs et al. provide a comprehensive introduction to non-stochastic run-time assurance systems, such as a system that overrides an existing controller when an extreme hazard is detected, without an emphasis on risk measures [20]. Two closely related works to our survey are Majumdar and Pavone's [21] and Wang and Chapman's [1]. Majumdar and Pavone propose axioms for risk measures suitable for robots and provide intuitive explanations for these axioms. However, their work does not take the form of a survey and present algorithms for risk-aware planning, control, and verification [21]. On the other hand, Wang and Chapman overview historical and modern research about risk-aware autonomous systems, but they do not focus on robotics applications, temporal logics, or certain tail risk measures such as entropic value-at-risk and total variation distance-based risk measure [1]. Our survey draws inspiration from Majumdar and Pavone [21] and Wang and Chapman [1] and aims to elucidate the concept of tail risk measures for the control systems community and to showcase their utility for planning, control, and verification of robotic systems. We provide the much-needed emphasis on these risk measures in robotics, helping to ensure that the development and application of autonomous systems remain safe, effective, and mindful of potential risks. As we move on in this survey, we will provide more pointers to relevant literature.

TAIL RISK MEASURES: DEFINITIONS AND NOTATION

To begin, we will define *tail-risk measures* and some corresponding notation that will be used throughout the article. Consider a probability space $(\Omega, \mathcal{F}, P)$, where Ω, $\mathcal{F}$, and P are the sample space, the σ-algebra over Ω, and the probability measure over $\mathcal{F}$, respectively. A random variable $X : \Omega \to \mathbb{R}$ denotes the cost of each outcome, and $\mathcal{X}$ is the set of all such random variables defined on Ω. For any random variable $X \in \mathcal{X}$, $F_X(x)$ refers to the cumulative distribution function with inverse $F_X^{-1}(\beta) = \inf\{x \in \mathbb{R} | F_X(x) \geq 1 - \beta\}$. For any two random variables $X, X' \in \mathcal{X}$, the expression $X \overset{d}{=} X'$ denotes that the random variables X, X' have the same distribution under P. Similarly, we use $X \leq X'$ as a shorthand notation to indicate that $X(\omega) \leq X'(\omega)$ for all $\omega \in \Omega$. Finally, U denotes the uniform random variable between $[0, 1]$.

A risk measure ρ is a function that maps a cost random variable to a real number, *i.e.* $\rho : \mathcal{X} \to \mathbb{R}$. Informally, tail risk measures refer to the behavior of the cost X in the tail of its distribution. Mathematically, consider the random variable $X \in \mathcal{X}$ and the risk-level $\beta \in (0, 1)$. As seen in [78], we define X_β to be the tail risk of X such that,

$$X_\beta = F_X^{-1}(1 - \beta + \beta \mathrm{U}). \tag{1}$$

In other words, the distribution of the random variable X_β is the distribution of X in its β-quantile (or tail) normalized to sum to 1. It is also clear that as $\beta \to 0$, $X_\beta \to \operatorname{ess\,sup}(X)$. We are now ready to formally define tail risk measures.

Definition 1 (Tail Risk Measures [78]). *For $\beta \in (0, 1)$, a risk measure ρ is a β-tail risk measure (or simply tail risk measure) if $\rho(X) = \rho(X')$ for all $X, X' \in \mathcal{X}$ satisfying $X_\beta \overset{d}{=} X'_\beta$.*

The aforementioned definition is a formal description of tail-risk measures, three of which have featured more prominently in recent literature — Value-at-Risk, Conditional-Value-at-Risk, and Entropic-Value-at-Risk. Their definitions will follow.

Value-at-Risk

Chance constraints can be reformulated by a commonly used risk measure called the *Value-at-Risk* (VaR). For a given confidence level $\beta \in (0, 1)$, VaR_β denotes the β-quantile value of the cost variable X and is defined as,

$$\mathrm{VaR}_\beta(X) := \inf\{z \in \mathbb{R} \,|\, F_X(z) \geq 1 - \beta\}.$$

Therefore, $\mathrm{VaR}_\beta(X) = F_X^{-1}(\beta)$. It follows that $\mathrm{VaR}_\beta(X) \leq 0 \implies P(X \leq 0) \geq 1 - \beta$.

Conditional Value-at-Risk

The *Conditional Value-at-Risk*, CVaR_β, measures the expected loss in the β-tail of the random variable X. Formally, for some $\beta \in (0, 1]$ CVaR_β is defined as follows per [14]:

$$\mathrm{CVaR}_\beta(X) := \inf_{z \in \mathbb{R}} \mathbb{E}\left[z + \frac{(X - z)^+}{\beta}\right] \tag{2}$$

where we use the notation $(X - z)^+ = \max(0, X - z)$. A value of $\beta \simeq 1$ corresponds to a risk-neutral case. A value of $\beta \to 0$ is rather a risk-averse case[2]. Importantly, CVaR is a coherent risk measure [79] (see the text below), prompting its widespread use in the recent literature. Furthermore, CVaR is a loose upper bound on VaR, *i.e.*,

$$\mathrm{VaR}_\beta(X) \leq \mathrm{CVaR}_\beta(X) \leq 0 \implies P(X \leq 0) \geq 1 - \beta. \tag{3}$$

Entropic Value-at-Risk

The *Entropic Value-at-Risk*, EVaR_β, derived using the Chernoff inequality for the random variable in question, is the tightest upper bound for VaR and CVaR. It was shown in [80] that EVaR_β and CVaR_β are equal only if there are no losses ($X \to -\infty$) below the VaR_β threshold. For some $\beta \in (0, 1]$,

$$\mathrm{EVaR}_\beta(X) := \inf_{z > 0}\left[z^{-1} \ln \frac{\mathbb{E}[e^{Xz}]}{\beta}\right]. \tag{4}$$

Similar to CVaR_β, for EVaR_β, the limit $\beta \to 1$ corresponds to a risk-neutral case; whereas, $\beta \to 0$ corresponds to a risk-averse case. In fact, it was demonstrated in [81, Proposition 3.2] that $\lim_{\beta \to 0} \mathrm{EVaR}_\beta(X) = \operatorname{ess\,sup}(X)$. Finally, EVaR is also a coherent risk measure and is an upper bound for both VaR and CVaR:

$$\mathrm{VaR}_\beta(X) \leq \mathrm{CVaR}_\beta(X) \leq \mathrm{EVaR}_\beta(X) \leq 0, \\ \implies P(X \leq 0) \geq 1 - \beta.$$

[2] Another, more intuitive, way to think the widely used *CVaR* metric is that it is the expectation of the random variable X conditioned on $X \geq \mathit{VaR}_\beta(X)$, i.e., $\mathit{CVaR}_\beta(X) = \mathbb{E}[X | X \geq \mathit{VaR}_\beta(X)]$. For example, the 5% CVaR risk of a portfolio is equivalent to the expected (mean) return on a portfolio in the worst 5% of scenarios over a specified time horizon (in this definition we assume that X is larger for worse returns).

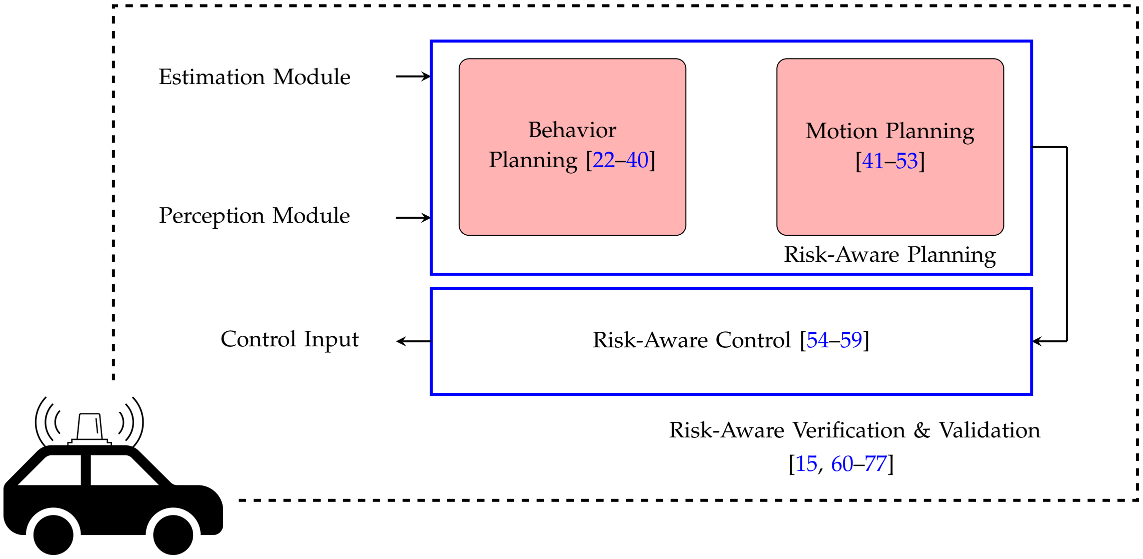
Estimation Module
Perception Module
Control Input
Behavior Planning [22–40]
Motion Planning [41–53]
Risk-Aware Planning
Risk-Aware Control [54–59]
Risk-Aware Verification & Validation [15, 60–77]

FIGURE 2: An overview of a typical (risk-aware) planning and verification pipeline in an autonomy stack.

Total Variation Distance-Based Risk Measure

Oftentimes the exact distribution of the random variable X to be analyzed is unknown. However, a family of distributions that includes the distribution of X may be known. Termed the ambiguity set, if we denote the probability measure for X to be P, we define this set $\mathcal{Q}_{\text{TVD}}$ as follows:

$$\mathcal{Q}_{\text{TVD}} := \Big\{ Q : \mathcal{F} \to [0,1] \mid \\ d_{TV}(P,Q) := \frac{1}{2}\|P-Q\|_1 \leq 1-\beta \Big\}.$$

We can choose actions by analyzing their cost over the distributions in this set. Formally, this results in a *total variation distance* (TVD) risk-measure termed the *total variation distance* (TVD) [82], defined as follows:

$$\text{TVD}_\beta(X) := \sup_{Q \in \mathcal{Q}_{\text{TVD}}} \mathbb{E}_Q(X) = (1-\beta) \sup_{x \in \Omega} x + \beta \, \text{CVaR}_\beta(X),$$

where $\mathbb{E}_Q(\cdot)$ is the expected value w.r.t. the measure Q. It follows from the definition of TVD,

$$\text{VaR}_\beta(X) \leq \text{CVaR}_\beta(X) \leq \text{TVD}_\beta(X) \leq 0, \\ \implies P(X \leq 0) \geq 1-\beta.$$

Coherent Risk Measures

Coherent risk measures are a prominent class of tail risk measures and are well-regarded for their robust mathematical properties and their inherent intuitiveness for risk analysis. Introduced by Artzner et al. [79] in the context of financial risk management, coherent risk measures satisfy four axiomatic properties: monotonicity, subadditivity, positive homogeneity, and translational invariance. Monotonicity indicates that adding a less risky outcome to a portfolio should not increase its risk. Subadditivity implies that diversifying a risk portfolio should not increase its overall risk. Positive homogeneity signifies that scaling all outcomes in a portfolio should proportionally scale its risk. Translational invariance denotes that adding a risk-free asset to a portfolio should decrease its risk correspondingly. Coherent risk measures offer a rich theoretical foundation to quantify and manage risk systematically. They enable us to capture extreme, but rare, high-consequence events and provide a mechanism to evaluate and compare different risk scenarios, making them particularly valuable for risk-aware planning, control, and verification in robotics. We are now ready to describe coherent risk measures.

Definition 2 (Coherent Risk Measure)**.** *We call the risk measure $\rho : \mathcal{X} \to \mathbb{R}$, a* coherent risk measure, *if it satisfies the following conditions*

- » ***Subadditivity:*** *$\rho(X + X') \leq \rho(X) + \rho(X')$, for all $X, X' \in \mathcal{X}$;*
- » ***Monotonicity:*** *If $X \leq X'$ then $\rho(X) \leq \rho(X')$ for all $X, X' \in \mathcal{X}$;*
- » ***Translational Invariance:*** *$\rho(X + c) = \rho(X) + c$ for all $X \in \mathcal{X}$ and $c \in \mathbb{R}$;*
- » ***Positive Homogeneity:*** *$\rho(\beta X) = \beta\rho(X)$ for all $X \in \mathcal{X}$ and $\beta \geq 0$.*

Note that the properties of subadditivity and positive homogeneity together imply that coherent risk measures are also convex. CVaR, EVaR, and TVD have been recognized as coherent risk measures. Conversely, VaR does not qualify as a coherent risk measure, as it does not satisfy the sub-additivity property. This fact limits its utility in situations where a joint assessment of risks is required, which is often the case in risk-aware robotic planning, control, and verification. While VaR is not a coherent risk measure, each of the measures defined above has its place in risk evaluation, depending on the specific requirements and constraints of the task at hand.

Coherent risk measures provide a snapshot of potential perils based on current conditions. These measures fall short when applied to dynamic systems, such as those common in robotics, where risks and conditions vary with time. *dynamic coherent risk measures* extend beyond the static approach, taking into account the time-varying nature of risk. They are particularly adept at characterizing the evolving risk landscape in dynamic environments. Dynamic risk measures continuously monitor and update risk assessments in response to changes in the system and its environment. This capability to adapt and provide a comprehensive understanding of risk in a fluctuating context aligns well with the realities of robotic operations in unstructured environments. To define these measures, we must first extend the traditional probability spaces mentioned previously.

Consider a probability space $(\Omega, \mathcal{F}, P)$, a filtration $\mathcal{F}_0 \subset \cdots \mathcal{F}_N \subset \mathcal{F}$, and an adapted sequence of random variables $X_t,\ t = 0, \ldots, N$, where $N \in \mathbb{N}_{\geq 0} \cup \{\infty\}$. For $t = 0, \ldots, N$, we further define the spaces $\mathcal{X}_t = \mathcal{L}_p(\Omega, \mathcal{F}_t, P)$, $p \in [1, \infty)$. Here $\mathcal{L}_p$ is the set of all p-bounded random variables, *i.e.* $\mathcal{L}_p(\Omega, \mathcal{F}_t, P) = \{X : \Omega \to \mathbb{R} \mid \mathbb{E}_P[|X|^p] < \infty\}$. Furthermore, let $\mathcal{X}_{t:N} = \mathcal{X}_t \times \cdots \times \mathcal{X}_N$ and $\mathcal{X} = \mathcal{X}_0 \times \mathcal{X}_1 \times \cdots$. We assume that the sequence $\boldsymbol{X} \in \mathcal{X}$ is almost surely bounded (with exceptions having probability zero), *i.e.*, $\max_t \operatorname{ess\,sup}\ |X_t(\omega)| < \infty$.

To describe how one can evaluate the risk of subsequence $X_t, \ldots, X_N$ from the perspective of stage t, we require the following definitions.

Definition 3 (Conditional Risk Measure)**.** *A mapping $\rho_{t:N} : \mathcal{X}_{t:N} \to \mathcal{X}_t$, where $0 \leq t \leq N$, is called a* conditional risk measure, *if it has the following monoticity property:*

$$\rho_{t:N}(\boldsymbol{X}) \leq \rho_{t:N}(\boldsymbol{X}'), \quad \forall \boldsymbol{X}, \forall \boldsymbol{X}' \in \mathcal{X}_{t:N} \text{ such that } \boldsymbol{X} \preceq \boldsymbol{X}'.$$

Definition 4 (Dynamic Risk Measure)**.** *A* dynamic risk measure *is a sequence of conditional risk measures $\rho_{t:N} : \mathcal{X}_{t:N} \to \mathcal{X}_t$, $t = 0, \ldots, N$.*

A key attribute of dynamic risk measures is their temporal consistency [83, Definition 3]: if two scenarios X and X' are identical for a time interval $[\tau, \theta]$, and if X is evaluated to be as favorable as X' at some future time point θ, then it stands to reason that X should not be viewed as more risky than X' at the earlier time point τ. This principle ensures that the risk assessment remains coherent and consistent across different points in time. The definition of a dynamic coherent risk measure [84, p. 298] then follows from its static counterpart.

Definition 5 (Dynamic Coherent Risk Measure)**.** *We call the one-step conditional risk measures $\rho_t : \mathcal{X}_{t+1} \to \mathcal{X}_t$, $t = 1, \ldots, N-1$ a* coherent risk measure *if it satisfies the following conditions*

- » ***Convexity:*** *$\rho_t(X + (1-\lambda)X') \leq \lambda\rho_t(X) + (1-\lambda)\rho_t(X')$, for all $\lambda \in (0,1)$ and all $X, X' \in \mathcal{X}_{t+1}$;*
- » ***Monotonicity:*** *If $X \leq X'$ then $\rho_t(X) \leq \rho_t(X')$ for all $X, X' \in \mathcal{X}_{t+1}$;*
- » ***Translational Invariance:*** *$\rho_t(X + X') = X + \rho_t(X')$ for all $X \in \mathcal{X}_t$ and $X' \in \mathcal{X}_{t+1}$;*
- » ***Positive Homogeneity:*** *$\rho_t(\beta X) = \beta\rho_t(X)$ for all $X \in \mathcal{X}_{t+1}$ and $\beta \geq 0$.*

A BRIEF INTRODUCTION TO TAIL RISK MEASURES IN ROBOTICS

A risk measure ρ maps a random variable X to a real number that indicates the risk associated with X. The random variable X can be used in many robotic contexts where a dynamical system produces random trajectories. To preface the following exposition, consider a (perhaps) nonlinear discrete-time control system at time t with state $\boldsymbol{x}(t) \in \mathcal{X} \subseteq \mathbb{R}^n$, input $\boldsymbol{u}(t) \in \mathcal{U}$, and system disturbances $\boldsymbol{d}(t) \sim \xi(\boldsymbol{x}, \boldsymbol{u}, t)$ where $\xi(\boldsymbol{x}, \boldsymbol{u}, t)$ is a (perhaps) state, input, and time-dependent probability distribution over $\mathbb{R}^n$:

$$\boldsymbol{x}(t+1) = f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t)). \tag{5}$$

Given a feedback controller $U : \mathcal{X} \to \mathcal{U}$, we can construct a closed-loop, stochastically evolving dynamical system. From a specific initial condition $\boldsymbol{x}_0 \in \mathcal{X}$, we denote via $\Sigma(\boldsymbol{x}_0)$ the set of random trajectories $\boldsymbol{x}$ realized by this closed-loop system, *i.e.*

$$\text{a sample of } \Sigma(\boldsymbol{x}_0) \text{ is } \boldsymbol{x} \triangleq \{\boldsymbol{x}_0 \equiv \boldsymbol{x}(0), \boldsymbol{x}(1), \ldots\}, \tag{6a}$$

where time update between $\boldsymbol{x}(t+1), \boldsymbol{x}(t)$ is provided in (5). To analyze the tail risk of these trajectories, we require a mapping from trajectory samples $\boldsymbol{x}$ to the real line. Denoting the space of possible trajectories as $\mathcal{S}^{\mathcal{X}}$, define a cost function

$$C : \mathcal{S}^{\mathcal{X}} \to \mathbb{R}.$$

The function C may denote the inverse distance of a robot's trajectory to an obstacle and thereby encode the system's robustness to a collision. These cost functions can be constructed from principled approaches when the system's desired behavior is expressed as a temporal logic specification– see Sidebars 3 and 4. The evaluation of cost C over random system trajectories, *i.e.*, $X = C(\Sigma(x_0))$, is likewise a real-valued random variable. For robotic planning, control, and verification, one can then minimize a risk cost measure, $\rho(C(X))$, or consider using risk as constraint–$\rho(C(X)) \leq r$ for a risk threshold r.

RISK-AWARE PLANNING & CONTROL

The robotic behavior, motion planning, and control problems focus on designing algorithms that allow a robot to interact intelligently and safely with its surroundings. This complex process involves deciding on possible actions, constructing trajectories that a robot can take to achieve specific high-level goals, e.g., safely navigating through unstructured environments, and controlling the instantaneous robot motions to track the trajectory. Behavior planning concerns the higher-level decision-making needed to achieve higher-level robot objectives, while motion planning plans the details of robot movement, determining how a robot should move from one location to another while avoiding obstacles, and factoring in the robot's kinematics and dynamics. The control level manages the details of motion execution by continually computing the system control inputs that minimize tracking error while accounting for uncertainty and unexpected events.

The process of designing robot plans and controls should critically consider potential risks and their overall system implications. These risks include physical risks to humans close to a robot as well as risks to the robot itself due to its unpredictable environment and imperfect robot sensing and perception systems. Notions of risk are particularly important in high-stakes robot tasks, such as search and rescue, or exploration of hazardous sites. To properly manage these risks, this paper advocates for the use of tail risk measures. Importantly, tail risk measures focus on more extreme but rarer events, and thus provide a systematic and principled approach to quantifying and managing risk in robotics. Integrating tail risk measures into robotic planning and control modules can lead to more robust and safer robot operation, effectively balancing performance and safety under uncertainty.

Historical Remark on Exponential Utility: Risk-aware control has been in development for at least fifty years. The earliest contributions concern the exponential utility measure (i.e., entropic risk measure):

$$\rho_{\mathrm{EU},\theta}(X) := \tfrac{-2}{\theta} \log \mathbb{E}\left[\exp(\tfrac{-\theta}{2}X)\right], \tag{7}$$

where $\theta \neq 0$ is a risk parameter and X is a nonnegative random variable, which can be interpreted as a mean-variance approximation [85]. When $\theta < 0$, the robotic system is risk averse, while the robot will exhibit more risk tolerant behavior when $\theta > 0$. When $\theta = 0$, the system is risk neutral. Notably, the exponential utility is *not* a tail-risk measure. To our knowledge, the first paper in the area of risk-aware control was a 1972 study about finite-state Markov decision processes by Howard and Matheson, in which performance was assessed by an exponential utility criterion [10]. The authors took inspiration from game theory [86]. One year later, Jacobson investigated the exponential utility criterion in the classical linear-quadratic setting (linear dynamics, quadratic costs, Euclidean spaces, and additive Gaussian noise) [11]. Whittle developed numerous contributions regarding risk-aware control in the linear-quadratic setting using the exponential utility measure, including the case of partially observable systems [85, 87]. While exponential utility continues to be investigated (e.g., see [88–90]), recent attention focuses on different types of risk-aware performance and safety criteria, motivating this survey on tail risk.

