Evaluating Trajectory Collision Probability through Adaptive Importance Sampling for Safe Motion Planning

Edward Schmerling, Marco Pavone

***Abstract*— This paper presents a tool for addressing a key component in many algorithms for planning robot trajectories under uncertainty: evaluation of the safety of a robot whose actions are governed by a closed-loop feedback policy near a nominal planned trajectory. We describe an adaptive importance sampling Monte Carlo framework that enables the evaluation of a given control policy for satisfaction of a probabilistic collision avoidance constraint which also provides an associated certificate of accuracy. In particular this adaptive technique is well-suited to addressing the complexities of rigid-body collision checking applied to non-linear robot dynamics. As a Monte Carlo method it is amenable to parallelization for computational tractability, and is generally applicable to a wide gamut of simulatable systems, including alternative noise models. Numerical experiments demonstrating the effectiveness of the adaptive importance sampling procedure are presented and discussed.**

I. Introduction

This paper addresses the problem of online planning of robot trajectories under uncertainty *and* performance/safety constraints. This problem has been recognized recently as a critical component towards deploying autonomous robotic systems (e.g., drones, self-driving cars, surgical robots, autonomous spacecraft, etc.) in unstructured environments [1], [2], [3], [4]. Conceptually, to enable an autonomous system (AS) to plan its actions under uncertainty (e.g., with respect to environment characterization), one needs to design a strategy (i.e., closed-loop policy) for a decision maker [5], a computationally expensive procedure in general [6]. Instead, a promising approach is to pose the decision-making problem as an optimization over a simpler class of *nominal* policies (or even open-loop action sequences), which are then evaluated via closed-loop predictions under the assumption that a local feedback control law strives to ensure nominal behavior [3], [6], [7], [8].

To illustrate this concept, inspired by recent results within the domain of Model Predictive Control (MPC) [9], and to motivate our work, let the dynamics of a robot be given by

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_{t-1} + \mathbf{v}_t^u) + \mathbf{v}_t^x, \quad \mathbf{z}_t = \mathbf{h}(\mathbf{x}_t) + \mathbf{w}_t, \quad (1)$$

where $\mathbf{x}_t \in \mathbb{R}^{d_x}$ is the state, $\mathbf{u}_t \in \mathbb{R}^{d_u}$ is the control input, $\mathbf{z}_t \in \mathbb{R}^{d_z}$ is the measurement, and $\mathbf{v}_t^u$, $\mathbf{v}_t^x$, and $\mathbf{w}_t$ represent control and propagation (together, process) and measurement noise, respectively. Let $\mathcal{X}_{\text{obs}}$ be the infeasible space (that is, the set of states that violate a given set of constraints), so that $\mathcal{X}_{\text{feas}} := \mathbb{R}^d \backslash \mathcal{X}_{\text{obs}}$ is the feasible space. Let $\mathcal{X}_{\text{goal}} \subset \mathcal{X}_{\text{feas}}$ and $\mathbf{x}_0 \in \mathcal{X}_{\text{feas}}$ be the goal region and initial state, and let π_t denote a feedback policy mapping the information available up to time t (i.e., the history of measurements and control actions) to a control action $\mathbf{u}_t$. Given a trajectory cost measure c and letting $\overline{\mathbf{x}_0, \dots, \mathbf{x}_T}$ denote the robot's random trajectory, the problem is

Closed-Loop Belief Space Planning:

$$\begin{aligned} \min_{\{\pi_t\}_t} \quad & \mathbb{E}\left[c(\overline{\mathbf{x}_0, \dots, \mathbf{x}_T})\right] \\ \text{s.t.} \quad & \mathbb{P}\left(\overline{\mathbf{x}_0, \dots, \mathbf{x}_T} \cap \mathcal{X}_{\text{obs}} \neq \varnothing\right) \leq \alpha \\ & \mathbb{P}\left(\mathbf{x}_T \in \mathcal{X}_{\text{goal}}\right) \geq 1 - \alpha \\ & \mathbf{x}_0 \sim \mathcal{N}(\mathbf{x}_0^{\text{nom}}, P_0) \\ & \text{Equation (1).} \end{aligned} \quad (2)$$

Problem (2) is, generally, intractable as it requires optimizing over output-feedback policies (i.e., the policies $\{\pi_t\}_t$) – a POMDP problem that, even for relatively small-scale problems, might be not compatible with the tempo

