



HAL
open science

Risk-Averse Trajectory Optimization via Sample Average Approximation

Thomas Lew, Riccardo Bonalli, Marco Pavone

► **To cite this version:**

Thomas Lew, Riccardo Bonalli, Marco Pavone. Risk-Averse Trajectory Optimization via Sample Average Approximation. IEEE Robotics and Automation Letters, 2023, 10.1109/LRA.2023.3331889 . hal-04305407

HAL Id: hal-04305407

<https://hal.science/hal-04305407>

Submitted on 24 Nov 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Risk-Averse Trajectory Optimization via Sample Average Approximation

Thomas Lew¹, Riccardo Bonalli², Marco Pavone¹

¹Department of Aeronautics and Astronautics, Stanford University

²Laboratory of Signals and Systems, University of Paris-Saclay, CNRS, CentraleSupélec

Abstract—Trajectory optimization under uncertainty underpins a wide range of applications in robotics. However, existing methods are limited in terms of reasoning about sources of epistemic and aleatoric uncertainty, space and time correlations, nonlinear dynamics, and non-convex constraints. In this work, we first introduce a continuous-time planning formulation with an average-value-at-risk constraint over the entire planning horizon. Then, we propose a sample-based approximation that unlocks an efficient and general-purpose algorithm for risk-averse trajectory optimization. We prove that the method is asymptotically optimal and derive finite-sample error bounds. Simulations demonstrate the high speed and reliability of the approach on problems with stochasticity in nonlinear dynamics, obstacle fields, interactions, and terrain parameters.

I. INTRODUCTION

Accounting for uncertainty in the design of decision-making systems is key to achieving reliable robotics autonomy. Indeed, modern autonomy stacks account for uncertainty [2, 57], whether it comes from noisy sensor measurements (e.g., due to perceptually-degraded conditions or a lack of features [2, 20]), dynamics (e.g., due to disturbances and difficult-to-characterize nonlinearities [16, 28, 39]), properties of the environment (e.g., due to unknown terrain properties for legged robots [18, 21] and Mars rovers [14, 29]), or interactions with other agents (e.g., in autonomous driving [30, 50, 52]).

Although trajectory optimization under uncertainty underpins a wide range of applications, existing approaches often make simplifying assumptions and approximations that reduce the range of problems they can reliably deal with. Specifically, there is a lack of methods capable of simultaneously handling

- sources of aleatoric uncertainty (e.g., external disturbances) and epistemic uncertainty (e.g., a drone transporting a payload of uncertain mass that introduces time correlations over the state trajectory) that depend on state and control variables (e.g., interactions between different agents),
- uncertainty of arbitrary (non-Gaussian) distribution correlated over time and space (e.g., uncertain terrain properties),
- uncertain nonlinear dynamics and non-convex constraints,
- trajectory-wise constraints that bound the risk of constraints violations over the entire duration of the problem.

As distinct systems may need specific numerical schemes, such methods should also have discretization-independent guarantees. Table I summarizes the capabilities of existing methods.

This work was supported by the NASA University Leadership Initiative (grant#80NSSC20M0163) and the Air Force under an STTR award with Altius Space Machines, but solely reflects the opinions and conclusions of its authors. Toyota Research Institute provided funds to support this work.

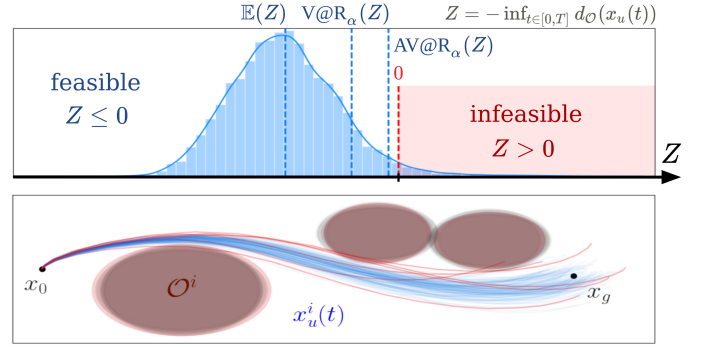


Fig. 1: Full-horizon collision statistics. We consider problems with average-value-at-risk (AV@R) constraints enforced over the entire trajectory of the system. Uncertain obstacles and nonlinear dynamics, with uncertain inertia properties and disturbances modeled as Brownian motion, introduce sources of aleatoric and epistemic uncertainty that are notoriously challenging to handle with existing approaches.

Contributions: We introduce an efficient risk-averse trajectory optimization algorithm satisfying the previous desiderata:

- First, we propose a risk-averse planning formulation with average-value-at-risk (AV@R [54]) constraints enforced over the entire planning horizon. This formulation is applicable to a wide range of robotics problems with sources of aleatoric and epistemic uncertainty. Its continuous-time nature guides the design of algorithms whose properties are independent of the chosen time discretization scheme (see Remark 2). Enforcing AV@R constraints enables accounting for tail events and facilitates numerical resolution due to their convexity properties (Remark 3). In addition, solving this formulation gives feasible solutions to notoriously challenging problems with joint chance constraints (Remark 1).
- Second, we propose a sample-based approximation rooted in the sample average approximation approach [40, 46, 54]. We derive asymptotic optimality guarantees (Theorem 1) and finite-sample error bounds (Lemma 1) for this reformulation. The analysis relies on mild assumptions ((A1)-(A4)), which enables considering a wide range of uncertainty sources that introduce spatial and temporal correlations and depend on state and control variables. The resulting approximated problem is smooth and sparse, facilitating efficient numerical resolution using off-the-shelf optimization tools.
- Finally, we show the reliability and speed of the proposed approach on problems with uncertain nonlinear dynamics, obstacle fields, interactions, and terrain parameters.

This work indicates that risk-averse planning problems can be efficiently tackled via trajectory optimization. These findings

TABLE I: Capabilities of existing approaches for risk-constrained planning and trajectory optimization.

	dynamics uncertainty	constraints uncertainty	aleatoric uncertainty	epistemic uncertainty	nonlinear dynamics	non-convex constraints	non-Gaussian distributions	time & space correlations
[25]		•		•	•		•	•
[32]	•		•			•	•	•
[51]	•		•		•	•	•	
[56]	•	•	•		•	•		
[38]	•		•		•	•		
[28]	•		•		•			
[18, 33]		•		•	•	•	•	
[26]	•	•	•	•	•	•	•	
[12]	•	•	•		•	•		
[15, 44]	•		•					
This work	•	•	•	•	•	•	•	•

challenge the popular belief that Monte-Carlo-based planning methods are computationally expensive [22, 38, 56]. The key is a particular AV@R-constrained formulation which, when coupled with a sample-based approximation and a smooth, sparse reformulation, unlocks the use of readily available optimization tools that enable efficient numerical resolution.

II. RELATED WORK

Risk-averse control, planning, and trajectory optimization methods typically enforce chance constraints (CCs) or average-value-at-risk (AV@R) constraints. AV@R constraints are more conservative than CCs, as they account for tail events, and have desired properties for robotics applications [13]. Two types of risk constraints are popular in the literature. First, *pointwise* risk constraints enforce individual constraints at each time $t \in [0, T]$, and thus do not provide sufficient constraints satisfaction guarantees over the *entire* time horizon $[0, T]$, see Remark 2. Instead, it is preferable to enforce *joint* risk constraints that should hold jointly at all times. To enforce such joint risk constraints, many approaches bound the total risk of constraints violations using Boole’s inequality [12, 41, 45], which neglects time correlations of uncertainty. As a result, these methods yield solutions that tend to be conservative and sensitive to the chosen time discretization [22, 32, 39].

The main challenge in risk-averse planning and control lies in the ability of efficiently evaluating the risk of constraints violations (e.g., the probability or the AV@R of constraints violations over the planning horizon). For instance, estimating the joint probability of collision of a robotic system until it completes a task requires accounting for the effects of disturbances across multiple timesteps, and generally requires evaluating an expectation integral over time and space, which is challenging for general uncertainty distributions and constraints. For this reason, many methods in the literature assume independent Gaussian-distributed random variables defining the problem [7, 21, 28, 40], polytopic constraints sets [25, 28, 50], or use Boole’s inequality to distribute risk over different constraints (e.g., obstacles) [28, 38, 51] which neglects correlations of uncertainty over the statespace. These are reasonable assumptions and approximations in some applications. However, these approaches may lead to infeasibility, under-estimate the true risk of constraints violation, or hinder performance. There is a lack of formulations and solution algorithms capable of truly capturing different sources of uncertainty, see Table I.

Monte-Carlo methods fulfill the desideratas introduced in Section I, but are often considered to be computationally

expensive [22, 38, 56]. Existing Monte-Carlo risk-averse trajectory optimization and planning approaches are limited to problems with randomly-moving obstacles of fixed polytopic shapes for deterministic linear systems [25], or require solving multiple problems for different paddings of the obstacles as a function of the constraints violation probability estimated from samples [32]. Both approaches are computationally expensive: the first requires using mixed-integer programming, and the latter requires solving multiple instances of the problem [32]. Methods based on the scenario approach also use samples to enforce chance constraints [3, 5, 11, 53], but they are limited to convex problems (e.g., to systems with linear dynamics).

Our proposed continuous-time risk-constrained problem formulation is more general than in prior work (see Table I), capturing a broad range of trajectory planning problems. In particular, in Section VIII, we study a problem with uncertain nonlinear dynamics, sources of epistemic and aleatoric uncertainty, and moving obstacles, and a legged robot navigating over uncertain terrain whose friction coefficient varies over space. Importantly, the proposed reformulation is amenable to efficient numerical resolution, thanks to using a particular AV@R-constrained formulation approximated using samples, unlocking the use of reliable off-the-shelf optimization tools.