Risk-Aware Behavior Planning

While behavior planning techniques rely on various mathematical frameworks, this section will focus on the popular Markov Decision Processes (MDPs), which are widely utilized for planning under uncertainty, such as in reinforcement learning. An MDP is a quadruple,

$$\mathcal{M} = (\mathcal{S}, Act, T, \kappa_0)$$

where $\mathcal{S} = \{s_1, \ldots, s_{|\mathcal{S}|}\}$ represents the states of the autonomous agent(s) and the world model, a set of actions $Act = \{\alpha_1, \ldots, \alpha_{|Act|}\}$ available to the agent, a transition function $T(s_j|s_i, \alpha)$ defining the likelihood of transitioning to state s_j from state s_i when taking action $\alpha \in Act$, and an initial distribution κ_0 over the states $s \in \mathcal{S}$. The transition function and initial state distribution must satisfy the following criteria:

$$\begin{cases} \sum_{s\in\mathcal{S}} T(s|s_i, \alpha) = 1, & \forall\ s_i \in \mathcal{S}, \forall\ \alpha \in Act, \\ \sum_{s\in\mathcal{S}} \kappa_0(s) = 1. \end{cases} \tag{8}$$

A *finite* Markov Decision Process has finite state and action spaces. Finally, a policy π maps states to actions, *i.e.* $\pi : \mathcal{S} \rightarrow Act$.

Then broadly speaking, risk-aware behavior planning can be approached in one of two ways. The first method aims to identify an optimal risk-aware policy. Specifically, consider the robot as a stochastic system, where the state at time instance t is represented by s_t and the action taken is denoted by a_t. Policies π are defined as a sequence of actions based on the state history, *i.e.* $\pi : (s_0, a_0, \ldots, s_{t-1}, a_{t-1}, s_t) \mapsto a_t$. Given that the risk associated with each state-action pair (s_t, α_t) is defined as a random variable $c(s_t, \alpha_t)$, the nominal goal of risk-aware policy development is to find a policy that optimizes for this risk measure. In the context of coherent risk measures, the optimal policy, π^*, minimizes the tail risk measure ρ as follows:

$$\pi^* = \arg\min_{\pi} \rho(c(s_t, \alpha_t)), \tag{9}$$

where $\alpha_t = \pi(s_0, \alpha_0, \ldots, s_{t-1}, \alpha_{t-1}, s_t)$, which is not necessarily a Markov (memoryless) policy. In a robotic system, s_t could denote the robot's state (position, velocity, etc.), α_t could be the applied control commands, and cost function $c(s_t, \alpha_t)$ could quantify an undesirable outcome such as risk of collision or deviation from a target path. The second method defines the optimal policy as that which selects the optimal action α_t^* at every state s_t. This optimal action is defined as the one that minimizes the tail risk measure ρ applied to the cost random variable at that state s_t:

$$\alpha_t^* = \arg\min_{\alpha_t} \rho(c(s_t, \alpha_t)) \tag{10}$$

Both of these formulations aim to minimize risk, not only considering average-case scenarios but also potential rare, adverse events captured by the risk measure ρ.

The Discounted MDP Problem

The risk-aware policy identification problem, Equation (9), considers sequences of actions, rather than a single action. For this purpose, define a finite sequence of risk measures $\{\rho_t\}_{t=0}^T$ and a similar sequence of cost functions $\{c_t\}_{t=0}^T$.

The concept of a "discounted MDP" emerges here, which seeks to find a policy that minimizes the total risk over sequences of actions, taking into account a discount factor. The discount factor, $\gamma \in [0,1]$, decreases the influence of future costs and risks in the decision-making process, implying that immediate risks and costs are given more weight than future ones. We define the finite-time discounted cost

$$J_\gamma(\kappa_0, \pi) = \rho_{0,T}(c_0, \gamma c_1, \ldots, \gamma^T c_T), \tag{11}$$

where

$$\rho_{0,t}(c_0, \gamma c_1, \ldots, \gamma^t c_t) = \rho_0 \Big(c_0 + \rho_1(\gamma c_1 + \rho_2(\gamma^2 c_2 + \cdots \\ + \rho_{t-1}\left(\gamma^{t-1} c_{t-1} + \rho_t(\gamma^t c_t)\right) \cdots)) \Big).$$

Then the optimal policy selection problem is to identify

$$\begin{aligned} \pi^* \in \ & \underset{\pi}{\text{argmin}} && J_\gamma(\kappa_0, \pi) \\ & \text{subject to} && \boldsymbol{D}_\gamma(\kappa_0, \pi) \preceq \boldsymbol{\beta} \end{aligned} \tag{12}$$

where $\boldsymbol{D}_\gamma(k_0, \pi)$ denotes a vector of discounted cost functionals, such as J in (11) which include both risk measures and cost functions. For the infinite-horizon case, the cost functional definition changes slightly as follows:

$$J_\gamma(\kappa_0, \pi) = \lim_{T\to\infty} \rho_{0,T}(c_0, \gamma c_1, \ldots, \gamma^T c_T). \tag{13}$$

Similarly, the vector of constraint functionals $\boldsymbol{D}_\gamma(k_0, \pi)$ can be finite or infinite horizon as well. For the infinite horizon case, if the cost functions c and component discounted cost functionals $d^i \ \forall \ i = 1, 2, \ldots, n_c$ (the components of $\boldsymbol{D}$) are non-negative and upper-bounded, and the discount factor $\gamma \in (0,1)$, then for an initial condition κ_0, and a policy π, we infer from [83, Theorem 3] that both $J_\gamma(\kappa_0, \pi)$ and $d_\gamma^i(\kappa_0, \pi)$ are well-defined (bounded). For the finite horizon case, a solution always exists.

Prior Work on Discounted MDPs: In an early work in this vein, the authors of [83] presented techniques for incorporating this measure into dynamic programming. This work resulted in a wave of new work evaluating risk measures in dynamic programming problems [21–24, 91]. E.g., in [25, 26] the authors identified locally optimal solutions via gradient descent, to MDP problems with CVaR constraints and total expected costs. Notably, [25] provides a convergence guarantee whereas [26] does not. The authors of [27, 28] extended these prior notions by developing sample-based saddle point algorithms to identify policies for MDPs whose cost is a coherent risk measure, though not specifically CVaR. Other relevant works include [29–32].

One question posed by the authors in [33] has caused a renewal of work in this vein. Specifically, the authors show that most risk-level dynamic programs cannot guarantee the recovery of a globally optimal value function despite discretized state space. To partially address that concern, in [34, 40] the authors generate optimal risk-aware policies for MDPs with dynamic coherent risk objectives and constraints. By phrasing policy generation as a difference convex program, solutions can also be rapidly identified. Despite these advances, the field of risk-aware discounted MDPs still holds numerous avenues for future exploration. New algorithms and techniques that can handle an expansive range of coherent risk measures and can effectively manage constraints in MDPs are needed.

The Un-discounted MDP Problem

When the discount factor γ is set to one, the discounted MDP problem transforms into an undiscounted (constrained) Markov Decision Process (MDP) problem, also referred to as a total cost/reward problem. Such problems can span both finite and infinite time horizons. However, it is worth noting that discovering solutions for risk-aware total cost MDPs is often more elusive compared to their discounted equivalents. This complexity primarily arises because the consideration of total costs requires a comprehensive understanding of the entire trajectory of states, as opposed to a myopic focus on immediate outcomes. To guarantee the existence of solutions, it often becomes necessary to consider a specific subclass of finite state MDPs, referred to as 'transient MDPs' in [35]. These MDPs encompass a distinctive cost-free goal state, or a 'termination' state $s^g \in \mathcal{S}$. The goal state transition probability $T(s^g \mid s^g, \alpha)$ is 1, and the cost function $c(s^g, \alpha)$ is 0, for all actions $\alpha \in Act$ (see Figure 3). The problem can readily be extended to multiple goal states: $\mathcal{S}_g \subset \mathcal{S}$.

Several significant methodologies for solving such MDPs have been proposed. For instance, the worst-case CVaR Stochastic Shortest Path (SSP) planning approach [36] uses dynamic programming to find a solution. The authors of [35] proposed to solve a total cost undiscounted MDPs featuring static CVaR measures via the use of a surrogate MDP, They demonstrated that the resolution of this surrogate approximates the optimal policy with an accuracy that can be arbitrarily close to the true solution. These groundbreaking strides lay a robust framework for future explorations and refinement of techniques that can effectively navigate the intricacies of transient MDPs and total cost/reward problems. Recently, a novel approach was proposed in [37], which applies to all coherent risk measures, a significant expansion of the problem's scope. The authors demonstrate the existence of optimal, stationary, Markovian policies and derive them through a specially formulated Bellman equation. Furthermore, they introduce an optimization-based computational technique, rooted in difference convex programs (DCPs), to determine the associated value functions and the risk-averse policies.

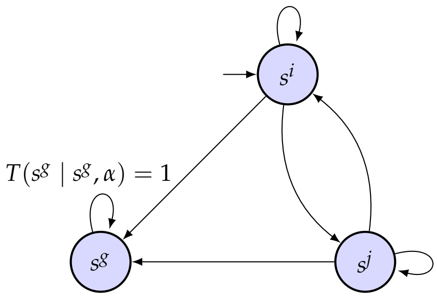

FIGURE 3: The transition graph of a transient MDP. The goal state s^g is cost-free and absorbing.

The POMDP Problem

Partially observable Markov Decision Processes (POMDPs) offer a valuable framework for studying decision-making under uncertainty, particularly when states of the agent or environment are not directly observable [92, 93]. While POMDPs can be difficult to design and solve, significant strides have been made in addressing coherent risk measure objectives. For instance, [38] explored POMDPs with coherent risk measure objectives. However, while their noteworthy theoretical contributions fell short of providing a computational method for designing policies applicable to general coherent risk measures. Ahmadi et al. [39] aimed to address this gap by proposing a method for finding finite-state controllers for POMDPs with objectives defined in terms of coherent risk measures. Their novel approach took advantage of convex optimization techniques, showcasing the potential of mathematical optimization in policy design. Nevertheless, their method has its limitations: it can only be applied when the risk transition mapping is affine in the policy.

Recognizing this limitation, Ahmadi et al.[40] extended their prior work [39] to incorporate a broader set of coherent risk measures. They proposed an innovative approach bounded policy iteration method that identifies finite-state risk-averse policies. This methodology breaks the problem down into manageable pieces, tackling convex optimization problems at each policy iteration step. This approach substantially ameliorates the computational tractability of synthesizing risk-averse policies for POMDPs. By iteratively solving these convex optimization problems, the policy synthesis process becomes markedly more feasible.

However, the methodology outlined in [40] has its limitations. One notable constraint is that the technique can currently only be applied to problems involving hundreds of states due to the computational limitations inherent to convex optimization. Despite existing limitations, the exploration of POMDPs in the context of coherent risk measures presents a promising field of study. As our understanding deepens and computational methods evolve, we can anticipate the development of more pragmatic solutions for planning under uncertainty under a broader range of coherent risk measures.

Risk-Aware Motion Planning and Control

The behavior planning layer of a hierarchical control system generates higher-level strategies for mission satisfaction. These strategies are then guided and tracked by the motion planning and control layers, accounting for the robot's kinematic constraints and control capabilities. Optimization-based motion planning methods are increasingly popular because they provide optimized system behaviors that respect the system's dynamics, while readily incorporating state and control constraints. Critically, one must account for sudden system changes or disturbances to ensure system safety. Risk-awareness methods can be integrated into optimal planning control problems.

Our risk-aware motion planning review considers discrete-time controlled stochastic systems, with the form:

$$\boldsymbol{x}(t+1) = f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t)). \tag{14}$$

Here, $\boldsymbol{x}(t) \in \mathbb{R}^{n_x}$ and $\boldsymbol{u}(t) \in \mathbb{R}^{n_u}$ are the system state and controls at time t, respectively. The system is affected by a stochastic process noise $\boldsymbol{d}(t) \in \mathbb{R}^{n_d}$ and $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_d} \to \mathbb{R}^{n_x}$. An optimization-based planner seeks to minimize a system cost $J(\boldsymbol{x}, \boldsymbol{u}) \in \mathbb{R}$ for initial condition $\boldsymbol{x}(0) = \boldsymbol{x}_0$ at time $t = 0$. The optimal controller $U = [\boldsymbol{u}(0), \dots, \boldsymbol{u}(N-1)]$ is the solution to the following optimization problem:

$$\begin{aligned} J^*(\boldsymbol{x}(0)) = \min_{U} \quad & \rho\Big(\sum_{t=0}^{N-1} J(\boldsymbol{x}(\boldsymbol{t}), \boldsymbol{u}(\boldsymbol{t}))\Big), \\ \text{s.t.} \quad & \boldsymbol{x}(t+1) = f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t)), \\ & \boldsymbol{x}(0) = \boldsymbol{x}_0, \quad \forall t \in \{0, \dots N-1\} \end{aligned}$$

Risk-Aware MPC

Model Predictive Control (MPC) applies the finite-horizon controller (18) in a receding-horizon fashion. Uncertainty can arise for many reasons. Uncertainty in the robot's dynamics model causes the true system motion to differ from the predicted one. Such effects are typically accounted for via process noise, $\boldsymbol{d}(t)$. Sensor noise and imprecise robot localization or estimation of environment states are other common sources of uncertainty. There are many ways to account for these uncertainties in an MPC framework. For example, Robust MPC accounts for worst-case disturbances in a set of bounded uncertainties [94]. Robust approaches are often too conservative because they focus on worst-case events. Conversely, stochastic MPC [95] only accounts for the average realization of the cost while respecting a bound on the probability of violating the state and control constraints, see Sidebar 1. The resulting policy, which is often too optimistic, minimizes the MPC objective in expectation instead of usefully accounting for events in the tail of the uncertainty distribution.

Risk-aware MPC optimizes risk-averse behavioral policies: they are not as conservative as in the robust case. But since they account for "risky" outcomes in the tail of the uncertainty distribution, they perform better in practice. In [41], the authors provide an MPC scheme for a discrete-time dynamical system with process noise whose objective was a Conditional Value-at-Risk (CVaR) measure. They further provided new Lyapunov conditions for risk-sensitive exponential stability. In [42], the authors devised an MPC scheme that expressed a distributionally-robust chance constraint along with a risk-aware cost in terms of a CVaR reformulation. Optimal control using distributionally robust CVaR constraints with second-order moment ambiguity sets is posed as a semidefinite program in [43]. A tree-based approach for MPC that enumerates all possible extreme disturbance signals and searches for feedback policies that account for a tradeoff between robustness and performance through CVaR metrics was proposed in [44]. In [45], the authors considered multistage risk-averse and risk-constrained optimal control for general coherent risk measures with conic representations.

Data-driven MPC that uses samples from the uncertainty distribution is becoming increasingly popular. Risk-aware MPC approaches provide the required robustness to account for the gap between enforcing the sample-based chance constraints for the empirical distribution and the true chance constraint for the actual uncertainty distribution. In [46, 47] the authors propose a distributionally robust data-enabled predictive control (DeePC) algorithm, that uses finite samples of an unknown system to make trajectory predictions. Instead of learning the system dynamics model, the authors propose an *equivalent* formulation using these data-driven trajectory predictions that enjoys strong out-of-sample guarantees using Wasserstein distributionally robust CVaR constraints. Reference [48] considers a learning MPC framework whose infinite horizon, CVaR-constrained, optimal control solution is approximated iteratively given a finite number of safe states and uncertainty samples. Through this iterative method, the authors construct a data-driven terminal set for distributionally robust CVaR-constrained iterative MPC with safety and feasibility guarantees.

MPC is also useful for obstacle avoidance in motion planning tasks. Risk-aware MPC accounts for varied obstacle behaviors and sensor and process noise not limited to Gaussian distributions. The MPC scheme in [49] avoids moving obstacles using a CVaR risk metric. Similar results were obtained in [50] on the Entropic Value-at-Risk (EVaR) metric for obstacle avoidance with additional guarantees of recursive feasibility and finite-time task completion while following a set of waypoints. In [51], these results were extended to general coherent risk measures for systems with process noise to obtain a disturbance feedback policy. The authors propose various constraint-tightening techniques to make the risk-aware obstacle avoidance MPC computationally tractable for motion planning. A risk-constrained MPC formulation was also studied in [52, 53] wherein the authors computed a risk map for traversing over rough terrain using CVaR. They incorporated this CVaR terrain map into MPC constraints to account for obstacles and terrain hazards.

Sidebar 1 (Model Predictive Control with Uncertainty)**.** *Consider a linear, discrete-time system given by*

$$\boldsymbol{x}(t+1) = A\boldsymbol{x}(t) + B\boldsymbol{u}(t) + D\boldsymbol{d}(t) \tag{16}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^{n_x}$ and $\boldsymbol{u}(t) \in \mathbb{R}^{n_u}$ are the system state and controls at time t, respectively. The system is affected by a stochastic, additive, process noise $\boldsymbol{d}(t) \in \mathbb{R}^{n_d}$.
Consider r_x state constraints of the form

$$\mathcal{X} := \{\boldsymbol{x} \in \mathbb{R}^{n_x} | F_x\boldsymbol{x} \leq g_x\}, F_x \in \mathbb{R}^{r_x \times n_x},\ g_x \in \mathbb{R}^{r_x}.$$

We also assume r_u control constraints having the form

$$\mathcal{U} := \{\boldsymbol{u} \in \mathbb{R}^{n_u} | F_u\boldsymbol{u} \leq g_u\}, F_u \in \mathbb{R}^{r_u \times n_x},\ g_u \in \mathbb{R}^{r_u}.$$

The goal is to steer the system to a set

$$\mathcal{X}_F := \{\boldsymbol{x} \in \mathbb{R}^{n_x} | F_f\boldsymbol{x} \leq g_f\}, F_f \in \mathbb{R}^{r_f \times n_x},\ g_x \in \mathbb{R}^{r_f},$$

while minimizing the control effort and deviation from the desired trajectory, i.e., we want to minimize the following cost:

$$J(\boldsymbol{x}, \boldsymbol{u}) := \boldsymbol{x}^T Q\boldsymbol{x} + \boldsymbol{u}^T R\boldsymbol{u},$$

where $Q \in \mathbb{R}^{n_x \times n_x}$ and $R \in \mathbb{R}^{n_u \times n_u}$ are weights on the state and control costs. Model Predictive Control (MPC) provides an optimization-based framework to compute the best N-step control input while satisfying the state and control constraints. The MPC optimization is given by,

$$\begin{aligned}
J_t^*(\boldsymbol{x}(t)) = \min_{U_t} \quad & \mathbb{E}\Big[\boldsymbol{x}_{t+N|t}^T P\boldsymbol{x}_{t+N|t} + && \text{(17a)}\\
& \sum_{k=t}^{t+N-1} \left(\boldsymbol{x}_{k|t}^T Q\boldsymbol{x}_{k|t} + \boldsymbol{u}_{k|t}^T R\boldsymbol{u}_{k|t}\right)\Big] && \text{(17b)}\\
\text{s.t.} \quad & \boldsymbol{x}_{k+1|t} = A\boldsymbol{x}_{k|t} + B\boldsymbol{u}_{k|t} + D\boldsymbol{d}_{k|t}, && \text{(17c)}\\
& Prob(\boldsymbol{x}_{k|t} \notin \mathcal{X}) \leq \beta,\ \boldsymbol{u}_{k|t} \in \mathcal{U}, && \text{(17d)}\\
& Prob(\boldsymbol{x}_{t+N|t} \notin \mathcal{X}_F) \leq \beta && \text{(17e)}\\
& \boldsymbol{x}_{t|t} = \boldsymbol{x}(t) \quad \forall k \in \{t, \dots t+N-1\}, && \text{(17f)}
\end{aligned}$$

where, $\boldsymbol{x}_{k|t}$ is the state at time k as predicted at the time t while starting from the current state $\boldsymbol{x}_{t|t} = \boldsymbol{x}(t)$ and β is the user chosen risk level. Uncertainty is propagated through the states as,

$$\boldsymbol{x}_{k+1|t} = A^k \boldsymbol{x}_{t|t} + \sum_{i=t}^{k} \left(A^{(k-i)} B \boldsymbol{u}_{i|t} + A^{(k-i)} D \boldsymbol{d}_{i|t} \right).$$

If the uncertainty is i.i.d Gaussian with $\boldsymbol{d}(t) \sim \mathcal{N}(0, \Sigma)$, the states $\boldsymbol{x}(t)$ are also Gaussian $\boldsymbol{x}_{k+1|t} \sim \mathcal{N}(\hat{\boldsymbol{x}}_{k|t}, \Sigma_{k|t})$ where, $\hat{\boldsymbol{x}}_{k|t} = A^k \boldsymbol{x}_{t|t} + \sum_{i=t}^{k} A^{(k-i)} B \boldsymbol{u}_{i|t}$ and $\Sigma_{k|t} = \sum_{i=t}^{k} D^T {A^{(k-i)}}^T \Sigma A^{(k-i)} D$ (the family of normal distributions is closed under linear transformations). Hence, we can rewrite the above uncertain MPC optimization as the following deterministic quadratic program,

$$\begin{aligned} J_t^*(\boldsymbol{x}(t)) = \min_{U_t} \quad & \mathbb{E}\Big[\boldsymbol{x}_{t+N|t}^T P \boldsymbol{x}_{t+N|t} + \\ & \sum_{k=t}^{t+N-1} \left(\boldsymbol{x}_{k|t}^T Q \boldsymbol{x}_{k|t} + \boldsymbol{u}_{k|t}^T R \boldsymbol{u}_{k|t} \right) \Big] \\ \text{s.t.} \quad & \boldsymbol{x}_{k+1|t} = A\boldsymbol{x}_{k|t} + B\boldsymbol{u}_{k|t} + D\boldsymbol{d}_{k|t}, \\ & F_x \hat{\boldsymbol{x}}_{k|t} + F_x \Phi^{-1}(1-\beta)\Sigma_{k|t} \leq g_x, \\ & F_u \boldsymbol{u}_{k|t} \leq g_u, \\ & F_f \hat{\boldsymbol{x}}_{t+N|t} + F_f \Phi^{-1}(1-\beta)\Sigma_{t+N|t} \leq g_f \\ & \boldsymbol{x}_{t|t} = \boldsymbol{x}(t) \, \forall k \in \{t, \ldots t+N-1\}. \end{aligned}$$

However, if the uncertainty distribution is non-Gaussian, the uncertain MPC (18) is not easily reformulated into a convex optimization program. In this case, must sample the uncertainty distribution and reformulate the MPC optimization as a much more computationally expensive mixed-integer program. Many tail risk measures such as CVaR, EVaR, TVD, etc, provide intuitive convex, inner approximations of chance constraints regardless of the uncertainty distribution. Hence, we propose risk-aware MPC formulations not only better account for uncertainty but also provide an efficient convex reformulation without making assumptions about the nature of the uncertainty. The resulting deterministic, risk-aware MPC formulation is given by,

$$\begin{aligned} J_t^*(\boldsymbol{x}(t)) = \min_{U_t} \quad & \rho\Big[\boldsymbol{x}_{t+N|t}^T P \boldsymbol{x}_{t+N|t} + \\ & \sum_{k=t}^{t+N-1} \left(\boldsymbol{x}_{k|t}^T Q \boldsymbol{x}_{k|t} + \boldsymbol{u}_{k|t}^T R \boldsymbol{u}_{k|t} \right) \Big] \\ \text{s.t.} \quad & \boldsymbol{x}_{k+1|t} = A\boldsymbol{x}_{k|t} + B\boldsymbol{u}_{k|t} + D\boldsymbol{d}_{k|t}, \\ & \rho_\beta(F_x \boldsymbol{x}_{k|t} - g_x) \leq 0, \; \boldsymbol{u}_{k|t} \in \mathcal{U}, \\ & \rho_\beta(F_f \boldsymbol{x}_{t+N|t} - g_f) \leq 0 \\ & \boldsymbol{x}_{t|t} = \boldsymbol{x}(t) \quad \forall k \in \{t, \ldots t+N-1\}. \end{aligned}$$

Risk-Aware Safety-Critical Control

The design of the feedback control layer also benefits from a risk-aware approach due to system uncertainties and the potential for adverse outcomes in real-world dynamic systems. Control inputs deemed optimal under a deterministic or risk-neutral framework might carry significant risks due to unpredictability in the system's response, environmental factors, or other variables. Traditional control design methods may fail to consider these risks, potentially resulting in strategies that are vulnerable to unexpected events. While robust controllers target worst-case performance enhancement in [3], they can be overly conservative under unlikely scenarios at the expense of sub-optimal performance under more common circumstances. Risk-aware control system design methodologies can strike a balance between worst-case and typical operating conditions. The resulting systems are not only resilient under extreme circumstances but also optimized for high performance during routine operations. Risk-aware control design thus enhances system robustness by offering a broader performance perspective, effectively bridging the gap between robustness under worst-case scenarios and optimization under nominal performance conditions.