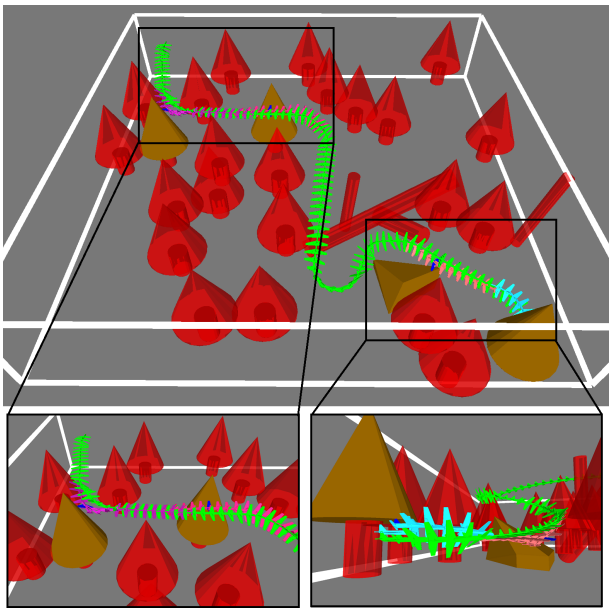


Fig. 1. Four likely collision modes for an airplane tracking a nominal trajectory (green) using a discrete time LQG controller. Tracking this nominal trajectory incurs an obstacle collision probability of 0.4%, which may be efficiently evaluated along with a tight associated confidence interval using variance-reduced Monte Carlo techniques (for this example in under 2 seconds to high confidence, a fraction of the 13 second trajectory duration).

Edward Schmerling is with the Institute for Computational & Mathematical Engineering, Stanford University, Stanford, CA 94305, schmrlng@stanford.edu.

Marco Pavone is with the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305, pavone@stanford.edu.

This work was supported by a Qualcomm Innovation Fellowship and by NASA under the Space Technology Research Grants Program, Grant NNX12AQ43G.

of AS-environment interaction. According to our previous discussion, the approach that we consider in this paper (see, e.g., [3], [6], [7], [8]) is to optimize over a *simpler* class of policies, e.g., obtained by combining a nominal control sequence $\{\mathbf{u}_t^*\}_t$ (yielding a nominal trajectory $\{\mathbf{x}_t^*\}_t$) with a "feedback correction" term that strives to ensure nominal behavior. For example, one can consider the set of affine feedback policies $\mathbf{u}_t := \mathbf{u}_t^* + \bar{\mathbf{u}}_t$, where $\mathbf{u}_t^*$ is a nominal control action and $\bar{\mathbf{u}}_t$ is a feedback correction term minimizing deviations from $\{\mathbf{x}_t^*\}_t$. The problem then becomes,

Belief Space Planning with Closed-Loop predictions:

$$\begin{aligned} \min_{\{\mathbf{u}_t^*\}_t} \quad & c(\overline{\mathbf{x}_0^*, \ldots, \mathbf{x}_T^*}) \\ \text{s.t.} \quad & \mathbf{u}_t = \mathbf{u}_t^* + \bar{\mathbf{u}}_t \\ & \mathbb{P}\left(\overline{\mathbf{x}_0, \ldots, \mathbf{x}_T} \cap \mathcal{X}_{\text{obs}} \neq \varnothing\right) \leq \alpha \\ & \mathbb{P}\left(\mathbf{x}_T \in \mathcal{X}_{\text{goal}}\right) \geq 1 - \alpha \\ & \mathbf{x}_0 \sim \mathcal{N}(\mathbf{x}_0^{\text{nom}}, P_0) \\ & \text{Equation (1).} \end{aligned} \tag{3}$$

Problem (3) is computationally much more tractable as it only involves optimization over open-loop sequences (i.e., $\{\mathbf{u}_t^*\}_t$), and represents a compromise between a POMDP formulation involving a minimization over the class of output-feedback control laws, and an open-loop formulation, in which the state is assumed to evolve in an open loop (i.e., it is conservatively assumed that over the future horizon no effort will be made to compensate for disturbances or new events).