III. RISK FUNCTIONS

Let $(\Omega, \mathcal{G}, \mathbb{P})$ be a probability space [37] and $Z : \Omega \rightarrow \mathbb{R}$ be a random variable encoding constraints of the form $Z \leq 0$. For instance, Z may denote the minimum negative distance to obstacles and $Z \leq 0$ may denote obstacle avoidance constraints, see Figure 1. In practice, enforcing constraints with \mathbb{P} -probability one (i.e., $Z(\omega) \leq 0$ for \mathbb{P} -almost all $\omega \in \Omega$) is infeasible, e.g., if disturbances are Gaussian-distributed and have unbounded support. Thus, we may enforce *risk constraints* instead. We define the *Value-at-Risk* (V@R) and *Average Value-at-Risk* (AV@R)¹ at risk level $\alpha \in (0, 1)$ as

$$\text{V@R}_\alpha(Z) = \inf_{t \in \mathbb{R}} \left(t : \mathbb{P}(Z > t) \leq \alpha \right), \quad (1)$$

$$\text{AV@R}_\alpha(Z) = \inf_{t \in \mathbb{R}} \left(t + \frac{1}{\alpha} \mathbb{E}[\max(Z - t, 0)] \right). \quad (2)$$

V@R $_\alpha(Z)$ is the $(1 - \alpha)$ -quantile of Z ($Z > \text{V@R}_\alpha(Z)$ with probability less than α) and AV@R $_\alpha(Z)$ is the expected value of values of Z larger than V@R $_\alpha(Z)$ [54, Theorem 6.2]. These

¹The AV@R is also often referred to as the *Conditional Value-at-Risk* (CV@R) in the literature [42], since $\text{AV@R}_\alpha(Z) = \mathbb{E}[Z | Z \geq \text{V@R}_\alpha(Z)]$ under certain regularity assumptions [54, Theorem 6.2].

risk functions yield two types of inequality constraints. First, the $V@R_\alpha$ gives chance constraints at probability level α :

$$V@R_\alpha(Z) \leq 0 \iff \mathbb{P}(Z > 0) \leq \alpha. \quad (3)$$

The $AV@R$ gives conservative formulations of chance constraints since $V@R_\alpha(Z) \leq AV@R_\alpha(Z)$ [54]:

$$AV@R_\alpha(Z) \leq 0 \implies \mathbb{P}(Z > 0) \leq \alpha. \quad (4)$$

IV. PROBLEM FORMULATION

We model the uncertain system in continuous-time using a stochastic differential equation (SDE). Let $(\Omega, \mathcal{G}, \mathcal{F}, \mathbb{P})$ be a filtered probability space [37], W be a n -dimensional Brownian motion on Ω [36] (e.g., the standard Wiener process), $\xi : \Omega \rightarrow \mathbb{R}^q$ be a random variable representing $q \in \mathbb{N}$ uncertain parameters, $n, m \in \mathbb{N}$ be state and control dimensions, $x_0 : \Omega \rightarrow \mathbb{R}^n$ be uncertain initial conditions, $U \subset \mathbb{R}^m$ be a compact control constraint set, $T > 0$ be the planning horizon, $b : \mathbb{R}^n \times U \times \mathbb{R}^q \rightarrow \mathbb{R}^n$ and $\sigma : \mathbb{R}^n \times U \times \mathbb{R}^q \rightarrow \mathbb{R}^{n \times n}$ be uncertain drift and diffusion coefficients. Given a control trajectory $u : [0, T] \rightarrow U$, we define the SDE

$$\begin{aligned} dx(t) &= b(x(t), u(t), \xi)dt + \sigma(x(t), u(t), \xi)dW_t, \quad t \in [0, T], \\ x(0) &= x_0, \end{aligned} \quad (\text{SDE})$$

with solution x_u . Given running and final costs $\ell : \mathbb{R}^n \times U \rightarrow \mathbb{R}$ and $\varphi : \mathbb{R}^n \rightarrow \mathbb{R}$, $N \in \mathbb{N}$ inequality and $n_h \in \mathbb{N}$ equality constraints functions $G_j : \mathbb{R}^n \times \mathbb{R}^q \rightarrow \mathbb{R}$ and $H : \mathbb{R}^n \rightarrow \mathbb{R}^{n_h}$, a risk parameter $\alpha \in (0, 1)$, and a control space \mathcal{U} (see Sections VI-VII), we define the optimal control problem (**OCP**):

$$\text{OCP} : \inf_{u \in \mathcal{U}} \mathbb{E} \left[\int_0^T \ell(x_u(t), u(t))dt + \varphi(x_u(T)) \right] \quad (5a)$$

$$\text{s.t. } AV@R_\alpha \left(\sup_{t \in [0, T]} G(x_u(t), \xi) \right) \leq 0, \quad (5b)$$

$$\mathbb{E}[H(x_u(T))] = 0, \quad (5c)$$

$$x_u \text{ satisfies (SDE)}, \quad (5d)$$

where $G(x_u(t), \xi) = \max_{j=1, \dots, N} G_j(x_u(t), \xi)$ is the maximum constraints violation at time t and we implicitly omit dependencies on uncertainties ω .

This formulation of **OCP** captures a broad range of robotics applications, see Section VIII for examples. **OCP** is challenging to solve due to the non-convexity of G , the $AV@R$ risk constraint (5b), the non-Gaussianity of the state trajectory x_u , and the dependency of all quantities on both epistemic uncertainty (modeled by the uncertain parameters ξ) and aleatoric uncertainty (modeled with the SDE in (5d)).

By (2), the $AV@R$ constraint (5b) in **OCP** is equivalent to

$$t + \frac{1}{\alpha} \mathbb{E} \left[\max \left(\sup_{s \in [0, T]} G(x_u(s), \xi) - t, 0 \right) \right] \leq 0, \quad (6)$$

so all constraints in **OCP** are expected value constraints. Thus, solving **OCP** amounts to evaluating expectations, which is generally challenging as it involves a nonlinear SDE and general nonlinear constraints functions. We propose a tractable approximation in Section V. Next, we discuss important considerations motivating this formulation.

Remark 1 (joint chance constraints (CCs)). Solving **OCP** yields feasible solutions to problems with joint CCs. Indeed, given a constraint $G(x_u(t), \xi) \leq 0$ that should hold at all times jointly with high probability $1 - \alpha$, we can define the joint CC

$$\mathbb{P} \left(\sup_{t \in [0, T]} G(x_u(t), \xi) > 0 \right) \leq \alpha. \quad (7)$$

Thanks to (3) and (4), a conservative formulation of the joint CC (7) is the corresponding constraint (5b) in **OCP**. Thus, by appropriately defining G and solving **OCP**, constraints that often appear in robotics can be enforced with high probability over the *entire* state trajectory, see Section VIII.

Remark 2 (pointwise-in-time risk constraints). Enforcing

$$AV@R_\alpha(G(x_u(t), \xi)) \leq 0 \quad \forall t \in [0, T], \quad (8)$$

instead (e.g., as in [25]) does not bound the risk of constraints violation over the entire trajectory. Similarly, pointwise-in-time chance constraints (e.g., as in [28, 38]) do not bound the probability of constraints violations over the entire planning horizon. Further, transposing discrete-time risk-averse control strategies to full-horizon settings via Boole's inequality [12, 41, 45] may lead to infeasibility as the resolution of the time discretization increases, see [22, 32, 39] for further discussion. Placing the time-wise supremum inside the $AV@R$ constraint (5b) and accounting for time correlations is key to obtaining trajectory-wise constraints satisfaction guarantees.

Example 1 (obstacle avoidance constraints). $N \in \mathbb{N}$ obstacles \mathcal{O}_j of uncertain positions and shapes to be avoided at all times can be captured via signed distance functions (SDFs) $d_{\mathcal{O}_j} : \mathbb{R}^n \times \mathbb{R}^q \rightarrow \mathbb{R}$, $(x, \xi) \mapsto d_{\mathcal{O}_j(\xi)}(x)$ such that $d_{\mathcal{O}_j}(x_u(t)) \geq 0$ if and only if the system is collision-free, i.e., $x_u(t) \notin \mathcal{O}_j$. For example, spherical obstacles at o_j of radii r_j can be described by $d_{\mathcal{O}_j}(x) = \|x - o_j\| - r_j$. The fact that $d_{\mathcal{O}_j}$ depends on randomness $\omega \in \Omega$ via the uncertain parameters $\xi(\omega)$ allows capturing obstacles of uncertain position and shape. For example, ellipsoidal obstacles centered at o_j of shape matrices Q_j can be encoded by the SDFs

$$d_{\mathcal{O}_j(\xi(\omega))}(x) = (x - o_j(\omega))^T Q_j(\omega)(x - o_j(\omega)) - 1. \quad (9)$$

Collision avoidance joint CCs can be written as

$$\mathbb{P}(x_u(t) \notin \mathcal{O}_j(\xi) \quad \forall t \in [0, T] \quad \forall j = 1, \dots, N) \geq 1 - \alpha. \quad (10)$$

By defining the N constraints function $G_j = -d_{\mathcal{O}_j}$, so that

$$G(x_u(t), \xi) = \max_{j=1, \dots, N} (-d_{\mathcal{O}_j(\xi)}(x_u(t))), \quad (11)$$

the obstacle avoidance joint CC (10) is equivalent to (7). Thus, by Remark 1, a conservative reformulation of (10) is (5b):

$$AV@R_\alpha \left(\sup_{t \in [0, T]} \max_{j=1, \dots, N} (-d_{\mathcal{O}_j(\xi)}(x_u(t))) \right) \leq 0. \quad (12)$$

V. SAMPLE AVERAGE APPROXIMATION (SAA)

Given M independent and identically distributed (iid) samples $\omega^i \in \Omega$, captured by the multi-sample $\bar{\omega} = (\omega^1, \omega^2, \dots)$

with joint distribution $\bar{\mathbb{P}}$, we approximate **OCP** by the following sampled optimal control problem (**SOCP**):

$$\text{SOCP}_{M(\bar{\omega})} \quad \inf_{\substack{u \in \mathcal{U} \\ t \in \mathbb{R}}} \frac{1}{M} \sum_{i=1}^M \int_0^T \ell(x_u^i(t), u(t)) dt + \varphi(x_u^i(T)) \quad (13a)$$

$$\text{s.t.} \quad t + \frac{1}{\alpha M} \sum_{i=1}^M \max \left(\sup_{s \in [0, T]} G(x_u^i(s), \xi^i) - t, 0 \right) \leq 0, \quad (13b)$$

$$- \delta_M \leq \frac{1}{M} \sum_{i=1}^M H(x_u^i(T)) \leq \delta_M, \quad (13c)$$

$$x_u \text{ satisfies (SDE)}, \quad (13d)$$

where $(x_u^i, \xi^i) = (x_u(\omega^i), \xi(\omega^i))$ for $i = 1, \dots, M$, (13c) corresponds to n_h (pointwise) inequality constraints, and $\delta_M > 0$ is a padding constant that decreases as the number of samples M increases (in practice, we set δ_M to a small value, see Theorem 1). This approximation is inspired from the sample average approximation (SAA) method [46, 54], which was recently extended to problems with equality constraints that often appear in robotics applications [40]. To the best of our knowledge, this approximation has not been applied to continuous-time problems taking the form of **OCP**.