Safety-critical autonomous systems, such as those found in aerospace and human-robot applications, must account for risk. These risks are often associated with the uncertainty of modeling intricate nonlinear dynamics, e.g. bipedal robots [96], and/or sensing extreme unstructured environments, e.g. subterranean or extraterrestrial exploration [97]. Safety is often formulated in terms of set-theoric properties of dynamical systems [98], e.g., reachability and invariance. Safety verification then involves ensuring that system solutions stay within a predefined safe set or, conversely, steer clear of a predetermined unsafe set. A common technique for this is to calculate the reachable set of a system under disturbances and controls [8, 99, 100]. Yet, for intricate, high-dimensional systems, these methods may be impractical or excessively conservative.

Historically, alternative methods for assessing reachability trace back to Nagumo's seminal research [101] on the set invariance of ordinary differential equations (ODEs). This work was later expanded to include ODEs with inputs by Aubin and others, under the framework of viability theory [102]. The rise of interest in hybrid systems in the 2000s led to the development of barrier certificates for safety verification [103]. The creation of these certificates, however, involves solving complex polynomial optimization problems that are challenging for high-dimensional systems, despite some progress made in the last decade [104]. The newer concept of barrier functions [105] offers a solution to the computational difficulties faced by barrier certificates. These functions can be formulated directly from the safe set's definition, simplifying the process. Utilizing this attribute, barrier functions have been effectively applied to design safe controllers (without an existing controller) and safety filters (with an existing controller) for dynamic systems like biped robots [106] and trucks [107]. These applications have demonstrated assured performance and robustness [108].

Conditional Value-at-Risk is a useful measure for assessing how far a realized trajectory may deviate from a safe region of operation [43, 54–58]. Defining safety in terms of CVaR is well-motivated when constraint violations may be unavoidable: the magnitude of the risk should be minimized during the undesired excursion [43, 57]. Sets of initial conditions whose safety is characterized by motions of CVaR can be estimated using dynamic programming [54–58]. Pointwise CVaR constraints have also been studied in [54]. Various problems which optimize a random CVaR objective cost have been studied, such as an*i.e.* upper bound approximation [56], a finite-time solution [57], and an infinite-time solution [58]).

Safety requirements can also be encoded and enforced via Control Barrier Functions (CBFs), which were proposed in [105]. CBFs have been used to design safe controllers for continuous-time dynamical systems, such as bipedal robots [106] and trucks [107], with guaranteed robustness [108, 109] (see the survey [110] and references therein). For discrete-time systems, discrete-time barrier functions were formulated in [111, 112] and applied to multi-robot coordination [113]. For a class of stochastic (Ito) differential equations, safety in probability and statistical mean was studied in [114–118] via stochastic barrier functions.

The first attempt to formulate risk-aware control barrier functions was carried out in [37], wherein the authors proposed CVaR control barrier functions as a composition of a dynamic CVaR metric with a CBF to study safety, in the CVaR sense, for a discrete-time dynamical system subject to stochastic uncertainty. A computational method based on difference convex programs (DCPs) was also proposed in order to synthesize CVaR-safe controllers. The method was applied to collision avoidance scenarios involving a bipedal robot subject to modeling uncertainty. The CVaR control barrier functions were generalized to risk-aware control barrier functions (RCBFs) with general coherent risk measures in [59, 119], where it was shown that the existence of such barrier functions implies invariance in a coherent risk sense. Furthermore, conditions were proposed based on finite-time RCBFs to guarantee finite-time reachability to a desired set. In recent work [120], sampling-based under-approximations of the CVaR for belief states were used to define risk CBFs.

Sidebar 2 (Risk-Aware Control Barrier Functions)**.** *Consider a discrete-time stochastic system given by*

$$\boldsymbol{x}(t+1) = f\left(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t)\right), \quad \mathbb{P}(x(0) = x_0) = 1, \quad (19)$$

where at time $t \in \mathbb{N}_{\geq 0}$, $\boldsymbol{x}(t) \in \mathcal{X} \subset \mathbb{R}^n$ is the state, $\boldsymbol{u}(t) \in \mathcal{U} \subset \mathbb{R}^m$ is the control input, $\boldsymbol{d}(t) \in \mathcal{D}$ is the stochastic uncertainty/disturbance, and $f : \mathbb{R}^n \times \mathcal{U} \times \mathcal{D} \to \mathbb{R}^n$. We assume that the initial condition $\boldsymbol{x}_0$ is deterministic and that $|\mathcal{D}|$ is finite, i.e., $\mathcal{D} = \{v_1, \ldots, v_{|\mathcal{D}|}\}$. At every time step t, for a state-control pair $(\boldsymbol{x}(t), \boldsymbol{u}(t))$, the process disturbance $\boldsymbol{d}(t)$ is drawn from set $\mathcal{D}$ according to the probabilities $p = [p_1, \ldots, p_{|\mathcal{D}|}]^T$, where $p_i := \mathbb{P}(\boldsymbol{d}(t) = v_i)$, $i = 1, 2, \ldots, |\mathcal{D}|$. Note that the probability mass function for the process disturbance is time-invariant, and that the process disturbance is independent of the process history and of the state-control pair $(\boldsymbol{x}(t), \boldsymbol{u}(t))$. Note that, in particular, system (19) can capture stochastic hybrid systems, such as Markovian Jump Systems.

In risk-aware safety analysis, we are interested in studying the properties of the solutions to (19) with respect to the compact set $\mathcal{S}$ described by:

$$\begin{aligned} \mathcal{S} &:= \{\boldsymbol{x} \in \mathcal{X} \mid h(\boldsymbol{x}) \geq 0\}, \\ \mathrm{Int}(\mathcal{S}) &:= \{\boldsymbol{x} \in \mathcal{X} \mid h(\boldsymbol{x}) > 0\}, \\ \partial\mathcal{S} &:= \{\boldsymbol{x} \in \mathcal{X} \mid h(\boldsymbol{x}) = 0\}, \end{aligned} \quad (20)$$

where $h : \mathcal{X} \to \mathbb{R}$ is a continuous function.

In the presence of stochastic uncertainties $\boldsymbol{d}$, assuring almost sure (with probability one) invariance or safety may not be feasible. Moreover, enforcing safety in expectation is only meaningful if the law of large numbers can be invoked and we are interested in the long-term performance, independent of the actual fluctuations. RCBFs focus on safety in the dynamic coherent risk measure sense with conditional expectation as a special case, allowing for more robust measures of safety.

Definition 6 (ρ-Safety)**.** *Given a "safe set" $\mathcal{S}$ in (20) and a time-consistent, dynamic coherent risk measure $\rho_{0:t}$, the solutions to (19), starting at $\boldsymbol{x}_0 \in \mathcal{S}$, are* ρ-safe *if and only if*

$$\rho_{0,t}\left(0, 0, \ldots, h(\boldsymbol{x}(t))\right) \geq 0, \quad \forall t \in \mathbb{N}_{\geq 0}. \quad (21)$$

When $\boldsymbol{x}_0 \in \mathcal{X} \setminus \mathcal{S}$, we often want to know if $\mathcal{S}$ can be reached in finite time.

Definition 7 (ρ-Reachability)**.** *Consider system (19) with initial condition $\boldsymbol{x}_0 \in \mathcal{X} \setminus \mathcal{S}$. Given a set $\mathcal{S}$ as in (20) and a time-consistent, dynamic coherent risk measure $\rho_{0:t}$, $\mathcal{S}$ is* ρ-reachable *if and only if there exists a constant t^* such that*

$$\rho_{0,t^*}\left(0, 0, \ldots, h(\boldsymbol{x}(t^*))\right) \geq 0. \quad (22)$$

Risk-Aware Safety with RCBFs

Definition 8 (Risk-Aware Control Barrier Function)**.** *For the discrete-time system (19) and a dynamic coherent risk measure ρ, the continuous function $h : \mathbb{R}^n \to \mathbb{R}$ is a* risk-aware control barrier function *(RCBF) for the set $\mathcal{S}$ as defined in (20), if there exists a convex class-$\mathcal{K}$ function α satisfying $\alpha(r) < r$ for all $r > 0$ such that*

$$\rho\left(h(\boldsymbol{x}(t+1))\right) \geq \alpha(h\left(\boldsymbol{x}(t)\right)), \quad \forall \boldsymbol{x}(t) \in \mathcal{X}. \quad (23)$$

In [59], the authors demonstrated that the existence of an RCBF implies invariance/safety in the coherent risk measure.

Theorem 1. *For discrete-time system (19) and the set $\mathcal{S}$ as* *described in* (20)*, let ρ be a coherent risk measure. Then, $\mathcal{S}$ is ρ-safe if there exists an RCBF as defined in Definition 8.*

Note that the most common choice for function α is a constant $\alpha = \alpha_0$, where $\alpha_0 \in (0, 1)$, as $\alpha_0 r < r$, $\forall r > 0$. To study risk-aware reachability, we require the following.

Definition 9 (Finite-Time RCBF)**.** *For discrete-time system* (19) *and dynamic coherent risk measure ρ, the continuous function $h : \mathcal{X} \to \mathbb{R}$ is a* finite-time RCBF *for set $\mathcal{S}$, as defined in* (20)*, if there exist constants $0 < \gamma < 1$ and $\varepsilon > 0$ such that*

$$\rho\left(h(\boldsymbol{x}(t+1))\right) - \gamma h(\boldsymbol{x}(t)) \geq \varepsilon(1-\gamma), \quad \forall \boldsymbol{x}(t) \in \mathcal{X}. \tag{24}$$

It was also shown in [59] that the existence of a finite-time RCBF implies ρ-reachability.

Theorem 2. *Consider the discrete-time system* (19) *and a dynamic coherent risk measure ρ. Let $\mathcal{S} \subset \mathcal{X}$ be as described in* (20)*. If there exists a finite-time RCBF $h : \mathcal{X} \to \mathbb{R}$ as in Definition 9, then for all $\boldsymbol{x}(0) \in \mathcal{X} \setminus \mathcal{S}$, there exists a $t^* \in \mathbb{N}_{\geq 0}$ such that $\mathcal{S}$ is ρ-reachable, i.e., inequality* (22) *holds. Furthermore,*

$$t^* \leq \log\left(\frac{\varepsilon - h\left(\boldsymbol{x}(0)\right)}{\varepsilon}\right) \Big/ \log\left(\frac{1}{\gamma}\right), \tag{25}$$

where the constants γ and ε are as defined in Definition 9.

Case Study: Risk-Aware Robotic Motion Planning in Subterranean Environments

This case study looks at a hierarchical risk-aware *traversability* and planning methodology that can be used for autonomous robot (legged or wheeled robot) traversal over extreme terrain [52, 53], as motivated by the DARPA Subterranean challenge. We first need to interpret which parts of the environment the robot can traverse. Evaluation of a natural terrain's *traversability* is difficult due to uncertainties arising from sensor noise and robot localization errors. Furthermore, there are multiple sources of terrain hazards such as steep slopes, loose surface material, sudden elevation drops, and physical obstacles. To account for these different sources of uncertainty systematically, we evaluate the conditional value-at-risk of the terrain hazards to obtain a *risk map* that can be used in the planning and control pipeline. The traversability estimate is given by the random variable $R : (\mathcal{M} \times \mathcal{X} \times \mathcal{U}) \longrightarrow \mathbb{R}$ that maps from the map belief, the robot state, and the applied control to a real-valued traversability cost that we use to assess the CVaR value. This CVaR risk evaluation, $\text{CVaR}_\beta(R)$, enables a robot engineer to define the allowable traversability risk level based on the mission criteria. Furthermore, one can dynamically adjust the risk level, β, online based on 1) the mission-level states, i.e., based on the robot's capabilities and the environment, and 2) whether the robot is stuck in a situation wherein there is no feasible path and decreasing the risk-level (and consequently being less conservative) might allow the robot to find a feasible, but possibly riskier path. The geometric planner and the kinodynamic MPC controller then utilize the risk evaluation, $\text{CVaR}_\beta(R)$, to obtain a risk-aware control policy.

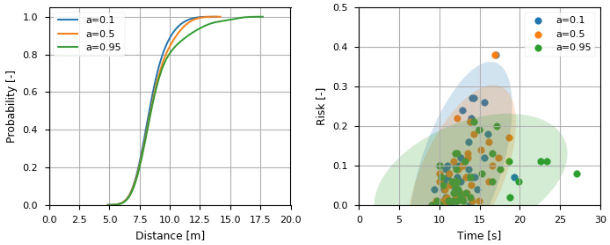

FIGURE 4: **Left:** Trade-off between the probability of reaching the goal and the distance traversed by the robot for different risk levels. **Right:** the trade-off between risk taken and time taken to reach the goal for different risk levels. Note that $\beta = 1 - a$. Figure taken from [52].

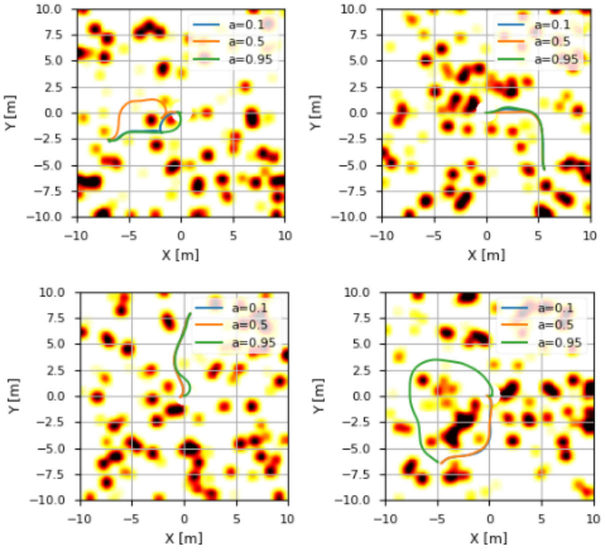

FIGURE 5: Four instances from a Monte-Carlo simulation illustrate how different choices of risk levels, $\beta = 1 - a$, affect the paths taken by the robot. Figure taken from [52].

The statistical performance of the aforementioned risk-aware controllers is evaluated in simulation with randomly generated environment maps and goals. This study illustrates the trade-off between the risk taken by the robot to reach the goal versus the time taken and total distance traversed by the robot for different allowable risk levels, β (see Figure 4). The robot uses longer, low-risk paths when the robot is risk-averse (low β) and shorter, higher-risk paths when the robot is risk-neutral (high β), see Figure 5.

This risk-aware traversability evaluation and planning framework was experimentally tested during the DARPA Subterranean Challenge and in other real-world subterranean environments. The final competition course of the DARPA Subterranean Challenge was comprised of tunnel, urban, and cave environments for which the traversability evaluation and navigation results are provided in Figure 6. The following list describes the difficult terrain hazards found within that environment:

Region A An office-like area with narrow corridors and small rooms where it is tough to find a feasible path if the maps are overinflated to avoid obstacles.

Region B A mock post-earthquake warehouse whose shelving and clutter is difficult to navigate around.

Region C A door connecting the urban and tunnel part of the course via stairs. The stairs act as a potential sudden drop-off (*i.e.*, a negative obstacle) for wheeled robots. The drop is hard to detect because of the narrow doorway.

Region D A narrow passage littered with debris, vertical pipes along the walls, and ceiling obstacles. The robot must correctly identify the pipes as obstacles.

Region E A small cave opening that mimics real caves, wherein humans must crawl through the small openings to reach another cave chamber. The upward-sloping cave floor and downward-sloping ceiling make it difficult to differentiate between the ceiling and the ground. The ceiling height at the opening is very close to the ground height at the end of the opening.

Region F A small limestone cave with rubble and loose rock piles. The robot must distinguish between traversable and non-traversable rubble.

A statistical analysis of the simulations and the experimental results from the field clearly show that a risk-aware traversability and planning pipeline provides a framework where the risk of the *entire* system can be adjusted by changing the risk-level β despite there being multiple risk sources, such as slopes, obstacles, low-ceilings, and mud. This framework is agnostic to the kind of ground robot utilized: it has been tested on wheeled robots (Clearpath Husky) and legged robots (Boston Dynamics Spot quadruped).

Case Study: Risk-Aware 3D Bipedal Walking

Control of bipedal walking presents significant challenges, as evidenced by the variety of approaches taken in the literature to handle the nonlinearity and complexity of bipedal robot dynamics [122]. In practice, bipedal walking dynamics are often simplified by approximate models subject to stochastic uncertainty [123]. The horizontal robot state at time t is denoted by $\boldsymbol{x}_h(t) = [c, p, v]^T$, where c represents the horizontal position of the robot's center of mass (COM) relative to an inertial frame, p denotes the horizontal COM position with respect to the stance foot, and v denotes the horizontal COM velocity. The step-to-step (S2S) dynamics of the horizontal COM state is expressed as $\boldsymbol{x}_h(t+1) = \mathcal{P}^h(\boldsymbol{x}(t), \tau(t))$, where τ represents the joints' input torques. In practice, deriving the S2S dynamics analytically is challenging due to the robot's nonlinear and hybrid dynamics.

The authors in [123, 124] suggested that a Hybrid-Linear Inverted Pendulum (H-LIP) walking model [124] provides an apt approximation for the actual horizontal S2S dynamics of robot walking. The H-LIP dynamics are represented as

$$\boldsymbol{x}_{\text{H-LIP}}(t+1) = A\boldsymbol{x}_{\text{H-LIP}}(t) + B\boldsymbol{u}_{\text{H-LIP}}(t), \tag{26}$$

where $\boldsymbol{x}_{\text{H-LIP}}(t+1) = [c_{\text{H-LIP}}, p_{\text{H-LIP}}, v_{\text{H-LIP}}]^T$ is the H-LIP's discrete pre-impact state, and $\boldsymbol{u}_{\text{H-LIP}}(t)$ denotes the step size. The specific expressions for A and B can be found in [123]. With this approximation, the S2S dynamics can be rewritten as

$$\boldsymbol{x}_h(t+1) = A\boldsymbol{x}_h(t) + B\boldsymbol{u}(t) + \boldsymbol{d}(t), \tag{27}$$

where $\boldsymbol{d}(t) := \mathcal{P}^h(x(t), \tau(t)) - A\boldsymbol{x}_h(t) - B\boldsymbol{u}(t) \in \mathcal{D}$ can be viewed as a stochastic disturbance to the linear system.

As seen in Figure 7, a CVaR RCBF-based controller can ensure safe, risk-aware 3D bipedal walking. The model discrepancy w is treated as a stochastic uncertainty and risk factor that could elicit undesirable walking behavior. To mitigate this risk, CVaR RCBF-based controllers are synthesized and act as safety filters for the H-LIP-based stepping controller. These barrier functions delineate safe regions in terms of robot COM's horizontal position, ensuring the robot's path is maintained within these obstacle-free areas. Estimation of the uncertainties $\boldsymbol{d}$ is carried out using extensive simulations that cover a wide range of walking behaviors. These simulations yield a polytopic set that bounds the uncertainties $\boldsymbol{d}$. A CVaR metric can be calculated on a uniform distribution over the uncertainties represented in the trajectory $\boldsymbol{d}$, which in turn supports the construction of risk-aware barrier function and risk-averse safe feedback controllers for the bipedal robot.

The simulation results seen in Figure 8 showcase the effectiveness of the CVaR-based CBF, especially when taking into account the inherent uncertainties in 3D bipedal dynamics. Figure 9 presents snapshots from an experiment conducted at the Caltech AMBER Lab using the Agility Robotics' Cassie bipedal Robot.

Open Questions and Future Directions

This section outlined recent advances in risk-aware planning and control. We applied these ideas to bipedal walking and terrain traversability analysis for wheeled and legged robots. Many future research directions are suggested by current work in risk-aware planning and control.

Computation. The best choice of a risk measure for a specific problem remains an open question. The popular CVaR and TVD risk measures are computationally attractive as they can be formed into linear programs. Other risk measures such as the KL divergence-based EVaR metric or the Wasserstein metric provide rich expressions of the uncertainty but are computationally expensive.

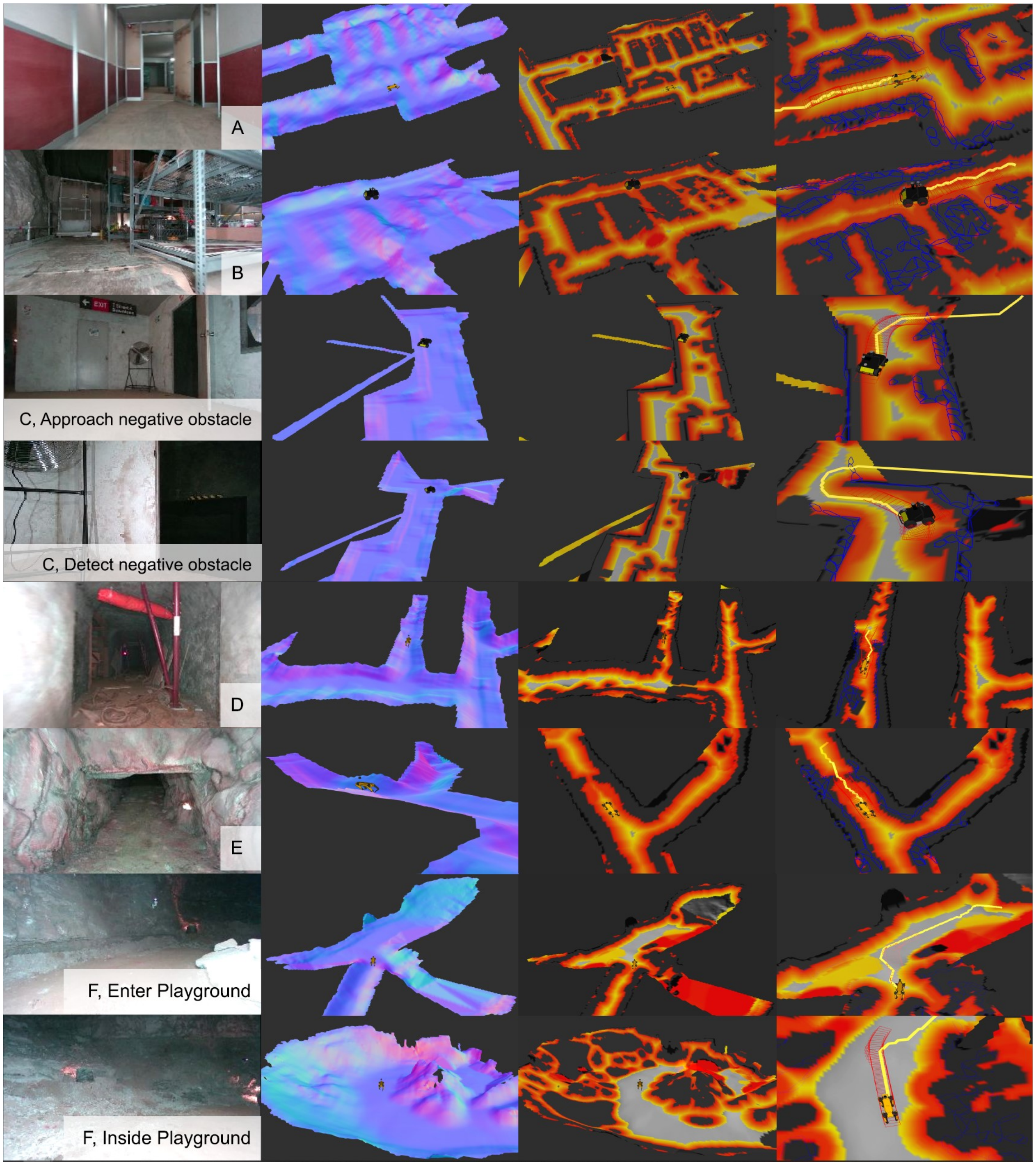

FIGURE 6: Risk-aware traversability analysis of the DARPA Subterranean Challenge final course. Columns, from left to right: robot front camera view, elevation map (colored by normal direction), risk map (colored by risk level - white: safe, yellow to red: moderate, black: risky), and planned geometric/kinodynamic paths (yellow lines/red boxes). Figure taken from [52].

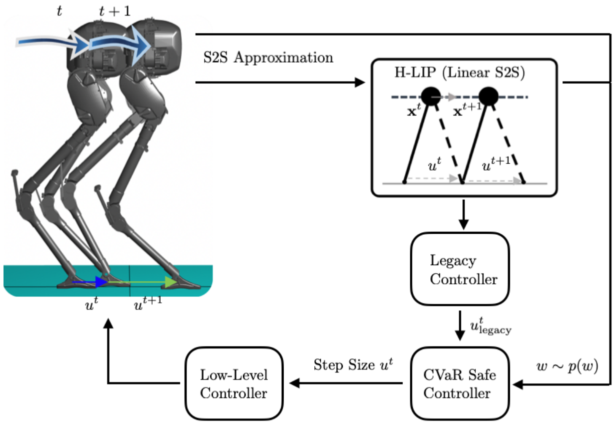

FIGURE 7: Schematic diagram of a risk-aware bipedal robot path planning method based on RCBFs with CVaR risk measure [121]. The diagram features the sequential flow of processes starting with the SS2 Approximation of robot dynamics, which leverages the dynamics of the walking robot, modeled after the Hybrid-Linear Inverted Pendulum (H-LIP) system, where $\boldsymbol{x}^t \equiv \boldsymbol{x}(t)$ denotes the horizontal position of the center of mass (COM) of the robot relative to the inertia frame. The H-LIP approximation is controlled via a legacy controller, such as a model predictive control, which outputs step size commands u_{legacy}. The difference in terms of the horizontal position of the COM between the H-LIP model and simulation/-experimental is measured offline and used to construct the discrete distribution over the uncertainty $p(w)$. The distribution over uncertainty is used to tune a CVaR-safe controller based on RCBFs. The outputs of the legacy controller are then amended online using the CVaR safe controller to ensure risk-aware safety in the presence of uncertainty $p(w)$, which adjusts the robot's locomotion parameters (in particular, step size u) in real-time.

Multi-agent interactions. Our discussion of risk-aware planning and control only considered a single agent. However, real-world dynamic agents may react to the motion of the controlled agent, and these effects could potentially cause unmodeled uncertainty distribution shifts. An important open problem is how to account for the interactions between dynamic agents in a risk-aware manner.

Approximations of risk. Many risk-aware planning and control techniques either assume that the uncertainty is discrete or use approximation techniques like Sample Average Approximation (SAA) for continuous distributions. How can we guarantee the correctness of risk evaluation and control design when using continuous distribution approximations? The next section introduces methods to verify the risk-aware behavior of autonomous systems.

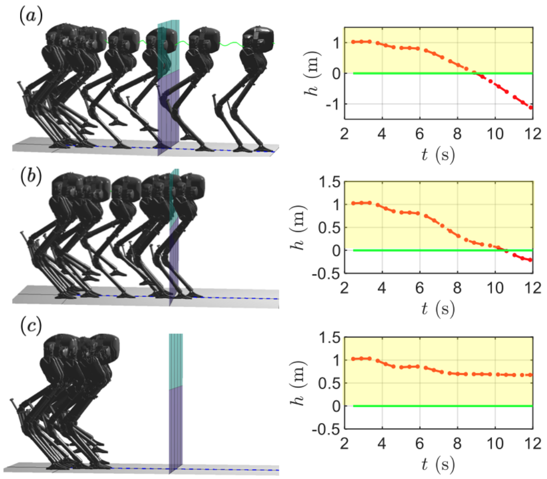

FIGURE 8: Risk-averse obstacle avoidance using CVaR barrier functions (robot behavior and barrier function evolution). The shaded yellow area denotes safe regions. (a) safety violation with no barrier function; (b) safety violation with risk-neutral barrier function; (c) safe behavior with CVaR barrier function. Plots on the right side show the values of the barrier functions [121].

RISK-AWARE VERIFICATION AND VALIDATION

The previous section provided a high-level summary of risk-aware planning and control, with a more in-depth review of existing literature. We highlighted the importance of analyzing the inherent uncertainties and risks in robotic operations, particularly when navigating through unstructured environments. This section briefly summarizes the important companion problem of "risk-aware verification and validation" (V&V) in robotic systems. The verification process determines whether a given system exhibits its desired behavior in the environments in which it is required to operate. This crucial process ensures that the integrated system performs safely, reliably, and as intended under a broad range of operating conditions.

Integration of risk awareness into the V&V process allows for a more comprehensive evaluation of a robotic system and its potential to properly respond to risky situations. Specifically, V&V aims to quantify robotic reliability and safety, explicitly considering interactions with uncertain and unstructured environments [125–128]. As such, risk-aware V&V requires probabilistic risk assessments, stochastic models, and rigorous testing methods that cover a wide range of potential scenarios. For instance, recent work in Human-Robot Interaction (HRI) has developed procedures to quantify the riskiness of actions taken by systems in a collaborative context [129–132]. These methods typically formalize the risk assessment against existing International Standards, *e.g.* ISO 14121 [133] and ISO 12100 [134]. Risk assessment of autonomous systems in other fields has also emerged [135–138].

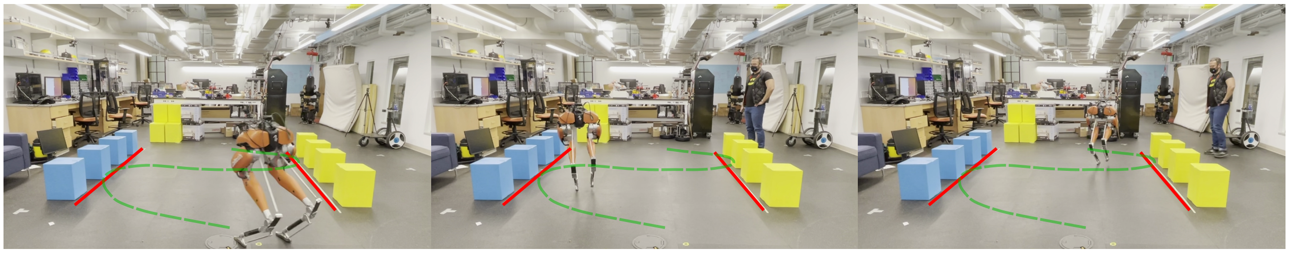

FIGURE 9: Snapshots from an experiment conducted at Caltech's AMBER Lab, featuring Agility Robotics' Cassie Robot. The OptiTrack system that was used for localization tracks reflective markers on the robot to determine its position and orientation with high precision. The data from OptiTrack was fed into the robot's control system in real-time, allowing it to make immediate adjustments to its path. The code that enables the robot to perform motion planning is executed on a computer embedded within the robot. (left) The initial setup, where the robot's trajectory is aligned with a sinusoidal path, represented by a dashed green line. (middle) Mid-course navigation highlighting the effectiveness of the risk control barrier function-based safety filter. This filter is designed to allow the robot to dynamically avoid obstacles and unsafe regions, which are indicated by the red solid lines, representing the boundaries of areas the robot should not enter. (right) successfully following a trajectory that has been adjusted by the risk-aware safety filter. This demonstrates the practical application of the risk-aware control method outlined in Figure 7, where a robot not only plans its path in consideration of potential risks but also dynamically adjusts its course in real time to maintain safe navigation.

As mentioned, prior notions of risk have typically been defined against a corresponding standard and are developed on a case-by-case basis. This observation has prompted recent work in risk-aware verification to adopt the same formal treatment of risk — tail-risk measures — as used by the synthesis community [21, 139]. This section delves into the key methodologies and approaches employed for risk-aware verification and validation in robotics, starting with a brief overview of its theoretical foundations and moving toward practical applications and case studies. As works in this area typically exploit the quantifiable semantics of temporal logics to quantify (un)safe system behavior to make risk-aware verification statements, we begin with a brief overview of temporal logics [140–142].

Temporal Logic

Temporal logics can be used to express complex system specifications and were originally developed for the analysis and design of reactive systems, i.e., systems with external inputs such as control systems [143–145]. Temporal logics are extensions of Boolean logic (propositions, negations, conjunctions, disjunctions) by adding temporal operators (until, eventually, always) to reason about the temporal properties of a system. One can make a distinction between temporal logics that reason over qualitative and quantitative temporal properties in their temporal operators. Linear temporal logic, arguably the most common temporal logic, only reasons over qualitative temporal properties, while real-time temporal logics such as metric temporal logic [146] can reason over quantitative temporal properties. This distinction is best exemplified where a qualitative temporal property is "eventually reach the goal region" while its quantitative counterpart could be "eventually within the next 5 minutes reach the goal region". We focus on temporal logic specifications with quantitative temporal reasoning that can encode combinations of timed reachability ("reach region A within 30 sec"), timed surveillance ("visit regions B, C, and C every $10-60$ sec while agents form a triangular formation"), timed safety ("always between $5-25$ sec stay at least 1 m away from region E"), and many others.

Temporal logics are formally defined by their syntax and semantics where the syntax defines the rules to construct a temporal logic specification ϕ while the semantics define when a temporal logic specification ϕ is satisfied (or violated). Spatiotemporal logics, as opposed to temporal logics, also permit reasoning about spatial properties, e.g., allowing a system designer to quantify to which extent (with what safety margin) an obstacle is avoided by a robot. It is this property that enables us to quantify how well a specification is satisfied *c.f.* how severely a specification is violated, which in turn helps define risk for system verification. Signal temporal logic is a commonly used spatiotemporal logic introduced in [147], and we provide a brief introduction to its syntax and semantics in the sidebar 3. For the remainder of the exposition in this section though, assume that ϕ denotes a system specification.

With respect to the use of these logics in verification and validation, over the past decades, the formal methods community proposed and studied a broad range of system verification techniques. Existing techniques focus on the verification of 1) deterministic systems, or 2) uncertain systems with the two previously discussed viewpoints of the risk-neutral and the worst-case paradigms. In fact, automated verification tools were developed for deterministic systems, e.g., model checking [144, 145] or theorem proving [148, 149]. Verification of uncertain systems was particularly studied in the risk-neutral paradigm using probabilistic model checking [150–152], an extension of deterministic model checking, or statistical model checking [153–156], which are sampling-based techniques for probabilistic system verification. System verification techniques, however, should not only be able to reason about the probability of violating a specification but also be able to reason about the severity of a violation in terms of rare harmful outcomes. As briefly discussed previously, spatiotemporal logics enable us to quantify how well (severely) a specification is satisfied (violated), and in turn, allow us to define risk in terms of this quantitative measure for stochastic systems.

Sidebar 3 (Signal Temporal Logic: Syntax and Semantics)**.** *Signal temporal logic (STL) specifications are interpreted over continuous-time signals $\boldsymbol{x} : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$. An STL specification ϕ is recursively constructed from atomic predicates by using Boolean operators and temporal operators. These atomic predicates are Boolean-valued functions $\mu : \mathbb{R}^n \to \{1, 0\}$ whose truth value is obtained after evaluation of a real-valued function $b : \mathbb{R}^n \to \mathbb{R}$. At time t, we obtain the truth value of μ as*

$$\mu(x(t)) := \begin{cases} 1 & \text{if } b(x(t)) \geq 0 \\ 0 & \text{otherwise.} \end{cases} \tag{28}$$

Predicate functions h can encode relationships between state variables, such as relative or absolute distances. The syntax of STL defines a set of rules according to which well-defined STL specifications can be constructed and is given as

$$\phi \ ::= \ 1 \mid \mu \mid \neg\phi' \mid \phi' \wedge \phi'' \mid \phi' U_I \phi'' \tag{29}$$

where the operators $\neg$, $\wedge$, and U_I encode negations, conjunctions, and the until over the time interval $I \subseteq \mathbb{R}_{\geq 0}$, respectively. The syntax in (29) can be understood as follows: the symbol ::= assigns one of the expressions from the right-hand side, which are separated by vertical bars, to the free variable ϕ on the left-hand side. The variables ϕ' and ϕ'' on the right-hand side are already well-defined STL specifications. While the meaning of negations and conjunctions is clear, the until operator $\phi' U_I \phi''$ encodes that ϕ' has to hold until ϕ'' holds, which has to happen within the time interval I. We can now use logical equivalences to derive the Boolean disjunction, implication, and equivalence operators and the temporal eventually and always operators. In what follows, $\top$ denotes True in the corresponding logical specification:

$$\begin{aligned} \phi' \vee \phi'' &:= \neg(\neg\phi' \wedge \neg\phi'') && \textit{(disjunction)}, \\ \phi' \Rightarrow \phi'' &:= \neg\phi' \vee \phi'' && \textit{(implication)}, \\ \phi' \Leftrightarrow \phi'' &:= (\phi' \Rightarrow \phi'') \wedge (\phi'' \Rightarrow \phi') && \textit{(equivalence)}, \\ F_I\phi' &:= \top U_I \phi' && \textit{(eventually)}, \\ G_I\phi &:= \neg F_I \neg\phi' && \textit{(always)}. \end{aligned}$$

The semantics of STL now define when a signal x satisfies an STL specification ϕ. These semantics are formally defined as a relation $\models$ between x and ϕ, and $(x, t) \models \phi$ means that the signal x satisfies the specification ϕ at time t. We recursively define the semantics as

$$\begin{aligned} (x,t) &\models 1 && \textit{iff} \quad \textit{holds by definition}, \\ (x,t) &\models \mu && \textit{iff} \quad h(x(t)) \geq 0 \\ (x,t) &\models \neg\phi' && \textit{iff} \quad (x,t) \not\models \phi' \\ (x,t) &\models \phi' \wedge \phi'' && \textit{iff} \quad (x,t) \models \phi' \textit{ and } (x,t) \models \phi'' \\ (x,t) &\models \phi' U_I \phi'' && \textit{iff} \quad \exists t'' \in t \oplus I \textit{ s.t. } (x,t'') \models \phi'' \textit{ and} \\ & && \qquad \forall t' \in (t, t''),\ (x,t') \models \phi'. \end{aligned}$$

Motivations for Tail-Risk Measures in Verification

As mentioned previously then, the existence of these spatiotemporal logics makes tail-risk measures uniquely suited to serve as the risk measure of choice for verification and validation, and this section will provide an example supporting that claim using the notation offered in Sidebar 4. Consider for the sake of argument that we have two controlled systems Σ_1, Σ_2, a Signal Temporal Logic specification ϕ denoting the desired behavior required of both systems and a robustness measure ρ^ϕ for the same specification ϕ. For context, every signal temporal logic specification ϕ comes equipped with a quantitative measure. Let's further assume that every time we query either system Σ_1 or Σ_2, we receive a random trajectory $\boldsymbol{x}_1$ or $\boldsymbol{x}_2$ respectively. Let R_i, $i = 1, 2$, be a random variable whose samples r_i are the robustness of the trajectories sampled from system Σ_i, *i.e.* $r_i = \rho^\phi(\boldsymbol{x}_i)$. Now, let's assume that an oracle tells us that for both systems, with probability $1 - \beta$ for some $\beta \in (0, 1]$, the random robustness exceeds a cutoff value $\epsilon > 0$. Furthermore, with probability β, Σ_1 exhibits robustness values between ϵ and 0, whereas Σ_2 exhibits robustness values strictly less than 0. In other words, Σ_1 will always realize the desired behavior, while Σ_2 will, in some rare cases, be unable to realize the desired behavior.

Put into the context of tail-risk measures, both systems exhibit a robustness value-at-risk at risk-level β that is positive. This fact arises as the oracle mentioned that with minimum probability $1 - \beta$, both systems exhibit robustnesses exceeding $\epsilon > 0$, *i.e.* $\text{VaR}_\beta(R_i) = \epsilon > 0$. Ending the analysis here results in the correct conclusion that both systems exhibit some minimum probability of realizing satisfactory behavior, and this is a traditional, probabilistic V&V statement. However, by considering conditional value-at-risk, we can further discriminate between the two systems, as system 1 is expected to exhibit satisfactory behavior even in the worst 100β% of cases, whereas system 2 is expected to produce unsatisfactory behavior in the same cases. This conclusion arises as even in the worse 100β% of cases, the oracle mentioned that system 1 exhibits robustness values $r_1 \in [0,\epsilon]$ whereas system 2 exhibits robustness values $r_2 < 0$. Taking into account the expected value over those cases - the definition of conditional-value-at-risk - we conclude that $\mathrm{CVaR}_\beta(R_1) \geq 0$ whereas $\mathrm{CVaR}_\beta(R_2) < 0$. Therefore, even if both systems exhibit similar minimum probabilities of specification satisfaction, system 1 is "better" than system 2 as it is still expected to exhibit satisfactory behavior in the worst $100\beta\%$ of cases.