Given M samples $\omega^i \in \Omega$ of the uncertainty, which define M sample paths of the Brownian motion $W(\omega^i)$, of the uncertain parameters $\xi(\omega^i)$, and of the initial conditions $x_0(\omega^i)$, the approximation $\text{SOCP}_{M(\bar{\omega})}$ is a tractable deterministic trajectory optimization problem, see Section VII. In the next section, we study the theoretical properties of this approach.

VI. THEORETICAL ANALYSIS

Depending on the samples ω^i , the computed control trajectories $u_M(\bar{\omega})$ that solve $\text{SOCP}_{M(\bar{\omega})}$ will be different. What can we say about the quality of these solutions? We provide analysis that relies on the following mild assumptions.

(A1) The drift and diffusion coefficients b and σ are continuous. Further, there is a bounded constant $K \geq 0$ such that for all $x, y \in \mathbb{R}^n$, all $u, v \in U$, and all values $\xi \in \mathbb{R}^q$, $\|b(x, u, \xi) - b(y, v, \xi)\| + \|\sigma(x, u, \xi) - \sigma(y, v, \xi)\| \leq K(\|x - y\| + \|u - v\|)$.

(A2) The cost and constraints functions (ℓ, φ, G, H) are continuous. Further, there is a bounded constant $L \geq 0$ such that for all $x, y \in \mathbb{R}^n$, all $u, v \in U$, and all values $\xi \in \mathbb{R}^q$, $\|G(x, \xi) - G(y, \xi)\| + \|H(x) - H(y)\| \leq L\|x - y\|$, $\|G(x, \xi)\| + \|H(x)\| \leq L$, $\|\ell(x, u) - \ell(y, v)\| + \|\varphi(x) - \varphi(y)\| \leq L(\|x - y\| + \|u - v\|)$.

(A3) The control space $\mathcal{U} \subset L^2([0, T], U)$ can be identified with a compact subset of \mathbb{R}^z for some $z \in \mathbb{N}$.

(A4) The uncertain initial state x_0 and parameters ξ are \mathcal{F}_0 -measurable and square-integrable.

(A1) is standard and guarantees the existence and uniqueness of solutions to (SDE). (A2) corresponds to standard smoothness assumptions of the cost and constraints functions. The constraints functions G and H can always be composed with a smooth cut-off function whose support contains the statespace

of interest to ensure the satisfaction of the boundedness condition in (A2). (A3) states that the control space \mathcal{U} is finite-dimensional, which is an assumption that is satisfied in practical applications once a numerical resolution scheme is selected. In particular, (A3) holds for the space of stepwise-constant control inputs $\mathcal{U} = (16)$ that we use in this work. (A4) makes rigorous the interpretation of x_0 and ξ as sources of epistemic uncertainty: (A4) states that the uncertain initial state x_0 and parameters ξ are randomized at the beginning of the episode and are independent of the Brownian motion W .

To describe the distance between solutions to SOCP_M and to **OCP**, given non-empty compact sets $A, B \subseteq \mathcal{U}$, we define

$$\mathbb{D}(A, B) = \sup_{u \in A} \inf_{v \in B} \|u - v\|. \quad (14)$$

As defined above, \mathbb{D} satisfies the property that $A \subseteq B$ if $\mathbb{D}(A, B) = 0$. Thus, if we show that the solution sets $S_M(\bar{\omega})$ and S of $\text{SOCP}_{M(\bar{\omega})}$ and **OCP** satisfy $\mathbb{D}(S_M(\bar{\omega}), S) = 0$, then we can conclude that $S_M(\bar{\omega}) \subseteq S$, i.e., any optimal solution of $\text{SOCP}_{M(\bar{\omega})}$ is an optimal solution of the original problem **OCP**. Theorem 1 below states that this result holds with probability one in the limit as the sample size M increases.

Theorem 1 (Asymptotic Optimality). *Given $M \in \mathbb{N}$ samples and any constants $C > 0$ and $\epsilon \in (0, \frac{1}{2})$, define $\delta_M = CM^{(\epsilon - \frac{1}{2})}$, and denote the sets of optimal solutions to **OCP** and $\text{SOCP}_{M(\bar{\omega})}$ by S and $S_M(\bar{\omega})$, respectively. Then, under assumptions (A1)-(A4), $\bar{\mathbb{P}}$ -almost-surely, $\lim_{M \rightarrow \infty} \mathbb{D}(S_M(\bar{\omega}), S) = 0$.*

The proof of Theorem 1 follows from recent results in [40], see the appendix. Theorem 1 gives convergence guarantees to optimal solutions to **OCP** (in particular, the AV@R constraint (5b) is satisfied) as the sample size increases, justifying the proposed approach. This asymptotic optimality result is derived in continuous-time and is thus independent of the chosen discretization scheme for numerical resolution. This contrasts with methods that start with a discrete-time formulation and may yield different results for different discretizations [22, 32].

The following result gives high-probability *finite-sample* error bounds for the average-value-at-risk constraint given a solution of the sample-based approximation. The proof (see the appendix) relies on concentration inequalities [4, 34, 40, 59].

Lemma 1 (Finite-Sample Error Bound). *Let $\epsilon > 0$, $\beta \in (0, 1)$, and $M \in \mathbb{N}$ be such that $M \geq \epsilon^{-2}(\tilde{C} + \bar{h}(2 \log(1/\beta))^{\frac{1}{2}})^2$ for some finite constants (\tilde{C}, \bar{h}) large-enough. Denote any solution to $\text{SOCP}_{M(\bar{\omega})}$ by $u_M(\bar{\omega})$. Then, under assumptions (A1)-(A4),*

$$\text{AV@R}_\alpha \left(\sup_{s \in [0, T]} G(x_{u_M(\bar{\omega})}(s), \xi) \right) \leq \epsilon \quad (15)$$

with $\bar{\mathbb{P}}$ -probability at least $(1 - \beta)$ over the M iid samples ω^i .

By Lemma 1, replacing constraint (13b) in $\text{SOCP}_{M(\bar{\omega})}$ with

$$t + \frac{1}{\alpha M} \sum_{i=1}^M \max \left(\sup_{s \in [0, T]} G(x_u^i(s), \xi^i) - t, 0 \right) \leq -\epsilon$$

would guarantee the satisfaction of the average-value-at-risk constraint in **OCP** if the sample size M is large-enough. The

error ϵ can be made arbitrarily small (with increasingly high probability $1-\beta$) by increasing the sample size M . In practice, the bound in Lemma 1 is conservative, and numerical results demonstrate that a few samples are sufficient to obtain high-quality solutions to challenging problems, see Section VIII.

VII. NUMERICAL RESOLUTION

Theorem 1 justifies approximating **OCP** using samples ω^i and searching for solutions to the deterministic relaxation **SOCP** $_M(\bar{\omega})$ instead. In this section, we describe a numerical method for efficiently computing solutions to **SOCP** $_M(\bar{\omega})$.

Control space parameterization: We optimize over open-loop controls u parameterized by $S \in \mathbb{N}$ stepwise-constant inputs u_s of duration $\Delta t = T/S$, described by the set

$$\mathcal{U} = \left\{ u : \begin{array}{l} u(t) = \sum_{s=0}^{S-1} u_s \mathbf{1}_{[s\Delta t, (s+1)\Delta t)}(t), \\ (u_0, \dots, u_{S-1}) \in U \times \dots \times U \end{array} \right\}. \quad (16)$$

This set \mathcal{U} clearly satisfies (A3), since it can be identified with a compact set of \mathbb{R}^{Sm} . Note that any square-integrable open-loop control u can be approximated arbitrarily well by some $u \in \mathcal{U}$ by increasing S , so this class of function is expressive.

Alternatively, one could also optimize over certain classes of closed-loop controllers (e.g., controls of the form $u = \bar{u} + Kx$); an approach that is common in the literature [24, 44].

Finite-dimensional approximation: Numerically solving general instances of **SOCP** $_M(\bar{\omega})$ requires discretizing the problem. In this work, we discretize **SOCP** $_M(\bar{\omega})$ as follows:

$$\inf_{\substack{u \in \mathcal{U} \\ t \in \mathbb{R}}} \frac{1}{M} \sum_{i=1}^M \left(\sum_{k=0}^{S-1} \ell(x_u^i(k\Delta t), u(k\Delta t)) \Delta t + \varphi(x_u^i(S\Delta t)) \right) \quad (17a)$$

$$\text{s.t. } t + \frac{1}{\alpha M} \sum_{i=1}^M \max \left(\max_{k=0, \dots, S} G(x_u^i(k\Delta t), \xi^i) - t, 0 \right) \leq 0, \quad (17b)$$

$$- \delta_M \leq \frac{1}{M} \sum_{i=1}^M H(x_u^i(S\Delta t)) \leq \delta_M, \quad (17c)$$

$$\begin{aligned} x_u^i((k+1)\Delta t) &= x_u^i(k\Delta t) + b(x_u^i(k\Delta t), u(k\Delta t), \xi^i) \Delta t + \\ &+ \sigma(x_u^i(k\Delta t), u(k\Delta t), \xi^i) (W_{(k+1)\Delta t}^i - W_{k\Delta t}^i) \quad (17d) \\ &\forall k = 0, \dots, S-1, i = 1, \dots, M, \end{aligned}$$

$$x_u^i(0) = x_0^i \quad \forall i = 1, \dots, M, \quad (17e)$$

where $(x_0^i, \xi^i, W^i) = (x_0(\omega^i), \xi(\omega^i), W(\omega^i))$ denote the M samples of the initial conditions, parameters, and sample paths of the Brownian motion, and $x_u^i(k\Delta t) = x_u(k\Delta t, \omega^i)$ denotes associated sample paths of the state trajectory, for conciseness. The constraint (17d) corresponds to an Euler-Maruyama discretization of (SDE). This discretization of **SOCP** allows for a simple and efficient implementation, although more accurate integration schemes could be used to formulate **SOCP** $_M(\bar{\omega})$.