The example described above highlights the utility of tail-risk measures in risk-aware V&V. By considering the robustness value-at-risk, we can make statements on the minimum probability with which a system exhibits a desired behavior in its operating environment(s) - this is the traditional notion of probabilistic V&V. Additionally, we can utilize tail-risk measures to also make statements on expected worst-case robustness using the conditional-value-at-risk, lower bound such expected worst-case robustness using entropic value-at-risk, and calculate these values without exact distribution knowledge as will be described in sections to follow.

Sidebar 4 (Signal Temporal Logic: Robust Semantics). *While the STL semantics tell us if a signal $\boldsymbol{x} : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ satisfies an STL specification ϕ, it does not give us any information about the quality of satisfaction. To obtain such information, one can define robust semantics that quantify how robustly the signal $\boldsymbol{x}$ satisfies the specification ϕ. If $\boldsymbol{x}$ satisfies ϕ, we would hence like to quantify how different a signal $\boldsymbol{x}^* : \mathbb{R} \to \mathbb{R}^n$ can be from $\boldsymbol{x}$ while still satisfying ϕ. To quantify this, we first define the closeness of two signals $\boldsymbol{x}, \boldsymbol{x}^* : \mathbb{R} \to \mathbb{R}^n$ as*

$$d(\boldsymbol{x},\boldsymbol{x}^*) := \sup_{t \in \mathbb{R}_{\geq 0}} \|\boldsymbol{x}(t) - \boldsymbol{x}^*(t)\|.$$

We now want to compute a value ρ^ϕ such that all signals $\boldsymbol{x}^*$ that are such that $d(\boldsymbol{x},\boldsymbol{x}^*) < \rho^\phi$ will also satisfy ϕ, i.e., $(\boldsymbol{x}^*,t) \models \phi$. To do so, we first define the signed distance of the signal value $\boldsymbol{x}(t)$ to the set of states that satisfy a predicate μ, denoted by $O^\mu : \{x \in \mathbb{R}^n | b(x) \geq 0\}$, as

$$Dist(\boldsymbol{x}(t), O^\mu) := \begin{cases} \inf\limits_{x^* \in cl(\mathbb{R}^n \setminus O^\mu)} \|x^* - \boldsymbol{x}(t)\| & \textit{if } \boldsymbol{x} \in O^\mu \\ -\inf\limits_{x^* \in cl(O^\mu)} \|x^* - \boldsymbol{x}(t)\| & \textit{otherwise.} \end{cases}$$

Note that $Dist(\boldsymbol{x}(t), O^\mu)$ quantifies the extent to which μ is satisfied (if $Dist(\boldsymbol{x}(t), O^\mu) > 0$) or violated (if $Dist(\boldsymbol{x}(t), O^\mu) < 0$). We can now recursively define the robust semantics of ϕ as a real-valued function $\rho^\phi(\boldsymbol{x},t)$ as follows

$$\begin{aligned} \rho^\top(\boldsymbol{x},t) &:= \infty, \\ \rho^\mu(\boldsymbol{x},t) &:= Dist(\boldsymbol{x}(t), O^\mu), \\ \rho^{\neg\phi'}(\boldsymbol{x},t) &:= -\rho^{\phi'}(\boldsymbol{x},t), \\ \rho^{\phi' \wedge \phi''}(\boldsymbol{x},t) &:= \min(\rho^{\phi'}(\boldsymbol{x},t), \rho^{\phi}(\boldsymbol{x},t)), \\ \rho^{\phi' U_I \phi''}(\boldsymbol{x},t) &:= \mathop{sup}_{t'' \in t \oplus I} \min(\rho^{\phi''}(\boldsymbol{x},t''), \mathop{inf}_{t' \in (t,t'')} \rho^{\phi'}(\boldsymbol{x},t')). \end{aligned}$$

Finally, it holds that $(\boldsymbol{x},t) \models \phi$ if and only if $(\boldsymbol{x}^*,t) \models \phi$ for all signals $\boldsymbol{x}^* : \mathbb{R} \to \mathbb{R}^n$ that are such that $d(\boldsymbol{x},\boldsymbol{x}^*) < |\rho^\phi(\boldsymbol{x},t)|$.

A General Overview of Risk-Aware V&V

For most relevant problems, the system to be verified can be recast as a discrete-time system with known state and input spaces and (perhaps) known dynamics and disturbance spaces. Formally, at some time $t \in \mathbb{Z}_+ = \{0,1,2,\dots\}$, let $\boldsymbol{x}(t) \in \mathcal{X}$ be the system state, $\boldsymbol{u}(t) \in \mathcal{U}$ be the system control input, $\boldsymbol{d}(t) \in \mathcal{D}$ be a randomly sampled disturbance. Then for some distribution function $\xi : \mathcal{X} \times \mathcal{U} \times \mathbb{Z}_+ \times \mathcal{D} \to [0,1]$ over $\mathcal{D}$,

$$\boldsymbol{x}(t+1) = f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t)), \tag{30a}$$

$$\text{s.t. } \boldsymbol{d}(t) \sim \pi(\boldsymbol{x}(t), \boldsymbol{u}(t), t). \tag{30b}$$

Finally, let $\mathbb{U} : \mathcal{X} \times \Theta \times \mathbb{Z}_+ \to \mathcal{U}$ be the controller closing the loop for the system to be verified. Note, the controller $\mathbb{U}$ is parameterized with a parameter $\theta \in \Theta$ to account for exogenous, user-specific inputs that may influence controller behavior, *e.g.* parameterized 3-space locations for packages in a warehouse that a warehouse robot receives on-the-fly from a central command station when a package is required to be collected.

Remark. For systems that operate in adversarial environments or in the presence of obstacles, the disturbance distribution ξ can be defined as the singleton distribution over the adversarial choice at the given state $\boldsymbol{x}(t)$, input $\boldsymbol{u}(t)$, and time t. For more information see dirac distributions [157] and adversarial testing works such as [158–161]. We can consider both cases — the case with non-adversarial, randomized disturbances and the case with adversarial or otherwise known disturbances — in the same stochastic setting. Furthermore, the state and input spaces have been left arbitrary to allow both the continuous space definitions in the controls community and the finite-state and input definitions in the (PO)MDP computer science literature.

Closing the loop between (30) and the assumed controller $\mathbb{U}$ generates a system Σ that when queried with a specific initial condition $x_0 \in \mathcal{X}_0 \subseteq \mathcal{X}$ and controller parameter $\theta \in \Theta$ produces a (perhaps) different state trajectory $\boldsymbol{x}$, which is an element of the signal space $\mathcal{S} = \{s : \mathbb{Z}_+ \to \mathcal{X}\}$:

$$\text{a sample of } \Sigma(x_0,\theta) \text{ is } \boldsymbol{x} \triangleq \{x_0 \equiv \boldsymbol{x}(0), \boldsymbol{x}(1), \dots\} \tag{31a}$$

$$\text{s.t. } \boldsymbol{x}(t+1) = f(\boldsymbol{x}(t), \mathbb{U}(\boldsymbol{x}(t),\theta), \boldsymbol{d}(t)). \tag{31b}$$

Finally, let $\boldsymbol{C}$ be a classifier function mapping from the signal and parameter spaces to the real-line, *i.e.* $\boldsymbol{C} : \mathcal{S} \times \Theta \to [-a,b]$ where parameters $a,b \in \mathbb{R}_{++}$ are finite. The classifier $\boldsymbol{C}$ delineates between satisfactory behavior — trajectory and parameter pairs $(\boldsymbol{x},\theta)$ that evaluate to a positive value, *i.e.* $\boldsymbol{C}(\boldsymbol{x},\theta) \geq 0$ — and unsatisfactory behavior — pairs that evaluate to a negative value. Examples of such a classifier could be the robustness functions ρ from signal temporal logic, the minimum value of a barrier function h over time [105], *etc*. For a description of robustness measures in Signal Temporal Logic, please see Sidebar 4. Generally speaking, we define the outcome of function $\boldsymbol{C}$ to be the *robustness* of the corresponding trajectory and parameter pair, *i.e.* for $r = \boldsymbol{C}(\boldsymbol{x}, \theta)$, where r is the trajectory and parameter pair's robustness value. More positive values of $\boldsymbol{C}(\boldsymbol{x}, \theta)$ indicate better, *more robust* realization of the desired behavior.

Remark The rationale to analyze multiple, non-unique trajectories $\boldsymbol{x}$ as defined in (31) arises from the fact that disturbances $\boldsymbol{d}$ in (30) are sampled randomly at each time t from the distribution function ξ. If the distribution function ξ were the singleton distribution corresponding to a specific disturbance $d \in \mathcal{D}$ for each state, input, and time $(x, u, k) \in \mathcal{X} \times \mathcal{U} \times \mathbb{Z}_+$, then $\Sigma(x_0, \theta)$ would always produce the same trajectory $\boldsymbol{x}$ upon successive queries at (x_0, θ), and this argument holds $\forall\ (x_0, \theta) \in \mathcal{X}_0 \times \Theta$.

The goal of risk-aware verification stems naturally from the existence of the system trajectory generation function Σ and the classifier $\boldsymbol{C}$. Specifically, the goal is to determine bounds on the risk measure evaluation of $\boldsymbol{C}$ for trajectories $\boldsymbol{x}$ realized by the system Σ at a chosen parameter θ and initial condition x_0. Stated formally, let χ be a tail-risk measure, *e.g.* Value-at-Risk, Conditional-Value-at-Risk, Entropic-Value-at-Risk, *etc*. Then, at some risk-level $\beta \in [0, 1]$, determine an upper or lower bound to $\chi\left(\boldsymbol{C}\left(\Sigma(x_0, \theta)\right)\right)$ for some $(x_0, \theta) \in \mathcal{X}_0 \times \Theta$. Figure 10 depicts this generic risk-aware verification pipeline, and the following section will specify how this pipeline has been implemented in a variety of recent works. To facilitate that discussion, we will define the *robustness* $R(x_0, \theta)$ to be the random variable whose samples $r = \boldsymbol{C}(\boldsymbol{x}, \theta)$, where $\boldsymbol{x}$ is a sample of the random variable $\Sigma(x_0, \theta)$. In other words, $R(x_0, \theta) = \boldsymbol{C}(\Sigma(x_0, \theta))$ — this term was first defined in [15].

Examples

Perhaps the most prevalent examples of risk-aware verification arise from a re-framing of traditional work in the Stochastic Model Checking (SMC) community [156, 162–164]. With respect to the aforementioned pipeline, SMC assumes the ability to collect system traces — trajectories $\boldsymbol{x}$ — and evaluate their satisfaction of a desired behavior. These behaviors are typically expressed as a specification ϕ in Probabilistic Computational Tree Logic [165], which is a form of Temporal Logic (see Sidebar 3). As such, each of these behaviors has satisfiability metrics — classifier functions $\boldsymbol{C}$ in our overarching methodology — with which to determine trace satisfaction of the desired behavior ϕ.

SMC consists of two different analyses. The first, hypothesis testing, asserts that the system Σ realizes the behavior ϕ with minimum probability p and determines the minimum number of system traces that have to be evaluated to accept or reject this hypothesis. The second, estimation, exploits either the Chernoff bound or Hoeffding's inequality to estimate the probability p with which Σ realizes ϕ within some tolerance bounds that are a function of the number of trajectories sampled and evaluated. In both cases, however, the probability of satisfaction p has a one-to-one correspondence with the Value-at-Risk of the random variable $R(x_0)$ (we omit θ in the notation here, as SMC typically does not consider parameterized trajectories). More specifically, p is such that $\mathrm{VaR}_{1-p}(R(x_0)) \geq 0$.

These are not the only works that take a Value-at-Risk approach to system verification. In [60], the authors use scenario optimization to lower bound the Value-at-Risk of the robustness random variable $R(x_0, \theta)$ for a user-defined $\beta \in [0, 1]$. Similarly, the authors of [61] use a sample-average-approximation procedure to estimate the Value-at-Risk of the same robustness variable for any user-defined $\beta \in [0, 1]$. In [62], the authors go one step further and express policy or controller synthesis as an optimization problem over a general class of risk measures for verification purposes. They show numerical examples of the success of a convex-concave procedure at identifying such policies for a Markov Decision Process. In this case, the policies optimize for a certain risk sensitivity as expressed by Value-at-Risk among other risk measures expressed in Cumulative Prospect Theory [166]. Finally, in [63], the authors modify a learned controller online whenever the learned controller outputs an infeasible trajectory. Via a gradient-descent method, they update controller parameters until the resulting trajectory passes an intermediary risk-aware verification step, before implementation of the modified controller. In general, however, any of the aforementioned risk-aware works could also be conceived of as Value-at-Risk-based verification, insofar as the classifier functions $\boldsymbol{C}$ were developed against specific standards for their respective applications [129–132, 135–138]

However, Value-at-Risk verification represents a smaller fraction of risk-aware verification efforts as compared to works using coherent risk measures, such as Conditional-Value-at-Risk. For example, in a similar paradigm as in [62], in [64] the author proves that there exist polynomial time algorithms to determine policies for an MDP that are verifiable by default. Verification arises as the policies are synthesized to achieve a minimum conditional value at risk with respect to objective satisfaction. Similarly, in [65] the authors utilize a CVaR constraint for their optimal controller and verify that the system remains within a risk-sensitive safe set defined by the same CVaR constraint. In [66], the authors develop a procedure for learning a controller to tackle simultaneous performance and safety tradeoffs for nonlinear systems and verify the learned controller by estimating the CVaR of a corresponding robustness random variable. In [67], the authors constrain

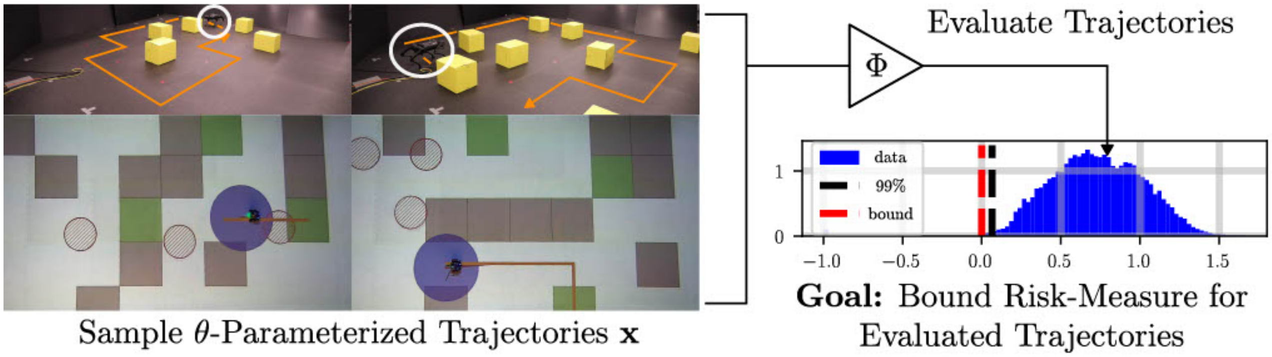
Evaluate Trajectories
Φ
data
99%
bound
1
0
−1.0
−0.5
0.0
0.5
1.0
1.5
Goal: Bound Risk-Measure for Evaluated Trajectories
Sample θ-Parameterized Trajectories x

FIGURE 10: A flowchart for a general risk-aware verification pipeline. In the figure, the parameters θ correspond to obstacle locations and waypoints for the robots highlighted by white and blue circles [77]. Risk-aware verification bounds the risk-measure evaluation of evaluated trajectories — the blue data shown in the right figure.

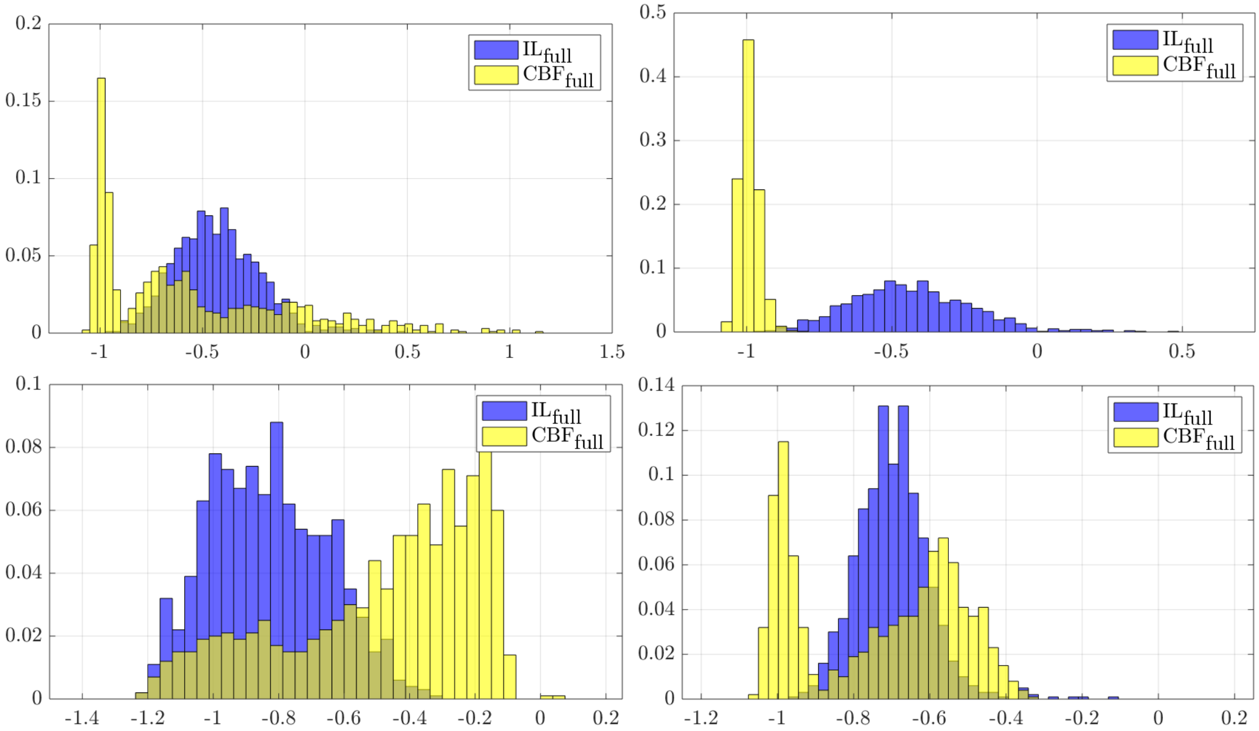

FIGURE 11: Empirical distributions for the Imitation Learning (IL) and Control Barrier Function-based (CBF) controllers and for the specifications ϕ_1-ϕ_4 as described in the case study on risk-aware lane-keeping. Figures taken from [68].

the optimal design of supersonic aircraft bodies against both VaR and CVaR requirements to ensure verifiable, risk-aware performance despite uncertainties arising from the transition from laminar to turbulent flow, manufacturing uncertainties, *etc*. Examples from a larger body of work, including those by the authors, can be found in references [15, 56, 61, 65, 68–74].

Case Study: Risk-Aware Verification of Lane Keeping Controllers

In this case study, the goal is to find the least risky controller among two neural network lane-keeping controllers in the autonomous driving simulator CARLA during a left-turn [75], see Fig. 12 (top and middle). Lanekeeping is realized by tracking a set of predefined waypoints. For verification, we consider the cross-track error c_e and the orientation error θ_e with respect to the current and next waypoint, see Fig. 12 (bottom). We consider an imitation learning (IL) controller [167] and a control barrier function

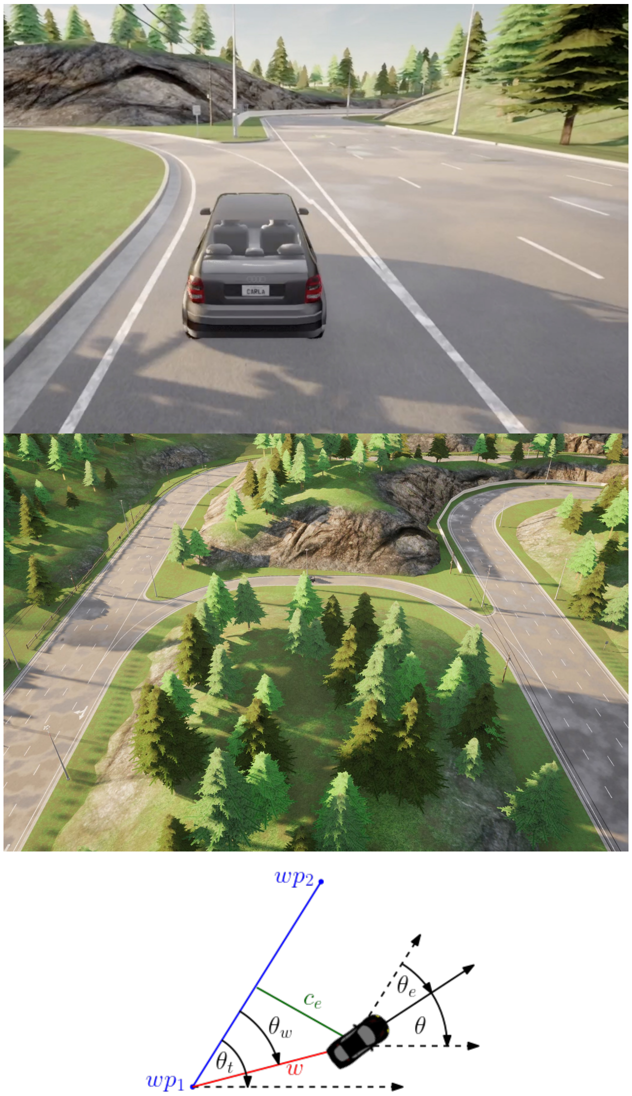

FIGURE 12: Top: Simulation environment in the CARLA autonomous driving simulator. Middle: Left turn on which we evaluate two neural network lane-keeping controllers. Bottom: The car's cross-track error c_e and orientation error θ_e with respect to waypoints. Figures taken from [68].

(CBF) controller [168]. The car model is stochastic, as the control inputs are subject to normally distributed noise, and we uniformly sample the car's initial position from the set $(c_e, \theta_e) \in [-1, 1] \times [-0.4, 0.4]$. We collected $N := 1000$ trajectories $\boldsymbol{x}_i$ in a validation set D_{val} for each controller.

We are first concerned with cross-track error and consider the specifications $\phi_1 := G_{[0,\infty)}(|c_e| \leq 2.25)$ where 2.25 is an empirically obtained threshold indicating that the car stays within the lane. For the following analysis, recall that a negative value of $-\rho^{\phi_1}$ indicates satisfaction of ϕ_1 and positive values indicate a failure to lane-keep. Upper bounds on $VaR_{0.95}(-\rho^{\phi_1}(\boldsymbol{x}))$, $CVaR_{0.85}(-\rho^{\phi_1}(\boldsymbol{x}))$, and

R \ Controller	IL	CBF
$VaR_{0.95}(-\rho^{\phi_1}(\boldsymbol{x}))$	0.462	1.125
$CVaR_{0.85}(-\rho^{\phi_1}(\boldsymbol{x}))$	1.436	1.818
$E(-\rho^{\phi_1}(\boldsymbol{x}))$	-0.248	-0.375
P_{ϕ_1}	0.975	0.863
$VaR_{0.95}(-\rho^{\phi_2}(\boldsymbol{x}))$	0.462	-0.794
$E(-\rho^{\phi_2}(\boldsymbol{x}))$	-0.254	-0.81
$VaR_{0.95}(-\rho^{\phi_3}(\boldsymbol{x}))$	-0.324	0.063
$E(-\rho^{\phi_3}(\boldsymbol{x}))$	-0.652	-0.297
$VaR_{0.95}(-\rho^{\phi_4}(\boldsymbol{x}))$	-0.13	-0.32
$E(-\rho^{\phi_4}(\boldsymbol{x}))$	-0.517	-0.533
P_{ϕ_5}	1	1

TABLE 1: Tabulated data from [68] for the case study on risk-aware lane-keeping of the Imitation Learning (IL) and Control Barrier Function (CBF) controllers.

$E(-\rho^{\phi_1}(\boldsymbol{x}))$ are reported in Table 1, along with the empirical satisfaction rate $P_{\phi_1} := |\{\boldsymbol{x}_i \in D_{\text{val}} | \boldsymbol{x}_i \text{ satisfies } \phi_1\}|/N$. Clearly, the IL controller is the least risky one in terms of $VaR_{0.95}$ and $CVaR_{0.85}$, and it also has the highest empirical satisfaction rate. Interestingly though, the CBF controller performs better on average. This result can also be seen in the empirical histograms of Fig. 11 (top left). We hypothesize that this behavior arises from the long tail of risky behavior for the CBF controller, which corresponds to transient system behavior. We also analyzed the controllers' behavior more closely by looking at the cross-track error during the steady-state and transient phases for the specifications $\phi_2 := G_{[10,\infty)}(|c_e| \leq 2.25)$ and $\phi_3 := F_{[0,5]}G_{[0,5]}(|c_e| \leq 1.25)$, respectively. The upper bounds of the $VaR_{0.95}(-\rho^{\phi_i}(\boldsymbol{x}))$ and $E(-\rho^{\phi_i}(\boldsymbol{x}))$ for ϕ_2 and ϕ_2 are shown in Table 1 as well.

Interestingly, the IL controller is the least risky one only during the transient phase, while the CBF controller is the least risky one in steady state. The corresponding empirical distributions are shown in Figs. 11 (top right and bottom left). Finally, let us verify the controller risk in terms of the orientation error θ_e. Consider $\phi_4 := G_{[0,\infty)}((c_e \geq 1.25) \implies F_{[0,2]}G_{[0,1]}(\theta_e \leq 0) \wedge (c_e \leq -1.25) \implies F_{[0,2]}G_{[0,1]}(\theta_e \geq 0))$ which expresses the need for the controller to react to large cross-track errors c_e using the right orientation adjustment.

For this specification, the CBF controller is the least risky controller, which aligns with our observation that it is a better controller during steady-state. It can further be observed that both controllers have the same empirical satisfaction probability, while our risk analysis better quantifies a controller's quality. The empirical distribution of both controllers is shown in Fig. 11 (bottom right).

Case Study: Risk-Aware Safety-Critical System Verification and Applications to Policy Synthesis

This case study aims to verify a quadruped's ability to render positive a collision-avoidance barrier function h for

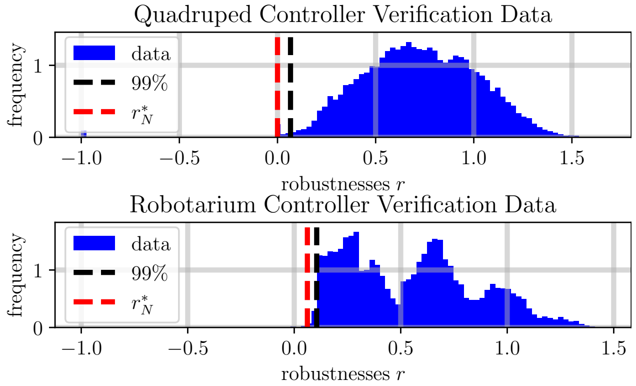

FIGURE 13: Validation data for probabilistic lower bounds reported on $\mathrm{VaR}_{0.99}(R)$ for controllers generated for a quadruped (top) and robotarium (bottom). As shown, the reported lower bounds (red) generated by the scenario approach mentioned in Sidebar 5 are accurate as they lower bounds the true $\mathrm{VaR}_{0.99}(R)$ (black). Information for this figure comes from [76].

at least $T = 150$ time-steps with a time-step $\Delta t = 0.1$ [76]. Keeping consistent with our notation for the general overview for risk-aware V&V, our parameters θ include the locations of 4 randomly placed static obstacles in a 5×5 meter grid, and the center coordinates of a goal region g in the same grid. Hence, $\theta \in [-5,5]^{10} \triangleq \Theta$. To simplify our analysis, we represent the quadruped as a unicycle system, and as such, we assume we can initialize the quadruped at a random planar position and angular orientation in the grid, *i.e.* $x_0 \in [-5,5]^2 \times [0,2\pi] \triangleq \mathcal{X}_0$. Our classifier function $\boldsymbol{C}$ evaluates the discrete-time fractional difference of a candidate barrier function h that the quadruped is to keep positive. As such, the classifier outputs the minimum value over all time-steps k of $h(\boldsymbol{x}(t+1)/h(\boldsymbol{x}(t))$, as realized by the quadruped over one trajectory $\boldsymbol{x} = \{\boldsymbol{x}(0), \boldsymbol{x}(1), \dots, \boldsymbol{x}(150)\}$. Therefore, $\boldsymbol{C}(\boldsymbol{x}) < 0$ is equivalent to stating that there existed an interior time-step $x_j \in \boldsymbol{x}$ such that $h(x_j) < 0$ and the quadruped failed to remain safe.

Slightly different from the general overview, however, instead of aiming to determine a lower bound on the Value-at-Risk level $\beta = 0.9$ of the robustness random variable $R(x_0, \theta)$, we also randomize over initial conditions and parameters (x_0, θ) from their combined space. As such, the evaluation r of a sampled trajectory $\boldsymbol{x}$ generated by first sampling $(x_0, \theta) \sim \mathrm{U}[\mathcal{X}_0 \times \Theta]$ is a sample of the *holistic robustness random variable R* — this term was first defined in [15]. That being said, we still aim to lower bound $\mathrm{VaR}_{\beta=0.9}(R)$, which, according to the sample-based methods detailed in Sidebar 5, has a known sample complexity (number N of trajectories to be evaluated) to determine such a lower bound. Therefore, after taking $N = 50$ trajectories, we can state with ≈ 99.5% confidence, that the quadruped will realize positive value trajectories with 90% probability, as the identified lower bound on $\mathrm{VaR}_{\beta=0.9}(R)$ was positive. The associated safety information is depicted in Figure 14.

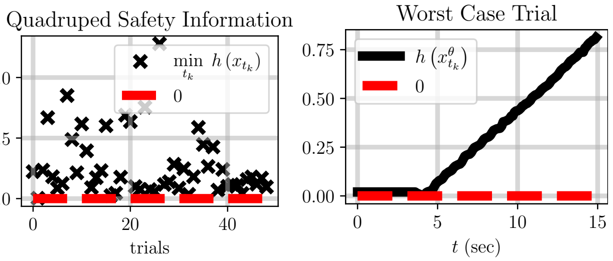

FIGURE 14: Worst case safety information for the quadrupedal case study. Over the 50 sampled trials, the quadruped realizes a positive barrier value every time, which, according to the concentration inequality results in Sidebar 5, implies that the system should always keep positive the barrier function h with minimum probability 90% and with confidence 99.5%. Information for this figure comes from [76].

We can also show that the reported probabilistic bounds are accurate. In [77] we employed the same, risk-aware verification procedure as described above, to validate the controllers for a Quadruped and a multi-agent robotic system [169]. The expressions for the classifier function were the same in both cases, though we will refrain from reproducing them here for the sake of brevity. Suffice it to say that any trajectory that evaluates to a positive value under $\boldsymbol{C}$ would have made non-trivial progress toward a goal while avoiding static/moving obstacles within 10 seconds. To that end then, we only implemented controllers on hardware systems once they exhibited a positive lower bound for $\mathrm{VaR}_{0.99}(R)$. To determine such a lower bound we sampled 300 trajectories for both controllers and calculated their robustness under the aforementioned classifier $\boldsymbol{C}$. Doing so for our chosen controllers indicated positive lower bounds, and we can verify the accuracy of these lower bounds by taking 20000 trajectories, evaluating them, numerically approximating the distribution of the holistic robustness random variable R, and reporting the numeric $\mathrm{VaR}_{0.99}(R)$ as our approximation. Figure 13 showcases the validity of the reported lower bounds, overlaid on the numeric approximation of the distribution for R, and Figure 15 shows tiles of the controllers implemented on their respective hardware systems. Note that in all of the randomized cases, the controllers steered the systems successfully to their goals while avoiding all obstacles. This controller reliability is the primary reason we take a tail-risk approach to verification, as the purpose of the procedure is to identify rare, unsafe phenomena and ensure that even in those rare cases, the system still performs admirably.

Sidebar 5 (Concentration Inequalities: Risk Measure Estimation)**.** *This sidebar will briefly describe two methods to estimate the tail risk measures expounded upon in this article. We will describe these methods as they are applied to arbitrary scalar random variables X over a probability space $(\Omega, \mathcal{F}, P)$.*

Sample-Average Approximation *This first method estimates $\mathrm{VaR}_\beta(X)$, $\mathrm{CVaR}_\beta(X)$ for any $\beta \in (0,1)$ and any scalar random variable X. Let $\{x_i\}_{i=1}^N$ be a set of N independently drawn samples of X. The empirical distribution function $\hat{\mathrm{F}}_\mathrm{N}(x)$ for X based on this set of samples is*

$$\hat{\mathrm{F}}_\mathrm{N}(x) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(x \leq x_i),\ \forall\ x_i \in \{x_i\}_{i=1}^N. \tag{32}$$

with $\mathbf{1}$ being the indicator function. The Sample-Average Approximation (SAA) exploits the Dvoretsky-Kiefer-Wolfowitz Inequality [170] built upon by Paul Massart in [171], which proves that the empirical distribution has bounded deviation with respect to the true cumulative distribution function F for X to within some probability $\delta \in (0,1)$:

$$\hat{\mathrm{F}}_\mathrm{N}(x) - \sqrt{\frac{1}{2N}\ln\left(\frac{2}{\delta}\right)} \leq F(x) \tag{33a}$$

$$\leq \hat{\mathrm{F}}_\mathrm{N}(x) + \sqrt{\frac{1}{2N}\ln\left(\frac{2}{\delta}\right)},\ \text{w.p.}\ \geq 1-\delta. \tag{33b}$$

The tail risk $\mathrm{VaR}_\beta(X)$ can be lower and upper bounded for any $\beta \in (0,1)$ using (33). Define upper bound $\overline{\mathrm{VaR}}_\beta(X)$ as:

$$\overline{\mathrm{VaR}_\beta}(X,\delta) =$$

$$\inf\left\{x \in \mathbb{R}\ \middle|\ \hat{\mathrm{F}}_\mathrm{N}(x) - \sqrt{\frac{1}{2N}\ln\left(\frac{2}{\delta}\right)} \geq 1-\beta\right\},$$

and let the lower bound $\underline{\mathrm{VaR}_\beta}(X)$ be defined as:

$$\underline{\mathrm{VaR}_\beta}(X,\delta) =$$

$$\inf\left\{x \in \mathbb{R}\ \middle|\ \hat{\mathrm{F}}_\mathrm{N}(x) + \sqrt{\frac{1}{2N}\ln\left(\frac{2}{\delta}\right)} \geq 1-\beta\right\}.$$

Then, using (33), the following result holds $\forall\ \beta, \delta \in (0,1)$

$$\underline{\mathrm{VaR}_\beta}(X,\delta) \leq \mathrm{VaR}_\beta(X) \leq \overline{\mathrm{VaR}_\beta}(X,\delta)\ \text{w.p.}\ \geq 1-\delta.$$

Note that as the number of samples, N, of the random variable X increases, the gap between the upper and lower bounds shrinks, as the bounds converge to the true value $\mathrm{VaR}_\beta(X)$. Similar methods exist to estimate $\mathrm{CVaR}_\beta(X)$ as well [172].

Scenario Bounds. *The second method upper bounds $\mathrm{VaR}_\beta(X), \mathrm{CVaR}_\beta(X), \mathrm{EVaR}_\beta(X)$ for any $\beta \in (0,1)$. As before, let $\{x_i\}_{i=1}^N$ be a set of N independently drawn samples of the scalar random variable X. Consider the following optimization problem, termed a scenario program [173]:*

$$\begin{aligned} \zeta_N^* = \operatorname*{argmin}_{\zeta \in \mathbb{R}} \quad & \zeta, \\ \text{subject to} \quad & \zeta \geq x_i,\ \forall\ x_i \in \{x_k\}_{k=1}^N. \end{aligned} \tag{34}$$

The theory of scenario optimization states that the solution to this optimization problem is an upper bound on $\mathrm{VaR}_\beta(X)$ with minimum probability $1-(1-\beta)^N$, i.e. if X has probability density function π, then

$$\mathbb{P}_\pi^N\left[\zeta_N^* \geq \mathrm{VaR}_\beta(X)\right] \geq 1-(1-\beta)^N. \tag{35}$$

The above result was proven in [60]. Note that (35) does not need the density function π for X to be known. It just requires an ability to take N independent samples of X. Therefore, if we have a constant $c \in \mathbb{R}$ such that $\mathbb{P}_\pi[x \leq c] = 1$, then we can exploit this inequality (35) to similarly upper bound $\mathrm{CVaR}_\beta(X)$ and $\mathrm{EVaR}_\beta(X)$. Details on this approach can be found in Section 3 of [15].

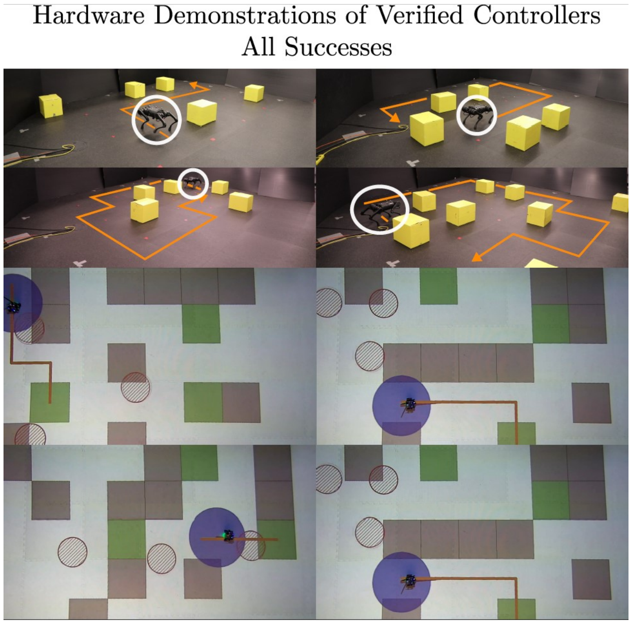

FIGURE 15: Hardware demonstration of controllers verified from a tail-risk perspective. As the implemented controllers were "verified" since the reported lower bound on their Robustness Value-at-Risk was positive, we expect decent behavior in practice. This notion is corroborated by the fact that the verified controllers performed admirably despite randomized test cases generated for each system. Figure adapted from [77].

Open Questions and Future Directions in V&V

Sample Complexity

There exist several open questions in risk-aware verification, though the most notable one concerns the small tail probability requirements for industrial applications. More specifically, for most product-level robotic systems requiring a verification statement, *i.e.* autonomous cars, factory robots, flight software, *etc*, current standards require these systems to be verified to arbitrarily high probabilities, *i.e.* $1-10^{-4}, 1-10^{-6}, 1-10^{-9}$ or even higher. If we assume that the underlying distributions are known, *i.e.* Gaussian as is typically done, then this analysis can be carried out in a tractable, even analytic, fashion. However, if we follow the philosophy underlying the sample-based works that have recently become more popular, as they do not assume underlying distributional knowledge for verification, then verifying systems to these probabilities could require hundreds of thousands of samples or even more. If each of those samples constitutes even one experimental run of the system, then this makes the direct application of these theoretical concepts exceedingly costly or time-intensive. As such, reducing this sample complexity, whether via intelligent test design or by leveraging partial system knowledge, would go a long way to facilitate the widespread industrial adoption of these currently theoretical pipelines.

Compositional Verification

In a similar vein as prior, this second open question stems from a primarily industrial concern as well. Namely, typical large-scale systems are composed of a variety of moving parts, each of which has to satisfy its own component specifications such that the larger, architectured system satisfies a grander objective. Per the prior pipeline, each component could be verified separately in a probabilistic fashion, but in systems with potentially hundreds of separately engineered parts, separate verification procedures could be potentially prohibitive. On the other hand, the system could be verified as a large-scale black-box system, though this could similarly fall under the sample-complexity questions as risen in the prior subsection. As such, determining an optimal way of breaking down these larger-scale, system-level specifications, into easily verifiable subcomponents for their respective systems remains an open problem. Indeed, the satisfiability of a given signal temporal logic specification itself remains a challenging problem. Determining the minimum number of such subcomponents would also mitigate any further sample-complexity issues arising from separate verification procedures as well. On the other hand, perhaps via smart instrumentation, all verification procedures for all subcomponents could be performed simultaneously.

OPEN PROBLEMS AND FUTURE DIRECTIONS

Our discourse up to this point has predominantly centered on the utilization of tail-risk measures in the domains of planning, control, and verification within robotic systems. Nonetheless, it is crucial to emphasize the broader applicability and potential impact of these notions. In this section, we detail several emerging areas that have gained substantial attention.

Risk-Aware Learning

There exists a rich body of literature exploring the integration of tail-risk measures within learning paradigms such as reinforcement learning, supervised learning, and unsupervised learning. These studies delve into diverse topics ranging from risk-sensitive reward functions and policy optimization to risk-aware feature learning and model training. Such research underscores the versatile role of tail-risk measures in not only guiding robotic behavior in uncertain environments but also enabling robots to learn and adapt in a risk-aware manner over time. Thus, to provide a comprehensive overview of the role of tail-risk measures in robotics, it is crucial to shed light on their applications in learning-based contexts as well.

The recent exploration into risk-averse reinforcement learning is well encapsulated by the work of Greenberg et al. [174]. They emphasized the challenges of optimizing risk measures, as conventional methods often overlook high-return strategies. To address this, they proposed a soft risk mechanism coupled with a Cross-Entropy module for efficient risk sampling. This innovative method, while maintaining risk aversion, demonstrated improved risk aversion across diverse benchmarks, setting a precedent for future exploration in this realm. In another interesting direction, Lacotte et al. [175] delved into a risk-sensitive Generative Adversarial Imitation Learning (GAIL) approach aimed to perform as well as or better than the expert regarding a risk profile.

Focusing on risk-constrained reinforcement learning, Chow et al. [22] developed algorithms for risk-constrained MDPs, using chance constraints or CVaR as the risk representation. Their work represents an important step towards understanding and implementing risk constraints in RL and how these can be used for practical applications. Finally, Kose and Ruszczynski's work [176] proposed a novel reinforcement learning methodology employing a Markov coherent dynamic risk measure. This work provided new risk-averse counterparts for basic and multistep methods of temporal differences, paving the way for future exploration in risk-averse learning methodologies.

Despite the notable strides made in these studies, the field of risk-aware reinforcement learning remains relatively under-explored, presenting a wide array of opportunities for further research. Given the complexity of real-world environments and the myriad ways in which risks can be quantified and managed, there is ample scope for the development of innovative, effective risk-aware reinforcement learning strategies. As such, the integration of tail risk measures is a promising direction, likely to yield valuable insights in the years to come.

Risk-Awareness with Nonstationary and Independent Data

This second area has garnered substantial interest insofar as it breaks the assumptions in the works discussed in this survey. Namely, the vast majority of the work discussed has all centered around risk-aware methodologies where either 1) the underlying distribution was known either exactly or in a distributionally robust sense, or 2) the distribution is queryable in a sample-based fashion, where the algorithm receives independent samples. What happens when either of these assumptions fails to hold? This is the central question in a new area of work that is just beginning in the risk-aware space, and for important reasons as well. Consider a robot ambulating over uneven terrain. As the robot traverses the space, any recording of the unevenness of the terrain would correspond to samples from a nonstationary distribution, and if the robot's controller builds a map of the terrain with this information and chooses actions predicated on this map, then successive data is necessarily not independent.

When the underlying distribution of the uncertainty changes during a motion planning task, we would ideally like to understand the level of this shift, so that we can account for it in our risk-aware planners. There has been a push towards identifying out-of-distribution data for learning-based tasks [177, 178]. In [178], the authors study task-driven OOD detection using Probably Approximately Correct (PAC)-Bayes theory for training the robot. The PAC-Bayes procedure provides a performance bound such that violating this bound signals that the robot is operating in an OOD environment.