The SDE constraint (17d) can be either explicitly enforced by optimizing over both state variables $x_u^i(k\Delta t)$ and control variables $u(k\Delta t)$ and enforcing (17d), or implicitly by only optimizing over the control variables $u(k\Delta t)$ that parameterize the state trajectory via (17d). In this work, we opt for the latter as it reduces the number of variables, albeit at a potential reduction in numerical stability. As shown in [19], the alternative option of parameterizing both state particles and control variables could also be computationally efficient.

By introducing M additional variables $y_i \in \mathbb{R}$, the inequality constraints (17b) are equivalent to the set of the constraints

$$\begin{cases} (M\alpha)t + \sum_{i=1}^M y_i \leq 0 \\ 0 \leq y_i \quad \forall i = 1, \dots, M \\ G_j(x_u^i(k\Delta t), \xi^i) - t \leq y_i \quad \begin{cases} \forall i = 1, \dots, M \\ \forall j = 1, \dots, N \\ \forall k = 0, \dots, S. \end{cases} \end{cases} \quad (18)$$

Note that these constraints are smooth if every G_j is smooth.

Numerical resolution: With this reformulation, **SOCP** can be solved using reliable off-the-shelf optimization tools such as IPOPT [58]. In this work, we leverage sequential convex programming (SCP). SCP consists of solving a sequence of convex approximations of **SOCP** until convergence. The main appeal in using SCP is that it typically only requires a few convex approximations of the original non-convex program to reach accurate solutions. Since the set of constraints (18) is sparse, the convex approximations of **SOCP** can be efficiently solved. In contrast, interior-point-methods for non-convex programming may take a larger number of small steps that each require evaluating gradients and Hessians of the original program. Since this gradient-hessian evaluation is potentially the computational bottleneck in solving **SOCP**, SCP is a promising solution scheme for this class of problems. See [7, 8, 38] for further details on SCP for trajectory optimization.

Remark 3 (AV@R vs chance constraints). *Beyond the ability to account for tail events, the smoothness of the reformulation in (18) motivates enforcing AV@R constraints instead of chance constraints that are often used in the literature. Indeed, a typical chance constraint $\mathbb{P}(\sup_t G(x_u(t), \xi) > 0) \leq \alpha$ is equivalently written as $\mathbb{E}[\mathbf{1}_{(0, \infty)}(\sup_t G(x_u(t), \xi))] \leq \alpha$, where $\mathbf{1}_{(0, \infty)}(z) = 1$ if $z > 0$ and 0 otherwise. This constraint only takes a simple form in particular cases [10]. Approximating this chance constraint from samples would give*

$$\frac{1}{M} \sum_{i=1}^M \mathbf{1}_{(0, \infty)} \left(\sup_t G(x_u^i(t), \xi^i) \right) \leq \alpha,$$

which is not smooth since $\mathbf{1}_{(0, \infty)}(\cdot)$ is a step function. To solve the resulting problem with gradient-based methods, one would need to formulate and solve smooth approximations instead [23, 48], or solve the problem multiple times with different constraints paddings [31], which is computationally expensive. In contrast, the reformulation in (18) is smooth and exact (up to errors from the sample-based and discrete-time approximations), which allows efficient numerical resolution.

Example 1 (obstacle avoidance constraints). *With M samples of the state trajectory x_u^i and of the obstacles $\mathcal{O}_j(\xi^i)$, the AV@R constraint in (12) can be approximated with the set of constraints (18) with $G_j(x_u^i(k\Delta t), \xi^i) = -d_{\mathcal{O}_j(\xi^i)}(x_u^i(k\Delta t))$. This formulation applies to general obstacle representations and can be specialized to particular obstacle shapes. For example, if the N uncertain obstacles \mathcal{O}_j are spheres of uncertain radii r_j and centers o_j , then the last term in (18) is*

$$r_j^i - \|x_u^i(k\Delta t) - o_j^i\| - t \leq y_i \quad (19)$$

for all i, j, k , where $(r_j^i, o_j^i) = (r_j, o_j)(\omega^i)$ are M iid samples. If the obstacles are ellipsoidal as in (9), then (18) becomes

$$1 - (x_u^i(k\Delta t) - o_j^i)^\top Q_j^i (x_u^i(k\Delta t) - o_j^i) - t \leq y_i \quad (20)$$

for all i, j, k . This gives a set of differentiable constraints that can be passed to a non-convex optimization algorithm.

We note that potential corner-cutting due to the time discretization of the AV@R constraint is easily addressed via different methods, e.g., with the approach in [15], by enforcing (17b) on a finer grid, or by padding obstacles.

VIII. APPLICATIONS AND RESULTS

We apply the proposed approach to three challenging planning problems with diverse sources of uncertainty. Code is available at <https://github.com/StanfordASL/RiskAverseTrajOpt>.

1) Drone planning with uncertain obstacles. The state is $x = (p, \dot{p}) \in \mathbb{R}^6$, the input is $u \in \mathbb{R}^3$. Dynamics are given by

$$b(x, u, \omega) = \begin{bmatrix} \dot{p} \\ 0 \end{bmatrix} + \frac{1}{m(\omega)} \left(-\beta_{\text{drag}} \begin{bmatrix} 0 \\ |\dot{p}|\dot{p} \end{bmatrix} + \begin{bmatrix} 0_{3 \times 3} \\ I_{3 \times 3} \end{bmatrix} (u + Kx) \right), \quad (21)$$

and $\sigma(x, u, \omega) = \frac{1}{m(\omega)} (0_{3 \times 3}, \beta_\sigma I_{3 \times 3})^\top$. The drone transports an uncertain payload, modeled by assuming that the total mass m of the system follows a uniform distribution. $(\beta_{\text{drag}}, \beta_\sigma)$ are drag and diffusion coefficients and K is a feedback gain. We consider three uncertain ellipsoidal obstacles whose shape matrices have uncertain axes distributed according to a uniform distribution, and enforce collision avoidance constraints as described in Example 1. The objective is reaching a goal $(H(x(T)) = x(T) - x_g)$ while minimizing control effort $\ell(x, u) = u^\top R u$ and satisfying collision avoidance constraints at risk level α as in (12). This nonlinear system has sources of aleatoric (disturbances modeled as a Wiener process) and epistemic (mass and obstacles) uncertainty, which makes solving the problem with classical approaches challenging.

We formulate the sample-based approximation with $S = 20$ nodes and solve it with SCP. We present an example of results in Figure 1 ($M = 50$, $\alpha = 10\%$) with the trajectory samples x_u^i obtained at convergence. We report the associated Monte-Carlo histogram of the negative minimum distance over $t \in [0, T]$ in (11) with the mean, $V@R_\alpha$ and $AV@R_\alpha$ minimum values of (11). The average-value-at-risk of collision is below 0, indicating that the solution of **SOCP** is feasible for **OCP**.

Sensitivity to risk parameter: We run the approach (SAA_α) for different values of α each and report results in Table II. The $AV@R_\alpha$ constraint is approximately satisfied ($AV@R_\alpha \approx 0$) for each value of α . As a result, the associated joint chance constraints for collision avoidance are satisfied, since the percentage of constraints violations is always below α , validating the discussion in Remark 1. Safer behavior is obtained for smaller values of α , effectively balancing the tradeoff between the efficiency and the risk of constraints violations.

Comparisons: The proposed approach (SAA_α) yields trajectories that closely match desired safety risk levels. In contrast, a baseline that neglects uncertainty (Deterministic) often violates constraints. We also compare with a method ($Gaussian_\alpha$) that bounds the total probability of constraints violation below α using Boole's inequality. This baseline uses an approximate Gaussian-distributed state representation [38] and optimizes

over the risk allocation, see the appendix for details. Due to the approximate uncertainty representation and the use of Boole's inequality (thus neglecting time and space correlations of uncertainty), this baseline is overly conservative.

Computation time: In Figure 2, we report total computation times for different sample sizes $M \in \{20, 30, 50\}$ (we report the median over 30 runs with $\alpha = 5\%$, evaluated on a laptop with an i7-10710U CPU (1.10 GHz) and 16 GB of RAM). We use a zero initial guess $\bar{u}_s = 0$ and stop after 10 SCP iterations, which is sufficient

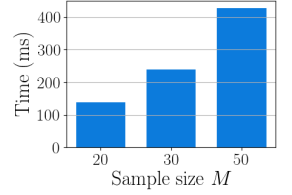


Fig. 2: Drone planning total computation times.

to obtain a final SCP iteration error $\|u^k - u^{k-1}\|/\|u^k\| \leq 1\%$. Albeit our implementation is written in Python (with JAX [9] and OSQP [55]), computation times are reasonable and amenable to real-time applications. Computation time scales roughly linearly in the sample size. Parallelization on a GPU could enable using a larger sample size M while retaining speed, albeit results show that using a small sample size suffices to obtain feasible solutions.

2) Autonomous driving with a pedestrian. The state is $x = (x_{\text{ego}}, x_{\text{ped}})$ with $x_{\text{ego}} = (p_{\text{ego}}^x, p_{\text{ego}}^y, v_{\text{ego}}, \phi_{\text{ego}})$ the ego-vehicle and $x_{\text{ped}} = (p_{\text{ped}}^x, p_{\text{ped}}^y, v_{\text{ped}}^x, v_{\text{ped}}^y)$ a pedestrian, and $u = (a, \tau)$ the ego control input. The coupled system follows $dx_{\text{ego}} = (v_{\text{ego}} \cos(\phi_{\text{ego}}), v_{\text{ego}} \sin(\phi_{\text{ego}}), u) dt$, $dp_{\text{ped}} = v_{\text{ped}} dt$, and $dv_{\text{ped}} = f(x, \omega) dt + \sigma dW_t$, from $(x_{\text{ego},0}, x_{\text{ped},0})$ with known $x_{\text{ego},0}$ but Gaussian-distributed $x_{\text{ped},0}$ due to uncertainty from perception. The term $f(x, \omega)$ represents interaction forces inspired from the Social Force Model [27]

$$f(x(t), \omega) = \omega_1 (v_{\text{ped}}^{\text{des}} - v_{\text{ped}}(t)) e_{\text{ped}} + \omega_2 \frac{p_{\text{ego}}(t) - p_{\text{ped}}(t)}{\|p_{\text{ego}}(t) - p_{\text{ped}}(t)\|}, \quad (22)$$

where (ω_1, ω_2) represents the tendency of a pedestrian to maintain a desired speed $v_{\text{ped}}^{\text{des}}$ in the direction e_{ped} and actively avoid the car. A pedestrian with large values of ω_1 and small values of ω_2 tends to keep the same speed while neglecting the car. (ω_1, ω_2) represents a source of epistemic uncertainty: the personality of the pedestrian is randomized and does not change during the planning episode. Such interactions could be learned from data [30, 52] and incorporated into our method.

The objective is reaching a destination $p_{\text{ego}}^{\text{des}}$ ($H(x(T)) = p_{\text{ego}}(T) - p_{\text{ego}}^{\text{des}}$) while minimizing control effort $\ell(x, u) = u^\top R u$ and maintaining a minimum separation distance d_{sep}

$$AV@R_\alpha \left(\sup_{t \in [0, T]} d_{\text{sep}} - \|p_{\text{ego}}(t) - p_{\text{ped}}(t)\| \right) \leq 0. \quad (23)$$

This constraint is reformulated as described in Example 1. The resulting **OCP** problem combines sources of aleatoric (from the Wiener process disturbances) and epistemic uncertainty (from the pedestrian's uncertain behavior and initial pose).

We discretize the problem with $S = 20$ nodes and report results in Figure 3 and Table II. Reducing the risk parameter α yields safer trajectories that maintain a larger distance with the pedestrian at the expense of greater control effort. For all values of α , the AV@R constraints and corresponding joint chance constraints are satisfied. In contrast, the joint-chance-constrained baseline with optimal risk allocation ($Gaussian_\alpha$)

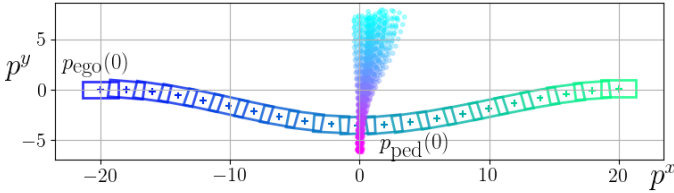


Fig. 3: Driving scenario results ($\alpha = 5\%$).

TABLE II: Monte-Carlo validation with 10^4 samples of disturbances and uncertain parameters. We report the median (over 30 runs with $M = 50$) of constraints violation percentage, risk, and control effort.

Drone planning		$\alpha = 5\%$	$\alpha = 10\%$	$\alpha = 20\%$	$\alpha = 30\%$
SAA $_{\alpha}$	constraints violations	2.1%	5.6%	8.4%	11.7%
	AV@R $_{\alpha}$	0.01	0.04	0.02	0.00
	cost	76.6	54.0	48.3	46.0
Gaussian $_{\alpha}$	constraints violations	1.3%	2.3%	3.9%	6.0%
	cost	69.9	61.8	54.5	50.7
Deterministic	constraints violations	73.8%			
	cost	31.4			
Autonomous driving		$\alpha = 1\%$	$\alpha = 2\%$	$\alpha = 5\%$	$\alpha = 10\%$
SAA $_{\alpha}$	constraints violations	0.2%	0.3%	0.8%	3.1%
	AV@R $_{\alpha}$	-0.03	-0.05	-0.05	-0.02
	cost	56.3	56.1	53.5	50.9
Gaussian $_{\alpha}$	constraints violations	0.0%	0.0%	0.3%	1.2%
	cost	60.2	58.1	55.0	52.5
Deterministic	constraints violations	53.1%			
	cost	42.5			

is overly conservative, whereas a baseline that neglects uncertainty (Deterministic) is unable to maintain a safe separation distance with the pedestrian at all times. Computation times are close to those for the drone planning problem in Figure 2 (see the appendix), again demonstrating the real-time capabilities of the proposed approach.

3) Legged navigation over uncertain terrain. Finally, we consider a legged robot whose goal is jumping as far as possible over uncertain terrain while limiting its risk of slippage. We consider the hopper robot in [35] with $x = (q, \dot{q})$ where $q = (p_x, p_z, \theta, r) \in \mathbb{R}^4$, $u = (\tau, f) \in \mathbb{R}^2$, and dynamics

$$M\ddot{q}(t) + C = B(q(t))u(t) + J_c(q(t))^{\top}\lambda(t), \quad x(0) = x^0, \quad (24)$$

where $\lambda = (\lambda_x, \lambda_z) \in \mathbb{R}^2$ are contact forces that must satisfy

$$\lambda_x(t) = \lambda_z(t) = 0 \quad \forall t \in \mathcal{T}_{\text{flight}}, \quad (25a)$$

$$J_{c,x}(q(t))^{\top}\dot{q}(t) = 0, \quad \lambda_z(t) \geq 0 \quad \forall t \in \mathcal{T}_{\text{contact}}, \quad (25b)$$

$$\text{and } \text{AV@R}_{\alpha} \left(\sup_{t \in \mathcal{T}_{\text{contact}}} \lambda_x(t) - \mu(q(t), \omega)\lambda_z(t) \right) \leq 0. \quad (25c)$$

These constraints encode the absence of contact force in the flight phase (25a), the no-slip and positive normal force conditions (25b), and a constraint on the risk of slippage (25c), over a pre-defined contact schedule $\mathcal{T}_{\text{flight}}, \mathcal{T}_{\text{contact}} \subset [0, T]$.

Characterizing terrain adhesion properties (i.e., terramechanics) is challenging and is often done via data-driven approaches. For instance, Cunningham et al. [14] fit Gaussian processes to observed data for slip prediction. Thus, we model the soil properties at the contact point $p_{\text{foot},x}(q)$ given by the

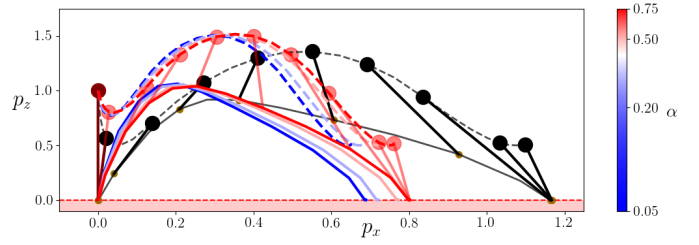


Fig. 5: Hopper system with uncertain terrain properties: jump trajectories (of the center of mass and the tip of the leg) for different risk parameter values α and for the deterministic baseline (in black).

system's kinematics with the Random Fourier Features [49] $\mu(q, \omega) = \bar{\mu} + \sum_{n=1}^{30} \omega_{n,1} \cos(\omega_{n,2} \cdot p_{\text{foot},x}(q) + \omega_{n,3})$ for some randomized parameters ω following a uniform distribution, see Figure 4. The coefficient μ encodes epistemic uncertainty varying over space, which is more realistic than assuming a constant friction coefficient as in [18]. As discussed in Section II, this type of uncertainty is challenging to deal with.

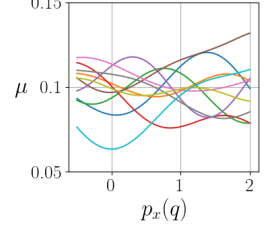


Fig. 4: Samples from μ .

The objective consists of jumping as far as possible (i.e., $\varphi(x(T)) = -p_x(T)$) while minimizing the control effort $\ell(x, u) = \|u\|_2^2$ and limiting the risk of slippage (25c). The resulting problem clearly takes the form of **OCP** with no dynamics uncertainty (i.e., $\sigma = 0$). Indeed, an important source of uncertainty in legged locomotion comes from uncertain terrain properties due to imperfect perception or inherent uncertainty in the problem. We are not aware of prior work in trajectory optimization tackling this formulation.

We solve the sampled problems (using $S = M = 30$) with IPOPT [58] and report results in Figure 5. By reducing the risk parameter α , the proposed trajectory optimizer is more cautious to avoid slip and returns shorter jumps, with slippage probability bounded by α . In contrast, a deterministic baseline that only considers the mean terrain parameter $\mu(x, \omega) = \bar{\mu}$ returns longer jumps but violates constraints 50% of the time.

IX. CONCLUSION

We proposed an efficient method for risk-averse trajectory optimization. This algorithm hinges on a continuous-time formulation with average-value-at-risk constraints. By approximating the problem using samples, we obtain a smooth, sparse program that allows for efficient numerical resolution. We demonstrated the speed and reliability of the method on problems with sources of epistemic and aleatoric uncertainty that are challenging to tackle with existing approaches.

Due to its generality in handling uncertain nonlinear dynamics and constraints, this work opens exciting future research directions. In particular, the approach could be interfaced with learned models: obstacles could be implicitly represented as deep SDFs [47] or neural radiance fields [1, 43], deep trajectory forecasting models [30, 52] could be used for planning in autonomous driving, learned terramechanics models [14] could allow more robust legged locomotion, and dynamics could be learned online, either to improve control performance [16, 28] or for active data-gathering [39, 60] under risk constraints.