In addition to *detection* of OOD scenarios, we would also like to *respond* to such scenarios online. Here, distribution-free prediction schemes like those offered by conformal prediction are gaining more traction [179]. As these tools offer ways of generating probabilistically accurate predictors provided streams of, potentially non-independent data, these predictors have been used for motion planning [180–183], confidence regions for learned classifiers [184, 185], and even for risk-aware decision making, though not in a tail-risk sense [186]. If we can identify and adapt to distribution shifts in a risk-aware manner, we can enable robotic systems to react to data drift, unseen data, or spurious correlations [187]. By dynamically adjusting the risk level to adapt to the changing uncertainty distribution and guarantee the desired level of safety for the motion planning task, robots can operate in a wider array of unstructured environments while guaranteeing safety, task completion, and efficiency.

REFERENCES

[1] Yuheng Wang and Margaret P Chapman. Risk-averse autonomous systems: A brief history and recent developments from the perspective of optimal control. *Artificial Intelligence*, page 103743, 2022.

[2] Dimitri P Bertsekas and Ian B Rhodes. On the minimax reachability of target sets and target tubes. *Automatica*, 7(2):233–247, 1971.

[3] Kemin Zhou and John Comstock Doyle. *Essentials of Robust Control*. Prentice Hall, Upper Saddle River, NJ, USA, 1998.

[4] James Blake Rawlings, David Q Mayne, and Moritz M Diehl. *Model Predictive Control: Theory, Computation, and Design*. Nob Hill Publishing, Madison, WI, USA, 2 edition, 2017.

[5] Mo Chen and Claire J Tomlin. Hamilton–Jacobi reachability: Some recent theoretical advances and applications in unmanned airspace management. *Annual Review of Control, Robotics, and Autonomous Systems*, 1:333–358, 2018.

[6] Dimitri P. Bertsekas and Steven E. Shreve. *Stochastic Optimal Control: The Discrete-Time Case*. Athena Scientific, Belmont, MA, USA, 1996.

[7] Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, USA, 2 edition, 2014.

[8] Alessandro Abate, Maria Prandini, John Lygeros, and Shankar Sastry. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, 44(11):2724–2734, 2008.

[9] Sean Summers and John Lygeros. Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem. *Automatica*, 46(12): 1951–1961, 2010.

[10] Ronald A Howard and James E Matheson. Risk-sensitive Markov decision processes. *Management Science*, 18(7):356–369, 1972.

[11] David Jacobson. Optimal stochastic linear systems with exponential performance criteria and their relation to deterministic differential games. *IEEE Transactions on Automatic control*, 18(2):124–131, 1973.

[12] Peter Whittle. Risk-sensitive linear/quadratic/Gaussian control. *Advances in Applied Probability*, 13:764–777, 1981.

[13] Harry Markowitz. Porfolio selection. The Journal of Finance, 7(1):77–91, 1952.

[14] R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.

[15] Prithvi Akella, Anushri Dixit, Mohamadreza Ahmadi, Joel W Burdick, and Aaron D Ames. Sample-based bounds for coherent risk measures: Applications to policy synthesis and verification. *arXiv preprint arXiv:2204.09833*, 2022.

[16] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory*. MPS-SIAM, Philadelphia, PA, USA, 2009.

[17] Wilko Schwarting, Javier Alonso-Mora, and Daniela Rus. Planning and decision-making for autonomous vehicles. *Annual Review of Control, Robotics, and Autonomous Systems*, 1:187–210, 2018.

[18] Erez Karpas and Daniele Magazzeni. Automated planning for robotics. *Annual Review of Control, Robotics, and Autonomous Systems*, 3:417–439, 2020.

[19] Lukas Brunke, Melissa Greeff, Adam W Hall, Zhaocong Yuan, Siqi Zhou, Jacopo Panerati, and Angela P

Schoellig. Safe learning in robotics: From learning-based control to safe reinforcement learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 5:411–444, 2022.

[20] Kerianne L. Hobbs, Mark L. Mote, Matthew C. L. Abate, Samuel D. Coogan, and Eric M. Feron. Run time assurance for safety-critical systems: An introduction to safety filtering approaches for complex control systems. *IEEE Control Systems Magazine*, 2022. to appear.

[21] Anirudha Majumdar and Marco Pavone. How should a robot assess risk? Towards an axiomatic theory of risk in robotics. In *Robotics Research*, pages 75–84. Springer, 2020.

[22] Yinlam Chow, Mohammad Ghavamzadeh, Lucas Janson, and Marco Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *The Journal of Machine Learning Research*, 18(1):6070–6120, 2017.

[23] LA Prashanth, Michael C Fu, et al. Risk-sensitive reinforcement learning via policy gradient search. *Foundations and Trends® in Machine Learning*, 15(5): 537–693, 2022.

[24] Nicole Bäuerle and Ulrich Rieder. More risk-sensitive markov decision processes. *Mathematics of Operations Research*, 39(1):105–120, 2014.

[25] L. Prashanth. Policy gradients for CVaR-constrained MDPs. In *International Conference on Algorithmic Learning Theory*, pages 155–169. Springer, 2014.

[26] Y. Chow and M. Ghavamzadeh. Algorithms for CVaR optimization in MDPs. In *Advances in Neural Information Processing Systems*, pages 3509–3517, 2014.

[27] A. Tamar, Y. Chow, M. Ghavamzadeh, and S. Mannor. Sequential decision making with coherent risk. *IEEE Transactions on Automatic Control*, 62(7):3323–3338, 2016.

[28] A. Tamar, Y. Chow, M. Ghavamzadeh, and S. Mannor. Policy gradient for coherent risk measures. In *Advances in Neural Information Processing Systems*, pages 1468–1476, 2015.

[29] W. B. Haskell and R. Jain. A convex analytic approach to risk-aware Markov decision processes. *SIAM Journal on Control and Optimization*, 53(3):1569–1598, 2015.

[30] Seyedshams Feyzabadi and Stefano Carpin. Risk-aware path planning using hirerachical constrained markov decision processes. In *2014 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 297–303. IEEE, 2014.

[31] Arvind A Pereira, Jonathan Binney, Geoffrey A Hollinger, and Gaurav S Sukhatme. Risk-aware path planning for autonomous underwater vehicles using predictive ocean models. *Journal of Field Robotics*, 30 (5):741–762, 2013.

[32] Thanh Lam, Arun Verma, Bryan Kian Hsiang Low, and Patrick Jaillet. Risk-aware reinforcement learning with coherent risk measures and non-linear function approximation. In *The Eleventh International Conference on Learning Representations*, 2023.

[33] Jia Lin Hau, Erick Delage, Mohammad Ghavamzadeh, and Marek Petrik. On dynamic program decompositions of static risk measures. *arXiv preprint arXiv:2304.12477*, 2023.

[34] Mohamadreza Ahmadi, Ugo Rosolia, Michel D Ingham, Richard M Murray, and Aaron D Ames. Constrained risk-averse markov decision processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 11718–11725, 2021.

[35] S. Carpin, Y. Chow, and M. Pavone. Risk aversion in finite Markov Decision Processes using total cost criteria and average value at risk. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 335–342. IEEE, 2016.

[36] C. Gavriel, G. Hanasusanto, and D. Kuhn. Risk-averse shortest path problems. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 2533–2538. IEEE, 2012.

[37] Mohamadreza Ahmadi, Anushri Dixit, Joel W Burdick, and Aaron D Ames. Risk-averse stochastic shortest path planning. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 5199–5204. IEEE, 2021.

[38] J. Fan and A. Ruszczyński. Risk measurement and risk-averse control of partially observable discrete-time Markov systems. *Mathematical Methods of Operations Research*, 88(2):161–184, 2018.

[39] M. Ahmadi, M. Ono, M. D. Ingham, R. M. Murray, and A. D. Ames. Risk-averse planning under uncertainty. In *2020 American Control Conference (ACC)*, pages 3305–3312. IEEE, 2020.

[40] Mohamadreza Ahmadi, Ugo Rosolia, Michel D Ingham, Richard M Murray, and Aaron D Ames. Risk-averse decision making under uncertainty. *IEEE Transactions on Automatic Control*, 2023.

[41] S. Singh, Y. Chow, A. Majumdar, and M. Pavone. A framework for time-consistent, risk-sensitive model predictive control: Theory and algorithms. *IEEE Transactions on Automatic Control*, 2018.

[42] Anushri Dixit, Mohamadreza Ahmadi, and Joel W. Burdick. Distributionally robust model predictive control with total variation distance. *IEEE Control Systems Letters*, 6:3325–3330, 2022. doi: 10.1109/LCSYS.2022.3184921.

[43] Bart P. G. Van Parys, Daniel Kuhn, Paul J. Goulart, and Manfred Morari. Distributionally robust control of constrained stochastic systems. *IEEE Transactions on Automatic Control*, 61(2):430–442, 2016. doi: 10.

1109/TAC.2015.2444134.

[44] Yuxiao Chen, Ugo Rosolia, Wyatt Ubellacker, Noel Csomay-Shanklin, and Aaron D Ames. Interactive multi-modal motion planning with branch model predictive control. *IEEE Robotics and Automation Letters*, 7(2):5365–5372, 2022.

[45] P. Sopasakis, M. Schuurmans, and P. Patrinos. Risk-averse risk-constrained optimal control. In *2019 18th European Control Conference (ECC)*, pages 375–380, 2019. doi: 10.23919/ECC.2019.8796021.

[46] Jeremy Coulson, John Lygeros, and Florian Dörfler. Data-enabled predictive control: In the shallows of the DeePC. In *2019 18th European Control Conference (ECC)*, pages 307–312. IEEE, 2019.

[47] Jeremy Coulson, John Lygeros, and Florian Dörfler. Distributionally robust chance constrained data-enabled predictive control. *IEEE Transactions on Automatic Control*, 67(7):3289–3304, 2022. doi: 10.1109/TAC.2021.3097706.

[48] Alireza Zolanvari and Ashish Cherukuri. Data-driven distributionally robust iterative risk-constrained model predictive control. In *2022 European Control Conference (ECC)*, pages 1578–1583, 2022. doi: 10.23919/ECC55457.2022.9838319.

[49] Astghik Hakobyan, Gyeong Chan Kim, and Insoon Yang. Risk-aware motion planning and control using CVaR-constrained optimization. *IEEE Robotics and Automation Letters*, 4(4):3924–3931, 2019.

[50] Anushri Dixit, Mohamadreza Ahmadi, and Joel W. Burdick. Risk-sensitive motion planning using entropic value-at-risk. In *European Control Conference*, 2021.

[51] Anushri Dixit, Mohamadreza Ahmadi, and Joel W Burdick. Risk-averse receding horizon motion planning for obstacle avoidance using coherent risk measures. *Artificial Intelligence*, 325:104018, 2023.

[52] Anushri Dixit, David D Fan, Kyohei Otsu, Sharmita Dey, Ali-Akbar Agha-Mohammadi, and Joel W Burdick. Step: Stochastic traversability evaluation and planning for risk-aware off-road navigation; results from the darpa subterranean challenge. *Field Robotics*, 4:182–210, 2024. doi: 10.55417/fr.2024006.

[53] David D Fan, Kyohei Otsu, Yuki Kubo, Anushri Dixit, Joel Burdick, and Ali-Akbar Agha-Mohammadi. Step: Stochastic traversability evaluation and planning for risk-aware off-road navigation. In *Robotics: Science and Systems*, pages 1–21. RSS Foundation, 2021.

[54] Samantha Samuelson and Insoon Yang. Safety-aware optimal control of stochastic systems using conditional value-at-risk. In *2018 Annual American Control Conference (ACC)*, pages 6285–6290, 2018. doi: 10.23919/ACC.2018.8430957.

[55] Margaret P Chapman, Jonathan Lacotte, Aviv Tamar, Donggun Lee, Kevin M Smith, Victoria Cheng, Jaime F Fisac, Susmit Jha, Marco Pavone, and Claire J Tomlin. A risk-sensitive finite-time reachability approach for safety of stochastic dynamic systems. In *2019 American Control Conference (ACC)*, pages 2958–2963. IEEE, 2019.

[56] Margaret P Chapman, Riccardo Bonalli, Kevin M Smith, Insoon Yang, Marco Pavone, and Claire J Tomlin. Risk-sensitive safety analysis using conditional value-at-risk. *IEEE Transactions on Automatic Control*, 67(12):6521–6536, 2022.

[57] Margaret P Chapman, Michael Fauß, and Kevin M Smith. On optimizing the conditional value-at-risk of a maximum cost for risk-averse safety analysis. *IEEE Transactions on Automatic Control*, in press, doi: 10.1109/TAC.2022.3195381, 2022. doi: 10.1109/TAC.2022.3195381.

[58] Chuanning Wei, Michael Fauß, and Margaret P. Chapman. CVaR-based safety analysis in the infinite time horizon setting. In *2022 American Control Conference (ACC)*, pages 2863–2870, 2022. doi: 10.23919/ACC53348.2022.9867285.

[59] Andrew Singletary, Mohamadreza Ahmadi, and Aaron D Ames. Safe control for nonlinear systems with stochastic uncertainty via risk control barrier functions. *IEEE Control Systems Letters*, 7:349–354, 2022.

[60] Prithvi Akella, Mohamadreza Ahmadi, and Aaron D Ames. A scenario approach to risk-aware safety-critical system verification. *arXiv preprint arXiv:2203.02595*, 2022.

[61] Lars Lindemann, Nikolai Matni, and George J Pappas. Stl robustness risk over discrete-time stochastic processes. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 1329–1335. IEEE, 2021.

[62] Murat Cubuktepe and Ufuk Topcu. Verification of markov decision processes with risk-sensitive measures. In *2018 Annual American Control Conference (ACC)*, pages 2371–2377. IEEE, 2018.

[63] Karen Leung and Marco Pavone. Semi-supervised trajectory-feedback controller synthesis for signal temporal logic specifications. *arXiv preprint arXiv:2202.01997*, 2022.

[64] Tobias Meggendorfer. *Verification of Discrete-Time Markov Decision Processes*. PhD thesis, Technische Universität München, 2021.

[65] Samantha Samuelson and Insoon Yang. Safety-aware optimal control of stochastic systems using conditional value-at-risk. In *2018 Annual American Control Conference (ACC)*, pages 6285–6290. IEEE, 2018.

[66] Navid Hashemi, Xin Qin, Jyotirmoy V Deshmukh, Georgios Fainekos, Bardh Hoxha, Danil Prokhorov, and Tomoya Yamaguchi. Risk-awareness in learning neural controllers for temporal logic objectives. *arXiv*

preprint arXiv:2210.07439, 2022.

[67] Domenico Quagliarella and Emiliano Iuliano. Robust design of a supersonic natural laminar flow wing-body. *IEEE Computational Intelligence Magazine*, 12 (4):14–27, 2017.

[68] Lars Lindemann, Lejun Jiang, Nikolai Matni, and George J Pappas. Risk of stochastic systems for temporal logic specifications. *ACM Transactions on Embedded Computing Systems*, 22(3):1–31, 2023.

[69] Lars Lindemann, Alena Rodionova, and George Pappas. Temporal robustness of stochastic signals. In *25th ACM International Conference on Hybrid Systems: Computation and Control*, pages 1–11, 2022.

[70] Marc Rigter, Bruno Lacerda, and Nick Hawes. Risk-averse bayes-adaptive reinforcement learning. *Advances in Neural Information Processing Systems*, 34: 1142–1154, 2021.

[71] Mathieu Godbout, Maxime Heuillet, Sharath Chandra, Rupali Bhati, and Audrey Durand. Carl: Conditional-value-at-risk adversarial reinforcement learning. *arXiv preprint arXiv:2109.09470*, 2021.

[72] L Jeff Hong and Guangwu Liu. Monte carlo estimation of value-at-risk, conditional value-at-risk and their sensitivities. In *Proceedings of the 2011 Winter Simulation Conference (WSC)*, pages 95–107. IEEE, 2011.

[73] Ji-Hui Kim, Jaehee Lee, and Sung-Kwan Joo. Conditional value-at-risk-based method for evaluating the economic risk of superconducting fault current limiter installation. *IEEE Transactions on Applied Superconductivity*, 25(3):1–4, 2015.

[74] Dohyeong Kim and Songhwai Oh. Efficient off-policy safe reinforcement learning using trust region conditional value at risk. *IEEE Robotics and Automation Letters*, 7(3):7644–7651, 2022.

[75] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Proceedings of the Conference on Robot Learning*, pages 1–16, Mountain View, California, November 2017.

[76] Prithvi Akella and Aaron D Ames. A barrier-based scenario approach to verifying safety-critical systems. *IEEE Robotics and Automation Letters*, 7(4): 11062–11069, 2022.

[77] Prithvi Akella, Wyatt Ubellacker, and Aaron D. Ames. Safety-critical controller verification via sim2real gap quantification. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10539–10545, 2023. doi: 10.1109/ICRA48891.2023.10161126.

[78] Fangda Liu and Ruodu Wang. A theory for measures of tail risk. *Math. Oper. Res.*, 46(3):1109–1128, aug 2021. ISSN 0364-765X. doi: 10.1287/moor.2020.1072.

[79] P. Artzner, F. Delbaen, Jean-Marc Eber, and D. Heath. Coherent measures of risk. *Mathematical finance*, 9(3): 203–228, 1999.

[80] A. Ahmadi-Javid and A. Pichler. An analytical study of norms and banach spaces induced by the entropic value-at-risk. *Mathematics and Financial Economics*, 11 (4):527–550, 2017.

[81] A. Ahmadi-Javid. Entropic value-at-risk: A new coherent risk measure. *Journal of Optimization Theory and Applications*, 155(3):1105–1123, 2012.

[82] Alexander Shapiro. Distributionally robust stochastic programming. *SIAM Journal on Optimization*, 27(4): 2258–2275, 2017.

[83] A. Ruszczyński. Risk-averse dynamic programming for Markov decision processes. *Mathematical programming*, 125(2):235–261, 2010.

[84] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on stochastic programming: modeling and theory*. SIAM, 2014.

[85] Peter Whittle. Risk-sensitive linear/quadratic/Gaussian control. *Advances in Applied Probability*, 13(4): 764–777, 1981.

[86] *Games and Decisions*. John F. Wiley & Sons, Inc., New York, NY, USA, 1957.

[87] Peter Whittle. *Risk-sensitive Optimal Control*. Wiley, New York, NY, USA, 1990.

[88] Naci Saldi, Tamer Başar, and Maxim Raginsky. Approximate Markov-Nash equilibria for discrete-time risk-sensitive mean-field games. *Mathematics of Operations Research*, 45(4):1596–1620, 2020.

[89] Rubén Blancas-Rivera, Rolando Cavazos-Cadena, and Hugo Cruz-Suárez. Discounted approximations in risk-sensitive average markov cost chains with finite state space. *Mathematical Methods of Operations Research*, 91:241–268, 2020.

[90] Margaret P Chapman and Kevin M Smith. Classical risk-averse control for a finite-horizon Borel model. *IEEE Control Systems Letters*, 6:1525–1530, 2021.

[91] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczynski. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.

[92] V. Krishnamurthy. *Partially observed Markov decision processes*. Cambridge University Press, 2016.

[93] M. Ahmadi, N. Jansen, B. Wu, and U. Topcu. Control theory meets POMDPs: A hybrid systems approach. *IEEE Transactions on Automatic Control*, 2020.

[94] Alberto Bemporad and Manfred Morari. Robust model predictive control: A survey. In *Robustness in identification and control*, pages 207–226. Springer, 1999.

[95] Ali Mesbah. Stochastic model predictive control: An overview and perspectives for future research. *IEEE Control Systems Magazine*, 36(6):30–44, 2016. doi: 10.1109/MCS.2016.2602087.

[96] Jenna Reher and Aaron D Ames. Dynamic walking:

Toward agile and efficient bipedal robots. *Annual Reviews*, 2020.

[97] Tomáš Rouček, Martin Pecka, Petr Čížek, Tomáš Petříček, Jan Bayer, Vojtěch Šalanskỳ, Daniel Heřt, Matěj Petrlík, Tomáš Báča, Vojěch Spurnỳ, et al. Darpa subterranean challenge: Multi-robotic exploration of underground environments. In *International Conference on Modelling and Simulation for Autonomous Systesm*, pages 274–290. Springer, 2019.

[98] Franco Blanchini and Stefano Miani. *Set-theoretic methods in control*, volume 78. Springer, 2008.

[99] Matthias Althoff, Olaf Stursberg, and Martin Buss. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In *2008 47th IEEE Conference on Decision and Control*, pages 4042–4048. IEEE, 2008.

[100] Ian M Mitchell, Alexandre M Bayen, and Claire J Tomlin. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on automatic control*, 50(7):947–957, 2005.

[101] Mitio Nagumo. Über die lage der integralkurven gewöhnlicher differentialgleichungen. *Proceedings of the Physico-Mathematical Society of Japan. 3rd Series*, 24: 551–559, 1942.

[102] Jean-Pierre Aubin, Alexandre M Bayen, and Patrick Saint-Pierre. *Viability theory: new directions*. Springer Science & Business Media, 2011.

[103] Stephen Prajna, Ali Jadbabaie, and George J Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8):1415–1428, 2007.

[104] Amir Ali Ahmadi and Anirudha Majumdar. Dsos and sdsos optimization: Lp and socp-based alternatives to sum of squares optimization. In *2014 48th annual conference on information sciences and systems (CISS)*, pages 1–5. IEEE, 2014.

[105] Aaron D Ames, Xiangru Xu, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2016.

[106] Quan Nguyen, Ayonga Hereid, Jessy W Grizzle, Aaron D Ames, and Koushil Sreenath. 3d dynamic walking on stepping stones with control barrier functions. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 827–834. IEEE, 2016.

[107] Yuxiao Chen, Ayonga Hereid, Huei Peng, and Jessy Grizzle. Enhancing the performance of a safe controller via supervised learning for truck lateral control. *Journal of Dynamic Systems, Measurement, and Control*, 141(10):101005, 2019.

[108] Xiangru Xu, Paulo Tabuada, Jessy W Grizzle, and Aaron D Ames. Robustness of control barrier functions for safety critical control. *IFAC-PapersOnLine*, 48(27):54–61, 2015.