REFERENCES

- [1] M. Adamkiewicz, T. Chen, A. Caccavale, R. Gardner, P. Culbertson, J. Bohg, and M. Schwager. Vision-only robot navigation in a neural radiance world. *IEEE Robotics and Automation Letters*, 7(2):4606–4613, 2022.
- [2] Agha, A. et al. NeBula: Quest for robotic autonomy in challenging environments; team CoSTAR at the DARPA SubT challenge. *Journal of Field Robotics*, 2021.
- [3] H. Ahn, C. Chen, I. M. Mitchell, and M. Kamgarpour. Safe motion planning against multimodal distributions based on a scenario approach. *IEEE Control Systems Letters*, 6:1142–1147, 2022.
- [4] P. L. Bartlett and S. Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2003.
- [5] D. Bernardini and A. Bemporad. Scenario-based model predictive control of stochastic constrained linear systems. In *Proc. IEEE Conf. on Decision and Control*, 2009.
- [6] R. Bonalli and B. Bonnet. First-order Pontryagin maximum principle for risk-averse stochastic optimal control problems. *SIAM Journal on Control and Optimization*, 2022.
- [7] R. Bonalli, T. Lew, and M. Pavone. Sequential convex programming for non-linear stochastic optimal control. *ESAIM: Control, Optimisation & Calculus of Variations*, 2022.
- [8] R. Bonalli, T. Lew, and M. Pavone. Analysis of theoretical and numerical properties of sequential convex programming for continuous-time optimal control. *IEEE Transactions on Automatic Control*, 2022.
- [9] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang. JAX: composable transformations of Python+NumPy programs, 2018.
- [10] G. C. Calafiore and L. El Ghaoui. On distributionally robust chance-constrained linear programs. *Journal of Optimization Theory & Applications*, 130(1):1–22, 2006.
- [11] G.C. Calafiore and M.C. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5):742–753, 2006.
- [12] M. Castillo-Lopez, P. Ludvig, S. A. Sajadi-Alamdari, J. L. Sanchez-Lopez, M. A. Olivares-Mendez, and H. Voos. A real-time approach for chance-constrained motion planning with dynamic obstacles. *IEEE Robotics and Automation Letters*, 5(2):3620–3625, 2020.
- [13] Y. Chow and M. Pavone. A framework for time-consistent, risk-averse model predictive control: Theory and algorithms. In *American Control Conference*, 2014.
- [14] C. Cunningham, M. Ono, I. Nesnas, J. Yen, and W. L. Whittaker. Locally-adaptive slip prediction for planetary rovers using Gaussian processes. In *Proc. IEEE Conf. on Robotics and Automation*, 2017.
- [15] M. da Silva Arantes, C. F. M. Toledo, B. C. Williams, and M. Ono. Collision-free encoding for chance-constrained nonconvex path planning. *IEEE Transactions on Robotics*, 35(2):433–448, 2019.
- [16] M. Deisenroth, D. Fox, and C. Rasmussen. Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 37(2):408–423, 2015.
- [17] M. Diehl, H. G. Bock, and J. P. Schlöder. A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM Journal on Optimization*, 43(5):1714–1736, 2005.
- [18] L. Drnack and Y. Zhao. Robust trajectory optimization over uncertain terrain with stochastic complementarity. *IEEE Robotics and Automation Letters*, 6(2):1168–1175, 2021.
- [19] R. Dyro, J. Harrison, A. Sharma, and M. Pavone. Particle MPC for uncertain and learning-based control. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2021.
- [20] D. Falanga, P. Foehn, P. Lu, and D. Davide Scaramuzza. PAMPC: Perception-aware model predictive control for quadrotors. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2018.
- [21] D. D. Fan, K. Otsu, Y. Kubo, A. Dixit, J. Burdick, and A. Agha-mohammadi. STEP: Stochastic traversability evaluation and planning for risk-aware off-road navigation. In *Robotics: Science and Systems*, 2021.
- [22] K. M. Frey, T. J. Steiner, and J. P. How. Collision probabilities for continuous-time systems without sampling. In *Robotics: Science and Systems*, 2020.
- [23] A. Geletu, A. Hoffmann, M. Klöppel, and P. Li. An inner-outer approximation approach to chance constrained optimization. *SIAM Journal on Optimization*, 27(3):1834–1857, 2017.
- [24] P. J. Goulart, E. C. Kerrigan, and J. M. Maciejowski. Optimization over state feedback policies for robust control with constraints. *Automatica*, 42(4):523–533, 2006.
- [25] A. Hakobyan, G. C. Kim, and I. Yang. Risk-aware motion planning and control using CVaR-constrained optimization. *IEEE Robotics and Automation Letters*, 4(4):3924–3931, 2019.
- [26] W. Han, A. M. Jasour, and B. Williams. Non-Gaussian risk bounded trajectory optimization for stochastic nonlinear systems in uncertain environments. In *Proc. IEEE Conf. on Robotics and Automation*, 2022.
- [27] D. Helbing and P. Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51(5):4282–4286, 1995.
- [28] L. Hewing, J. Kabzan, and M. N. Zeilinger. Cautious model predictive control using Gaussian process regression. *IEEE Transactions on Control Systems Technology*, 28(6):2736–2743, 2020.
- [29] S. Higa, Y. Iwashita, K. Otsu, M. Ono, O. Lamarre, A. Didier, and M. Hoffmann. Vision-based estimation of driving energy for planetary rovers using deep learning and terramechanics. *IEEE Robotics and Automation Letters*, 4(4):3876–3883, 2019.
- [30] B. Ivanovic, A. Elhafi, G. Rosman, A. Gaidon, and M. Pavone. MATS: An interpretable trajectory forecasting representation for planning and control. In *Conf. on Robot Learning*, 2020.
- [31] L. Janson, B. Ichter, and M. Pavone. Deterministic sampling-based motion planning: Optimality, complexity, and performance. In *Int. Symp. on Robotics Research*, 2015.
- [32] L. Janson, E. Schmerling, and M. Pavone. Monte Carlo motion planning for robot trajectory optimization under uncertainty. In *Int. Symp. on Robotics Research*, 2015.
- [33] W. Jasour, A. M. Han, , and B. Williams. Convex risk bounded continuous-time trajectory planning in uncertain nonconvex environments. In *Robotics: Science and Systems*, 2021.
- [34] V. Koltchinskii. Local Rademacher complexities and oracle inequalities in risk minimization. *The Annals of Statistics*, 34(6), 2006.
- [35] S. Le Cleac’h, T. A. Howell, S. Yang, C.-Y. Lee, J. Zhang, A. Bishop, M. Schwager, and Z. Manchester. Fast contact-implicit model predictive control. <https://arxiv.org/abs/2107.05616>, 2021.
- [36] J. F. Le Gall. *Brownian Motion, Martingales, and Stochastic Calculus*. Springer, 2016.
- [37] J. F. Le Gall. *Measure Theory, Probability, and Stochastic Processes*. Springer, 2022.
- [38] T. Lew, R. Bonalli, and M. Pavone. Chance-constrained sequential convex programming for robust trajectory optimization. In *European Control Conference*, 2020.
- [39] T. Lew, A. Sharma, J. Harrison, A. Bylard, and M. Pavone. Safe active dynamics learning and control: A sequential exploration-exploitation framework. *IEEE Transactions on Robotics*, 38(5):2888–2907, 2022.
- [40] T. Lew, R. Bonalli, and M. Pavone. Sample average approximation for stochastic programming with equality constraints. 2023. <https://arxiv.org/abs/2206.09963>.
- [41] Y. Ma, S. Vichik, and F. Borrelli. Fast stochastic MPC with optimal risk allocation applied to building control systems. In *Proc. IEEE Conf. on Decision and Control*, 2012.
- [42] A. Majumdar and M. Pavone. How should a robot assess risk? Towards an axiomatic theory of risk in robotics. In *Int. Symp. on Robotics Research*, 2017.
- [43] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conf. on Computer Vision*, 2020.
- [44] F. Oldewurtel, C. N. Jones, and M. Morari. A tractable approximation of chance constrained stochastic MPC based on affine disturbance feedback. In *Proc. IEEE Conf. on Decision and Control*, 2008.
- [45] M. Ono and B. C. Williams. Iterative risk allocation: A new approach to robust model predictive control with a joint chance constraint. In *Proc. IEEE Conf. on Decision and Control*, 2008.
- [46] B. K. Pagnoncelli, S. Ahmed, and A. Shapiro. Sample average approximation method for chance constrained programming: Theory and applications. *Journal of Optimization Theory and Applications*, 142(2):399–416, 2009.
- [47] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2019.
- [48] A. Peña-Ordieres, J. R. Luedtke, and A. Wächter. Solving chance-constrained problems via a smooth sample-based nonlinear approximation. *SIAM Journal on Optimization*, 30(3):2221–2250, 2020.
- [49] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Conf. on Neural Information Processing Systems*, pages 1177–1184, 2007.
- [50] K. Ren, H. Ahn, and M. Kamgarpour. Chance-constrained trajectory planning with multimodal environmental uncertainty. *IEEE Control Systems Letters*, 7:13–18, 2023.
- [51] S. Safaoui, B. J. Gravell, V. Renganathan, and T. H. Summers. Risk-averse RRT* planning with nonlinear steering and tracking controllers for nonlinear robotic systems under uncertainty. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2021.
- [52] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *European Conf. on Computer Vision*, 2020.
- [53] G. Schildbach, K. Fagiano, C. Frei, and M. Morari. The scenario approach for stochastic model predictive control with bounds on closed-loop constraint violations. *Automatica*, 50(12):3009–3018, 2014.
- [54] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory, Second Edition*. Society for Industrial and Applied Mathematics, 2014.
- [55] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd. OSQP: an operator splitting solver for quadratic programs. *Mathematical Programming Computation*, 12(4):637–672, 2020.
- [56] T. Thomas, F. Mastrogiovanni, and M. Baglietto. Exact and bounded collision probability for motion planning under Gaussian uncertainty. *IEEE Robotics and Automation Letters*, 7(1):167–174, 2022.
- [57] M. et al. Tranzatto. CERBERUS: Autonomous legged and aerial robotic exploration in the tunnel and urban circuits of the DARPA subterranean challenge. *Journal of Field Robotics*, 2:274–324, 2022.
- [58] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.
- [59] M. J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019.
- [60] W. Zhang, M. Togonon, L. Ott, R. Siegwart, and J. Nieto. Active model learning using informative trajectories for improved closed-loop control on real robots. In *Proc. IEEE Conf. on Robotics and Automation*, 2021.

APPENDIX

A. Proofs of Theorem 1 and of Lemma 1

Proof of Theorem 1: Thanks to (A1)-(A4), the solution to (SDE) is well-defined [6, 36] and all functions of interest are measurable (see [40, Appendix A.2]). In particular, the map $\omega \in \Omega \mapsto t + \alpha^{-1} \max(\sup_{s \in [0, T]} G(x_u(s, \omega), \xi(\omega)) - t, 0) \in \mathbb{R}$ is measurable for all $(u, t) \in \mathcal{U} \times \mathbb{R}$ since $z \mapsto \max(z, 0)$ is continuous). Further, using Kolmogorov's Lemma [36, Theorem 2.9], one can show that up to a modification, the map from controls $u \in \mathcal{U}$ to state trajectories x_u is Hölder continuous (for the uniform norm on the space of continuous functions $C([0, T], \mathbb{R}^n)$) with probability one, see [40, Appendix E].

Thanks to (A2) and since the max is 1-Lipschitz continuous, the functions G and H are Lipschitz continuous. Thus, the maps $(u, t) \mapsto t + \alpha^{-1} \max(\sup_{s \in [0, T]} G(x_u(s), \xi) - t, 0)$ and $u \mapsto H(x_u(T))$ are Hölder continuous with probability one.

Thanks to (A2), the optimal risk parameter t^* is achieved in a bounded interval $\mathcal{T} \subset \mathbb{R}$, see [54]. Thus, thanks to (A3), **OCP** is a stochastic program over a compact subset of \mathbb{R}^{z+1} .

The conclusion follows from [40, Theorem 5.1]. \blacksquare

The proof of Lemma 1 relies on a concentration inequality from [40]. Let $(\Omega, \mathcal{G}, \mathbb{P})$ be a probability space, $d \in \mathbb{N}$, $\mathcal{U} \subset \mathbb{R}^d$ be a compact set, $h : \mathcal{U} \times \Omega \rightarrow \mathbb{R}$ be Carathéodory (i.e., $h(u, \cdot)$ is \mathcal{G} -measurable for all $u \in \mathcal{U}$ and $h(\cdot, \omega)$ is continuous \mathbb{P} -almost-surely), and consider the following two assumptions.

(B1) \mathbb{P} -almost-surely, the map $u \mapsto h(u, \omega)$ is $\tilde{\alpha}$ -Hölder continuous for some exponent $\tilde{\alpha} \in (0, 1]$ and Hölder constant $M(\omega)$ satisfying $\mathbb{E}[M(\cdot)^2] < \infty$, such that

$$|h(u_1, \omega) - h(u_2, \omega)| \leq M(\omega) \|u_1 - u_2\|_2^{\tilde{\alpha}} \quad \forall u_1, u_2 \in \mathcal{U}.$$

(B2) For some $\bar{h} < \infty$, \mathbb{P} -almost-surely, $\sup_{u \in \mathcal{U}} |h(u, \omega)| \leq \bar{h}$.

Let $\{\omega^i\}_{i=1}^M$ be $M \in \mathbb{N}$ independent and identically distributed (iid) samples of ω . We have the following result.

Lemma 2 (Concentration Inequality [40]). *Assume that h satisfies (B1) and (B2). Let $D = 2 \sup_{u \in \mathcal{U}} \|u\|_2$, $C = 256$, $u_0 \in \mathcal{U}$, Σ_0 denote the covariance matrix of $h(u_0, \cdot)$, and*

$$\tilde{C} = \left(CD^{\frac{\tilde{\alpha}+1}{2}} d^{\frac{1}{2}} \mathbb{E}[M^2]^{\frac{1}{2}} \tilde{\alpha}^{-\frac{1}{2}} + \text{Trace}(\Sigma_0)^{\frac{1}{2}} \right). \quad (26)$$

Let $\epsilon > 0$, $\beta \in (0, 1)$, and the sample size $M \in \mathbb{N}$ be such that $M \geq \epsilon^{-2} (\tilde{C} + \bar{h} (2 \log(1/\beta))^{\frac{1}{2}})^2$. Then, with \mathbb{P} -probability at least $1 - \beta$ over the M iid samples ω^i ,

$$\sup_{u \in \mathcal{U}} \left| \frac{1}{M} \sum_{i=1}^M h(u, \omega^i) - \mathbb{E}[h(u, \cdot)] \right| \leq \epsilon. \quad (27)$$

Lemma 2 provides a high-probability finite-sample error bound for the sample average approximation of the expected value of $h(u, \cdot)$ that holds uniformly over all $u \in \mathcal{U}$.

To prove Lemma 1, we define the map $Z : \mathcal{U} \times \Omega \rightarrow \mathbb{R}$ by

$$Z_u(\omega) = \sup_{s \in [0, T]} G(x_u(s, \omega), \xi(\omega)), \quad (28)$$

which is bounded and Hölder continuous under assumptions (A1)-(A4). Then, given a large-enough compact set $\mathcal{T} \subset \mathbb{R}$, we define the measurable map $g : (\mathcal{T} \times \mathcal{U}) \times \Omega \mapsto \mathbb{R}$,

$$g((t, u), \omega) \mapsto t + \alpha^{-1} \max(Z_u(\omega) - t, 0). \quad (29)$$

Since g is a composition of Hölder continuous functions and \mathcal{T} is bounded, g is also bounded and Hölder continuous, i.e., g satisfies assumptions (B1) and (B2) over $(t, u) \in \mathcal{T} \times \mathcal{U}$.

Corollary 1 (Finite-Sample Error Bound). *Given M iid samples $\omega^i \in \Omega$, formulate the sample average approximation **SOCP** $_M(\bar{\omega})$ of **OCP**. Define the map g as in (29). Let $\epsilon > 0$ and $\beta \in (0, 1)$. Then, under assumptions (A1)-(A4), assuming $M \geq \epsilon^{-2} (\tilde{C} + \bar{h} (2 \log(1/\beta))^{\frac{1}{2}})^2$ for (\tilde{C}, \bar{h}) large-enough,*

$$\sup_{(t, u) \in \mathcal{T} \times \mathcal{U}} \left| \frac{1}{M} \sum_{i=1}^M g((t, u), \omega^i) - \mathbb{E}[g((t, u), \cdot)] \right| \leq \epsilon \quad (30)$$

with \mathbb{P} -probability at least $(1 - \beta)$.

Proof: Under assumptions (A1)-(A4), g satisfies Assumptions (B1)-(B2). The conclusion follows from Lemma 2. \blacksquare

The desired result (Lemma 1) follows from Corollary 1.

Proof of Lemma 1: For any $u \in \mathcal{U}$, define the random variable $Z_u(\omega) = (28)$, so that

$$\text{AV@R}_\alpha \left(\sup_{s \in [0, T]} G(x_{u(\bar{\omega})}(s), \xi) \right) = \text{AV@R}_\alpha (Z_{u(\bar{\omega})}).$$

Then, given an interval \mathcal{T} large-enough containing the optimal risk parameter t^* of **OCP** (see the proof of Theorem 1 and [54]), define the function $g : (\mathcal{T} \times \mathcal{U}) \times \Omega \mapsto \mathbb{R}$ as in (29). Denote the solution to **SOCP** $_M(\bar{\omega})$ by $(t(\bar{\omega}), u(\bar{\omega}))$ and

$$g((t(\bar{\omega}), u(\bar{\omega})), \omega) = t(\bar{\omega}) + \alpha^{-1} \max(Z_{u(\bar{\omega})}(\omega) - t(\bar{\omega}), 0).$$

Since $(t(\bar{\omega}), u(\bar{\omega}))$ solves **SOCP** $_M(\bar{\omega})$,

$$\begin{aligned} \frac{1}{M} \sum_{i=1}^M g((t(\bar{\omega}), u(\bar{\omega})), \omega^i) \\ = t(\bar{\omega}) + \frac{1}{\alpha M} \sum_{i=1}^M \max(Z_{u(\bar{\omega})}(\omega^i) - t(\bar{\omega}), 0) \leq 0. \end{aligned}$$

By Corollary (1), with probability at least $1 - \beta$,

$$\sup_{(t, u) \in \mathcal{T} \times \mathcal{U}} \left| \frac{1}{M} \sum_{i=1}^M g((t, u), \omega^i) - \mathbb{E}[g((t, u), \cdot)] \right| \leq \epsilon.$$

From the last two inequalities², with probability at least $1 - \beta$,

$$\mathbb{E}[g((t(\bar{\omega}), u(\bar{\omega})), \cdot)] \leq \epsilon.$$

We obtain that

$$\begin{aligned} \text{AV@R}_\alpha(Z_{u(\bar{\omega})}) &= \inf_{t \in \mathbb{R}} \left(t + \alpha^{-1} \mathbb{E}[\max(Z_{u(\bar{\omega})} - t, 0)] \right) \\ &\leq t(\bar{\omega}) + \alpha^{-1} \mathbb{E}[\max(Z_{u(\bar{\omega})} - t(\bar{\omega}), 0)]. \\ &= \mathbb{E}[t(\bar{\omega}) + \alpha^{-1} \max(Z_{u(\bar{\omega})} - t(\bar{\omega}), 0)]. \\ &= \mathbb{E}[g((t(\bar{\omega}), u(\bar{\omega})), \cdot)] \\ &\leq \epsilon \end{aligned}$$

with probability at least $1 - \beta$, concluding the proof. \blacksquare

²Assuming $t(\bar{\omega}) \in \mathcal{T}$, which can be enforced by restricting the search of solutions to **SOCP** $_M(\bar{\omega})$ to the compact set $\mathcal{U} \times \mathcal{T}$ with \mathcal{T} arbitrarily large.

B. Smooth reformulation (18) of AV@R constraint in $\widehat{\text{SOCP}}$

$\widehat{\text{SOCP}}$ is a deterministic non-convex program. Due to the max operation, its inequality constraint (17b) from the AV@R constraint is not smooth even if each term in G is smooth. Many optimization tools use gradient information and assume that constraints are twice differentiable, which is not the case with (17b). This constraint can be reformulated by introducing M variables $y_i \in \mathbb{R}$ and reformulating (17b) as follows:

$$\begin{aligned}
 t + \frac{1}{\alpha M} \sum_{i=1}^M \max \left(\max_{k=0, \dots, S} G(x_u^i(k\Delta t), \xi^i) - t, 0 \right) &\leq 0 \\
 &\iff \\
 \begin{cases} (M\alpha)t + \sum_{i=1}^M y_i \leq 0 \\ y_i \geq 0 & \forall i = 1, \dots, M \\ y_i \geq \max_{k=0, \dots, S} G(x_u^i(k\Delta t), \xi^i) - t & \forall i = 1, \dots, M \end{cases} & \\
 &\iff \\
 \begin{cases} (M\alpha)t + \sum_{i=1}^M y_i \leq 0 \\ y_i \geq 0 & \forall i = 1, \dots, M \\ y_i \geq \max_{k=0, \dots, S} \max_{j=1, \dots, N} G_j(x_u^i(k\Delta t), \xi^i) - t & \forall i = 1, \dots, M \end{cases} & \\
 &\iff \\
 \begin{cases} (M\alpha)t + \sum_{i=1}^M y_i \leq 0 \\ 0 \leq y_i & \forall i = 1, \dots, M \\ G_j(x_u^i(k\Delta t), \xi^i) - t \leq y_i & \begin{cases} \forall i = 1, \dots, M \\ \forall j = 1, \dots, N \\ \forall k = 0, \dots, S. \end{cases} \end{cases} & \quad (18)
 \end{aligned}$$

In contrast to (17b), the set of constraints (18) is smooth if every G_j is smooth. With (18) and the additional variable $y = (y_1, \dots, y_M)$, we reformulate $\widehat{\text{SOCP}}$ as follows:

$$\begin{aligned}
 &\widehat{\text{SOCP}}_M(\bar{\omega}) \\
 &\inf_{u \in \mathcal{U}, t \in \mathbb{R}, y \in \mathbb{R}^M} (17a)(u) \text{ such that } (17c), (17d), (17e), (18).
 \end{aligned}$$

$\widehat{\text{SOCP}}$ can be solved using off-the-shelf optimization tools.