[109] Shishir Kolathaya and Aaron D Ames. Input-to-state safety with control barrier functions. *IEEE control systems letters*, 3(1):108–113, 2018.

[110] Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European Control Conference (ECC)*, pages 3420–3431. IEEE, 2019.

[111] Mohamadreza Ahmadi, Andrew Singletary, Joel W Burdick, and Aaron D Ames. Safe policy synthesis in multi-agent pomdps via discrete-time barrier functions. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 4797–4803. IEEE, 2019.

[112] Ayush Agrawal and Koushil Sreenath. Discrete Control Barrier Functions for Safety-Critical Control of Discrete Systems with Application to Bipedal Robot Navigation. In *Robotics: Science and Systems*, 2017.

[113] Mohamadreza Ahmadi, Andrew Singletary, Joel W Burdick, and Aaron D Ames. Barrier functions for multiagent-pomdps with dtl specifications. In *The 59th IEEE Conference on Decision and Control*, 2020.

[114] Andrew Clark. Control barrier functions for complete and incomplete information stochastic systems. In *2019 American Control Conference (ACC)*, pages 2928–2935. IEEE, 2019.

[115] Cesar Santoyo, Maxence Dutreix, and Samuel Coogan. A barrier function approach to finite-time stochastic system verification and control. *arXiv preprint arXiv:1909.05109*, 2019.

[116] Shakiba Yaghoubi, Keyvan Majd, Georgios Fainekos, Tomoya Yamaguchi, Danil Prokhorov, and Bardh Hoxha. Risk-bounded control using stochastic barrier functions. *IEEE Control Systems Letters*, 5(5):1831–1836, 2020.

[117] Shakiba Yaghoubi, Georgios Fainekos, Tomoya Yamaguchi, Danil Prokhorov, and Bardh Hoxha. Risk-bounded control with kalman filtering and stochastic barrier functions. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 5213–5219. IEEE, 2021.

[118] Matti Vahs, Christian Pek, and Jana Tumova. Belief control barrier functions for risk-aware control. *IEEE Robotics and Automation Letters*, 2023.

[119] Andrew W Singletary, Aaron D Ames, and Mohamadreza Ahmadi. Controlling a moveable device utilizing risk control barrier functions, November 30 2023. US Patent App. 18/324,718.

[120] Matti Vahs and Jana Tumova. Risk-aware control for robots with non-gaussian belief spaces. *arXiv preprint arXiv:2309.12857*, 2023.

[121] Mohamadreza Ahmadi, Xiaobin Xiong, and Aaron D Ames. Risk-averse control via CVaR barrier func-

tions: Application to bipedal robot locomotion. *IEEE Control Systems Letters*, 6:878–883, 2021.

[122] Jessy W Grizzle, Christine Chevallereau, Ryan W Sinnet, and Aaron D Ames. Models, feedback control, and open problems of 3D bipedal robotic walking. *Automatica*, 50(8):1955–1988, 2014.

[123] X. Xiong and A. D. Ames. Dynamic and versatile humanoid walking via embedding 3D actuated SLIP model with hybrid LIP based stepping. *IEEE Robotics and Automation Letters*, 5(4):6286–6293, 2020. doi: 10.1109/LRA.2020.3013924.

[124] Xiaobin Xiong, Jenna Reher, and Aaron Ames. Global position control on underactuated bipedal robots: Step-to-step dynamics approximation for step planning. *To appear in 2021 IEEE/RSJ International Conference on Robotics and Automation (ICRA), arXiv:2011.06050*.

[125] Jérémie Guiochet, Mathilde Machin, and Hélène Waeselynck. Safety-critical advanced robots: A survey. *Robotics and Autonomous Systems*, 94:43–52, 2017.

[126] Junbeom Yoo, Eunkyoung Jee, and Sungdeok Cha. Formal modeling and verification of safety-critical software. *IEEE software*, 26(3):42–49, 2009.

[127] Adina Aniculaesei, Daniel Arnsberger, Falk Howar, and Andreas Rausch. Towards the verification of safety-critical autonomous systems in dynamic environments. *arXiv preprint arXiv:1612.04977*, 2016.

[128] Tomás Grimm, Djones Lettnin, and Michael Hübner. A survey on formal verification techniques for safety-critical systems-on-chip. *Electronics*, 7(6):81, 2018.

[129] Federico Vicentini, Mehrnoosh Askarpour, Matteo G Rossi, and Dino Mandrioli. Safety assessment of collaborative robotics through automated formal verification. *IEEE Transactions on Robotics*, 36(1):42–61, 2019.

[130] Xu Cheng, Lizhi Zhou, Wentao Liu, Yijian Li, Mou Peng, and Yinhuai Wang. Construction and verification of risk predicting models to evaluate the possibility of venous thromboembolism after robot-assisted radical prostatectomy. *Annals of Surgical Oncology*, pages 1–10, 2022.

[131] Mehrnoosh Askarpour, Dino Mandrioli, Matteo Rossi, and Federico Vicentini. Safer-hrc: Safety analysis through formal verification in human-robot collaboration. In *International Conference on Computer Safety, Reliability, and Security*, pages 283–295. Springer, 2016.

[132] Tom P Huck, Nadine Münch, Luisa Hornung, Christoph Ledermann, and Christian Wurll. Risk assessment tools for industrial human-robot collaboration: Novel approaches and practical needs. *Safety Science*, 141:105288, 2021.

[133] International Organization for Standardization (ISO). Iso/tr 14121-2:2012 safety of machinery - risk assessment - part 2: Practical guidance and examples of methods, . URL https://www.iso.org/obp/ui/#iso:std:iso:tr:14121:-2:ed-2:v1:en.

[134] International Organization for Standardization (ISO). Iso 12100:2010 safety of machinery - general principles for design - risk assessment and risk reduction, . URL https://www.iso.org/obp/ui/#iso:std:iso:12100:ed-1:v1:en.

[135] Ming-Yuan Yu, Ram Vasudevan, and Matthew Johnson-Roberson. Occlusion-aware risk assessment for autonomous driving in urban environments. *IEEE Robotics and Automation Letters*, 4(2):2235–2241, 2019.

[136] Mark Strickland, Georgios Fainekos, and Heni Ben Amor. Deep predictive models for collision risk assessment in autonomous driving. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4685–4692. IEEE, 2018.

[137] Cian Ryan, Finbarr Murphy, and Martin Mullins. End-to-end autonomous driving risk analysis: A behavioural anomaly detection approach. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1650–1662, 2020.

[138] Stéphanie Lefèvre, Dizan Vasquez, and Christian Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *ROBOMECH journal*, 1(1):1–14, 2014.

[139] BCBS. Fundamental review of the trading book: A revised market risk framework, 2013.

[140] Edmund M Clarke and Jeannette M Wing. Formal methods: State of the art and future directions. *ACM Computing Surveys (CSUR)*, 28(4):626–643, 1996.

[141] Jim Woodcock, Peter Gorm Larsen, Juan Bicarregui, and John Fitzgerald. Formal methods: Practice and experience. *ACM computing surveys (CSUR)*, 41(4):1–36, 2009.

[142] https://shemesh.larc.nasa.gov/fm/fm-what.html.

[143] Amir Pnueli. The temporal logic of programs. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science*, pages 46–57, Washington, DC, October 1977.

[144] Christel Baier, Boudewijn Haverkort, Holger Hermanns, and Joost-Pieter Katoen. Model checking continuous-time markov chains by transient analysis. In *International Conference on Computer Aided Verification*, pages 358–372. Springer, 2000.

[145] Edmund M Clarke. Model checking. In *International Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 54–56. Springer, 1997.

[146] Ron Koymans. Specifying real-time properties with metric temporal logic. *Real-time systems*, 2(4):255–299, 1990.

[147] Oded Maler and Dejan Nickovic. Monitoring temporal properties of continuous signals. In *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*, pages 152–166. Springer, 2004.

[148] Yasser Shoukry, Pierluigi Nuzzo, Alberto L Sangiovanni-Vincentelli, Sanjit A Seshia, George J Pappas, and Paulo Tabuada. Smc: Satisfiability modulo convex optimization. In *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control*, pages 19–28, 2017.

[149] Mary Sheeran, Satnam Singh, and Gunnar Stålmarck. Checking safety properties using induction and a sat-solver. In *International conference on formal methods in computer-aided design*, pages 127–144. Springer, 2000.

[150] Andrea Bianco and Luca de Alfaro. Model checking of probabilistic and nondeterministic systems. In *International Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 499–513. Springer, 1995.

[151] Hans Hansson and Bengt Jonsson. A logic for reasoning about time and reliability. *Formal aspects of computing*, 6(5):512–535, 1994.

[152] Marta Kwiatkowska, Gethin Norman, and David Parker. Prism 4.0: Verification of probabilistic real-time systems. In *International conference on computer aided verification*, pages 585–591. Springer, 2011.

[153] Håkan LS Younes and Reid G Simmons. Probabilistic verification of discrete event systems using acceptance sampling. In *International Conference on Computer Aided Verification*, pages 223–235. Springer, 2002.

[154] Håkan LS Younes and Reid G Simmons. Statistical probabilistic model checking with a focus on time-bounded properties. *Information and Computation*, 204 (9):1368–1409, 2006.

[155] Axel Legay, Anna Lukina, Louis Marie Traonouez, Junxing Yang, Scott A Smolka, and Radu Grosu. Statistical model checking. In *Computing and Software Science*, pages 478–504. Springer, 2019.

[156] Axel Legay, Benoît Delahaye, and Saddek Bensalem. Statistical model checking: An overview. In *International conference on runtime verification*, pages 122–135. Springer, 2010.

[157] Ram P Kanwal. *Generalized functions theory and technique: Theory and technique*. Springer Science & Business Media, 1998.

[158] Prithvi Akella, Mohamadreza Ahmadi, Richard M Murray, and Aaron D Ames. Formal test synthesis for safety-critical autonomous systems based on control barrier functions. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 790–795. IEEE, 2020.

[159] Prithvi Akella, Mohamadreza Ahmadi, Richard M Murray, and Aaron D Ames. Barrier-based test synthesis for safety-critical systems subject to timed reach-avoid specifications. *arXiv preprint arXiv:2301.09622*, 2023.

[160] Jin Zhang and Jingyue Li. Testing and verification of neural-network-based safety-critical control software: A systematic literature review. *Information and Software Technology*, 123:106296, 2020.

[161] Shromona Ghosh, Felix Berkenkamp, Gireeja Ranade, Shaz Qadeer, and Ashish Kapoor. Verifying controllers against adversarial examples with bayesian optimization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7306–7313. IEEE, 2018.

[162] Marta Kwiatkowska, Gethin Norman, and David Parker. Stochastic model checking. In *International School on Formal Methods for the Design of Computer, Communication and Software Systems*, pages 220–270. Springer, 2007.

[163] Koushik Sen, Mahesh Viswanathan, and Gul Agha. On statistical model checking of stochastic systems. In *International Conference on Computer Aided Verification*, pages 266–280. Springer, 2005.

[164] Gul Agha and Karl Palmskog. A survey of statistical model checking. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 28(1):1–39, 2018.

[165] Frank Ciesinski and Marcus Größer. On probabilistic computation tree logic. In *Validation of Stochastic Systems*, pages 147–188. Springer, 2004.

[166] Peter Wakker and Amos Tversky. An axiomatization of cumulative prospect theory. *Journal of risk and uncertainty*, 7(2):147–175, 1993.

[167] Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 661–668, Sardinia, Italy, May 2010.

[168] Lars Lindemann, Alexander Robey, Lejun Jiang, Stephen Tu, and Nikolai Matni. Learning robust output control barrier functions from safe expert demonstrations. *arXiv preprint arXiv:2111.09971*, 2021.

[169] Sean Wilson, Paul Glotfelter, Li Wang, Siddharth Mayya, Gennaro Notomista, Mark Mote, and Magnus Egerstedt. The robotarium: Globally impactful opportunities, challenges, and lessons learned in remote-access, distributed control of multirobot systems. *IEEE Control Systems Magazine*, 40(1):26–44, 2020.

[170] A. Dvoretzky, J. Kiefer, and J. Wolfowitz. Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator. *The Annals of Mathematical Statistics*, 27(3):642–669, 1956. ISSN 00034851.

[171] P. Massart. The Tight Constant in the Dvoretzky-

Kiefer-Wolfowitz Inequality. *The Annals of Probability*, 18(3):1269 – 1283, 1990. doi: 10.1214/aop/1176990746.

[172] Philip Thomas and Erik Learned-Miller. Concentration inequalities for conditional value at risk. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6225–6233. PMLR, 09–15 Jun 2019.

[173] M. C. Campi and S. Garatti. The exact feasibility of randomized solutions of uncertain convex programs. *SIAM Journal on Optimization*, 19(3):1211–1230, 2008. doi: 10.1137/07069821X.

[174] Ido Greenberg, Yinlam Chow, Mohammad Ghavamzadeh, and Shie Mannor. Efficient risk-averse reinforcement learning. *arXiv preprint arXiv:2205.05138*, 2022.

[175] Jonathan Lacotte, Mohammad Ghavamzadeh, Yinlam Chow, and Marco Pavone. Risk-sensitive generative adversarial imitation learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2154–2163. PMLR, 2019.

[176] Ümit Köse and Andrzej Ruszczyński. Risk-averse learning by temporal difference methods with Markov risk measures. *Journal of Machine Learning Research*, 22:1–34, 2021.

[177] Rohan Sinha, Apoorva Sharma, Somrita Banerjee, Thomas Lew, Rachel Luo, Spencer M. Richards, Yixiao Sun, Edward Schmerling, and Marco Pavone. A system-level view on out-of-distribution data in robotics, 2022.

[178] Alec Farid, Sushant Veer, and Anirudha Majumdar. Task-driven out-of-distribution detection with statistical guarantees for robot learning. In Aleksandra Faust, David Hsu, and Gerhard Neumann, editors, *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pages 970–980. PMLR, 08–11 Nov 2022.

[179] Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

[180] Anushri Dixit, Lars Lindemann, Skylar X Wei, Matthew Cleaveland, George J Pappas, and Joel W Burdick. Adaptive conformal prediction for motion planning among dynamic agents. In *Learning for Dynamics and Control Conference*, pages 300–314. PMLR, 2023.

[181] Lars Lindemann, Matthew Cleaveland, Gihyun Shim, and George J Pappas. Safe planning in dynamic environments using conformal prediction. *arXiv preprint arXiv:2210.10254*, 2022.

[182] Rachel Luo, Shengjia Zhao, Jonathan Kuck, Boris Ivanovic, Silvio Savarese, Edward Schmerling, and Marco Pavone. Sample-efficient safety assurances using conformal prediction. In *Algorithmic Foundations of Robotics XV: Proceedings of the Fifteenth Workshop on the Algorithmic Foundations of Robotics*, pages 149–169. Springer, 2022.

[183] Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners, 2023.

[184] Janette Vazquez and Julio C Facelli. Conformal prediction in clinical medical sciences. *Journal of Healthcare Informatics Research*, 6(3):241–252, 2022.

[185] Harris Papadopoulos, Alex Gammerman, and Volodya Vovk. Reliable diagnosis of acute abdominal pain with conformal prediction. *Engineering Intelligent Systems*, 17(2):127, 2009.

[186] Shreyas Ramakrishna, Baiting Luo, Yogesh Barve, Gabor Karsai, and Abhishek Dubey. Risk-aware scene sampling for dynamic assurance of autonomous systems. In *2022 IEEE International Conference on Assured Autonomy (ICAA)*, pages 107–116. IEEE, 2022.

[187] Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre Alvise-Rebuffi, Ira Ktena, Krishnamurthy Dvijotham, and Taylan Cemgil. A fine-grained analysis on distribution shift, 2021.

AUTHOR BIOGRAPHIES

Prithvi Akella received the B.S. degree in Mechanical Engineering from the University of California, Berkeley, in 2018, and the M.S. and Ph.D. degrees in Mechanical Engineering from California Institute of Technology in 2020 and 2023, respectively. He is the recipient of the Bell Family Graduate Fellowship in Engineering and Applied Sciences. His current research focuses on the automated test and evaluation of cyber-physical systems.

Anushri Dixit is a Postdoctoral Researcher in the Department of Mechanical & Aerospace Engineering at Princeton University. She received her Ph.D. in Control and Dynamical Systems from California Institute of Technology in 2023 and her B.S. in Electrical Engineering from Georgia Institute of Technology in 2017. Her research focuses on motion planning and control of robots in unstructured environments while accounting for uncertainty in a principled manner. She has received the Outstanding Student Paper Award at the Conference on Decision and Control, Best Student Paper Award at the Conference of Robot Learning (as a co-author), and was selected as a Rising Star in Data Science by The University of Chicago.

Mohamadreza Ahmadi is currently a technical lead in planning at Gatik AI, Mountain View, CA. He finished his DPhil (PhD) in Engineering Science in November, 2016 at the University of Oxford, UK, as a Clarendon Scholar. His PhD was followed by research positions at the University of Texas, Austin, TX, the Center for Autonomous Systems and Technologies (CAST) at the California Institute of Technology, Pasadena, CA, NASA Jet Propulsion Laboratory, La Canada, CA, and TuSimple, La Jolla, CA. He is the recipient of the Sloan-Robinson Engineering Fellowship, an Edgell-Sheppee Award, and an ICES Postdoctoral Fellowship. His current research is on planning and control under uncertainty with application to autonomous driving, in particular, short-haul to long-haul trucks.

Lars Lindemann is currently an Assistant Professor in the Thomas Lord Department of Computer Science at the University of Southern California where he is also a member of the Ming Hsieh Department of Electrical and Computer Engineering (by courtesy), the Robotics and Autonomous Systems Center, and the Center for Autonomy and Artificial Intelligence. Between 2020 and 2022, he was a Postdoctoral Fellow in the Department of Electrical and Systems Engineering at the University of Pennsylvania. He received the Ph.D. degree in Electrical Engineering from KTH Royal Institute of Technology in 2020. His research interests include systems and control theory, formal methods, and autonomous systems. Professor Lindemann received the Outstanding Student Paper Award at the 58th IEEE Conference on Decision and Control and the Student Best Paper Award (as a co-advisor) at the 60th IEEE Conference on Decision and Control. He was finalist for the Best Paper Award at the 2022 Conference on Hybrid Systems: Computation and Control and for the Best Student Paper Award at the 2018 American Control Conference.

Margaret P. Chapman is an Assistant Professor with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering at the University of Toronto. Her research focuses on risk-aware control theory, and she also investigates different control-theoretic applications, including leukemia treatment. Margaret earned her Ph.D. degree in Electrical Engineering and Computer Sciences (EECS) from the University of California Berkeley (UC Berkeley) in May 2020. In 2021, Margaret received the Leon O. Chua Award for outstanding achievement in nonlinear science from EECS at UC Berkeley, and in 2023, Margaret received the Connaught New Researcher Award from the Office of the Vice-President, Research and Innovation at the University of Toronto.

George J. Pappas received the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, Berkeley, CA, USA, in 1998. He is currently the Joseph Moore Professor in and the chair of the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104, USA. He also holds a secondary appointment with the Department of Computer and Information Sciences and the Department of Mechanical Engineering and Applied Mechanics. He is a member of the General Robotics, Automation, Sensing, and Perception Lab and the Penn Research in Embedded Computing and Integrated Systems Engineering Center. He was previously the deputy dean for research with the School of Engineering and Applied Science. His research interests include control theory and, in particular, hybrid systems, embedded systems, cyberphysical systems, and hierarchical and distributed control systems, with applications to unmanned aerial vehicles, distributed robotics, green buildings, and biomolecular networks. He was a recipient of various awards, such as the Antonio Ruberti Young Researcher Prize, the IEEE Control Systems Society George S. Axelby Award, the O. Hugo Schuck Best Paper Award, the George H. Heilmeier Award, the National Science Founda- tion Presidential Early Career Award for Scientists and Engineers, and numerous best student papers awards. He is a Fellow of IEEE.

Aaron D. Ames received the B.S. degree in mechanical engineering and the B.A. degree in mathematics from the University of St. Thomas, Saint Paul, MN, USA in 2001, and the M.A. degree in mathematics and the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, CA, USA, in 2006. From 2006 to 2008, he served as a PostDoctoral Scholar in control and dynamical systems with the California Institute of Technology (Caltech), Pasadena, CA, USA. In 2008, he began his faculty career at Texas A&M University, College Station, TX, USA. He was an Associate Professor with the Woodruff School of Mechanical Engineering and the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA. Since 2017, he has been a Bren Professor of Mechanical and Civil Engineering and Control and Dynamical Systems at Caltech. His research interests include the areas of robotics, nonlinear, safety-critical control, and hybrid systems, with a special focus on applications to dynamic robots—both formally and through experimental validation. Dr. Ames was a recipient of the 2005 Leon O. Chua Award for Achievement in Nonlinear Science and the 2006 Bernard Friedman Memorial Prize in Applied Mathematics from the University of California, Berkeley. He received the NSF CAREER award in 2010, the 2015 Donald P. Eckman Award, and the 2019 IEEE CSS Antonio Ruberti Young Researcher Prize.

Joel W. Burdick , the Richard L. and Dorothy M. Hayman Professor of Mechanical Engineering and Bioengineering, received his undergraduate degrees in mechanical engineering and chemistry from Duke University and M.S. and Ph.D. degrees in mechanical engineering from Stanford University. He has been with the department of mechanical engineering at the Caltech since May 1988, where he has been the recipient of the NSF Presidential Young Investigator award, the Office of Naval Research Young Investigator award, and the Feynman fellowship. He has also received the ASCIT Award for Excellence in Undergraduate Teaching and the GSA Award for Excellence in Graduate Student Education, and received the Popular Mechanics Breakthrough Award in 2011. In addition to mechanical engineering, he is a core faculty of the control and dynamical systems option, as well as a faculty affiliate in the options of bioengineering (BE) and computational and neural systems (CNS). His research interests lie mainly in the areas of robotics, kinematics, and mechanical systems.