C. Additional implementation details and results

We open-source code to reproduce the experiments in Section VIII at <https://github.com/StanfordASL/RiskAverseTrajOpt>. Our implementation is in Python and uses JAX [9] to evaluate constraints gradients. To solve $\widehat{\text{SOCP}}$, for the drone planning and autonomous driving problems, we use a standard sequential convex programming (SCP) scheme. We use OSQP [55] to solve the convexified problems at each SCP iteration. For the hopper system, we solve $\widehat{\text{SOCP}}$ using IPOPT [58] due to its improved numerical stability compared to SCP.

1) *Drone planning problem*: The computation time generally depends on the choice of solver, desired accuracy, and use case. For example, using our proposed approach within a risk-aware MPC controller would allow warm-starting for faster convergence, see also [17]. In this work, we use $\bar{u}_s = 0$ as a zero initial guess, disable the obstacle avoidance risk constraint for the first two SCP iterations to obtain more robust convergence, set the error tolerance threshold for OSQP to 10^{-3} , and use the normalized L^2 -error over control iterates $\|u^k - u^{k-1}\|/\|u^k\|$ to detect the convergence of SCP. We

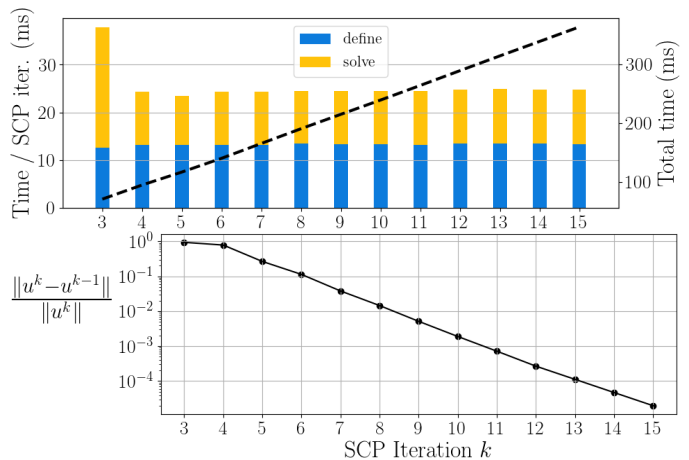


Fig. 6: Drone planning problem: median computation time and SCP iteration error statistics over 30 experiments ($\alpha = 0.05$, $M = 30$).

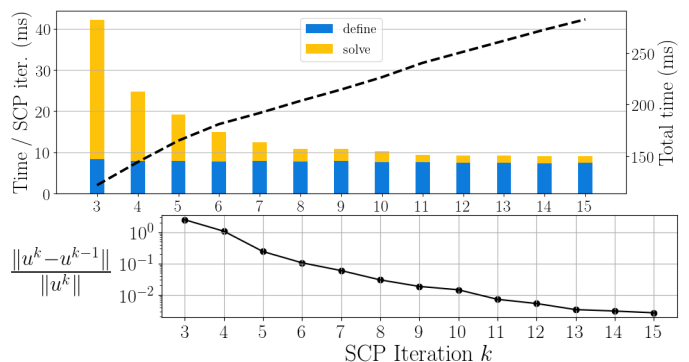


Fig. 7: Autonomous driving planning problem. Median statistics over 30 experiments ($\alpha = 0.02$, $M = 50$). Top plots: computation times and SCP iteration errors. Right plots: computation times for 10 SCP iterations and different risk parameters α .

report computation times and accuracy statistics in Figure 6, evaluated on a laptop with an i7-10710U CPU (1.10 GHz) and 16 GB of RAM. We observe that 10 SCP iterations are sufficient to obtain accurate results. Albeit our implementation is written in Python, computation times are reasonable and amenable to real-time applications. By warm-starting within an MPC loop, one could achieve a replanning frequency of 30 Hz using $M = 30$ samples and by returning the solution after a single SCP iteration. Finally, computation time scales roughly linearly in the number of samples. Parallelization (e.g., on a GPU) would enable using a larger numbers of samples M while retaining speed, albeit our results show that a small number of samples is sufficient to obtain feasible solutions.

2) *Autonomous driving problem*: Computation times and SCP statistics are reported in Figure 7. Results are comparable to those for the drone planning problem, again demonstrating the real-time capabilities of the approach. Computation times do not appear to be sensitive to the risk parameter α .

3) *Hopper robot problem*: We consider the hopper robot in [35] with $x = (q, \dot{q})$ where $q = (p_x, p_z, \theta, r) \in \mathbb{R}^4$, $u = (\tau, f) \in \mathbb{R}^2$. The system follows deterministic dynamics

$$M\ddot{q}(t) + C = B(q(t))u(t) + J_c(q(t))^T \lambda(t), \quad x(0) = x^0, \quad (24)$$

TABLE III: Hopper system. Monte-Carlo validation for 10^5 samples of the terrain parameter $\mu(q, \omega)$. We report the percentage of failures (the number of times the no-slip constraint in (25c) is violated), the empirical AV@R $_{\alpha}$ value, and the final jump distance $p_x(T)$.

		$\alpha = 5\%$	$\alpha = 10\%$	$\alpha = 20\%$	$\alpha = 50\%$
SAA $_{\alpha}$	constraints violations	0.5%	1.2%	2.2%	7.1%
	AV@R $_{\alpha}$	-0.06	-0.08	-0.09	-0.10
	$p_x(T)$	0.58	0.61	0.64	0.69
Baseline	constraints violations	50.2%			
	$p_x(T)$	1.10			

where $M \in \mathbb{R}^{4 \times 4}$ is the inertia matrix, $C \in \mathbb{R}^4$ comprises of Coriolis and conservative forces, $B(q(t)) \in \mathbb{R}^{4 \times 2}$ is the control matrix, $J_c(q(t)) \in \mathbb{R}^{4 \times 2}$ is the contact Jacobian, and $\lambda = (\lambda_x, \lambda_z) \in \mathbb{R}^2$ are contact forces that must satisfy (25).

Monte-Carlo validation of the satisfaction of the risk constraint for the trajectories represented in Figure 5 (obtained with a sample size $M = 30$) are provided in Table III.

D. Chance-constrained baseline with risk allocation

The risk-averse baseline used in the comparisons tackles the discrete-time joint-chance-constrained problem

$$\inf_{u \in U^S} \mathbb{E} \left[\sum_{k=0}^{S-1} \ell(x_k, u_k) + \varphi(x_S) \right] \quad (31a)$$

$$\text{s.t. } \mathbb{P} \left(\max_{k=1, \dots, S} G(x_k, \xi) > 0 \right) \leq \alpha, \quad (31b)$$

$$\mathbb{E}[H(x_S)] = 0, \quad (31c)$$

$$x_{k+1} = b(x_k, u_k, \xi) \Delta t + \sigma(x_k, u_k, \xi) w_k, \quad (31d)$$

where $G(x_k, \xi) = \max_{j=1, \dots, N} G_j(x_k, \xi)$ (see Section IV), $\Delta t = T/S$, $U^S = U \times \dots \times U$ (S times), and each w_k is (iid) Gaussian-distributed of mean 0 and variance Δt .

To conservatively reformulate the joint chance constraint (31b), this baseline uses Boole's inequality, which states that

$$\mathbb{P} \left(\max_{k=1, \dots, S} G(x_k, \xi) > 0 \right) \leq \sum_{k=1}^S \sum_{j=1}^N \mathbb{P}(G_j(x_k, \xi) > 0). \quad (32)$$

Thus, by introducing NS variables $\alpha_{jk} \in (0, 1)$, the optimization problem in (31) can be replaced by the more conservative optimization problem with *pointwise* chance constraints

$$\inf_{(u_k, \alpha_k)} \quad (31a) \quad \text{s.t.} \quad (31c), (31d), \quad (33a)$$

$$\mathbb{P} \left(G_j(x_k, \xi) > 0 \right) \leq \alpha_{jk} \quad \left(\begin{array}{l} \forall j = 1, \dots, N \\ \forall k = 1, \dots, S \end{array} \right) \quad (33b)$$

$$\alpha_{jk} > 0 \quad \left(\begin{array}{l} \forall j = 1, \dots, N \\ \forall k = 1, \dots, S \end{array} \right) \quad (33c)$$

$$\sum_{k=1}^S \sum_{j=1}^N \alpha_{jk} \leq \alpha. \quad (33d)$$

To approximately solve this problem (33), we use the standard method described next. The state trajectory is modeled as Gaussian-distributed ($x_k \sim \mathcal{N}(\mu_k, \Sigma_k)$) with mean and covariance trajectories computed approximately with a linearization

approach [38]. Since the terms $G_j(x_k, \xi)$ correspond to collision avoidance constraints with convex obstacles in the drone and driving problems, we linearize the terms $G_j(x_k, \xi)$ in x_k , which enables reformulating the corresponding pointwise chance constraints in (33b) using the $(1 - \alpha_k)$ -th quantiles of the inverse cumulative function of the standard normal distribution [38, Equations (5) and (26)]. The resulting program is solved using off-the-shelf optimization techniques.

As discussed in Remark 2 and in details in [22, 32], this baseline is conservative due to the use of Boole's inequality in (32). Using this inequality, this baseline neglects correlations of uncertainty. As such, the conservatism of this baseline increases with the number S of discretization timesteps [22, 32].