This approach, while very appealing from a computational standpoint, poses a number of challenges, chiefly that of quickly and reliably evaluating the probability of constraint violation (henceforth, we will focus on collision avoidance and refer to such probability as "collision probability" – CP – with the understanding that the constraints in (3) can also model performance or other safety requirements). This problem is difficult as safety/performance constraints do not usually possess an additive structure, i.e., they can not be expressed as the expectation of a summation of a set of random variables. This is typical, for example, for chance constraints for obstacle avoidance as in Problem (3), and a fortiori for more complex constraints (expressed, e.g., as logical formulas). The presence of non-linear dynamics and the possible need of considering non-Gaussian noise models further compounds the difficulty of the problem. Within this context, the goal of this paper is to design an algorithm for the *the fast and reliable evaluation of collision probabilities*. We take particular consideration of non-linear state space dynamics with full translational and rotational rigid-body collision checking (as opposed to point-robot-based projections of workspace obstacles into $\mathcal{X}_{\text{obs}}$).

Related Work: Evaluating a controller's chance of fulfilling safety/performance constraints while guiding a robot along a reference trajectory has hitherto primarily been considered in an approximate fashion. As discussed in [6], most previous methods essentially rely on two approaches. In the first approach (see, e.g., [10], [11], [12], [13]), referred to as the "additive approach," a trajectory CP is approximately evaluated by using Boole's inequality (i.e., $\mathbb{P}\left(\cup_i A_i\right) \leq \sum_i \mathbb{P}\left(A_i\right)$) and *summing* pointwise CPs at a certain number of waypoints along the reference trajectory. In contrast, in the second approach (see, e.g., [3], [8], [14]), referred to as the "multiplicative approach," a trajectory CP is approximately evaluated by *multiplying* the complement of pointwise CPs at a certain number of waypoints along the reference trajectory. In a nutshell, the additive approach treats waypoint collisions as mutually exclusive, while the multiplicative approach treats them as independent. Since neither mutual exclusivity nor independence hold in general, nor is any accounting made for continuous collision checking between waypoints (although [8] does consider the case of general collision geometries which in some cases may be amenable to such discrete collision checking), such approaches can be off by large multiples and hinder the computation of feasible trajectories [6]. Even worse, it was shown in [6] that such approaches are asymptotically tautological, i.e., as the number of waypoints approaches infinity, a trajectory CP is approximated with a number *greater than or equal to one*, contrary to what one might expect from a refinement procedure. (We note, however, that such approximations would still be very useful in those cases where the problem is unconstrained and the sole objective is to optimize safety/performance, as what is needed in such cases is the characterization of *relative* CPs among trajectories. However, in a constrained setting, one needs an accurate, *absolute* measure of CP.) The work of [7] introduced a first-order approximate correction to the independence assumption of the multiplicative approach that appears empirically to mitigate its aforementioned drawbacks, but can still be off by a considerable margin [6] and can both under-approximate or over-approximate the true CP.

The limitations of the above approximation schemes motivated the approach in [6] to consider variance-reduced Monte Carlo (MC) to estimate trajectory CPs. In theory, the exact CP can be computed up to arbitrary accuracy by simulating a large number of trajectories and counting the number that violate a constraint. When the CP has a low value (e.g., $\leq 1\%$), however, as is desired for most robotic applications, an enormous number of Monte Carlo samples may be required to achieve confidence in the estimate. The work in [6] demonstrates statistical variance-reduction techniques based on the methods of importance sampling (IS) and control variates (CV) that reduce the required number of samples to a few hundred per evaluation, amenable to real-time implementation (notably, the samples may be processed in parallel to achieve run times on the order of a few milliseconds [15]). However, these techniques were designed and validated only for linear time-invariant point-robot systems under the influence of Gaussian process and measurement noise. As both assumptions do not always hold in practice, one objective of this work is to extend the variance-reduced method introduced in [6] to the case of non-linear dynamics and rigid-body collision models. Furthermore, similar to the approximation heuristics mentioned above, the mixture

importance sampling algorithm proposed in [6] may also suffer in its ability to reduce variance as time discretization approaches zero (although it will still provide an unbiased, asymptotically-exact estimator) due to correlated mixture terms corresponding to close potential collision points [16].

Statement of Contributions: We consider a discrete LQG controller equipped with an EKF (as considered in [3], [7]) applied to the linearized dynamics and Gaussianized noise (first-and-second moment-matching) of a general problem setting as our closed-loop feedback policy. We consider state space dynamics that map into an $SE(3)$ (translation and rotation) configuration space for collision checking in an $\mathbb{R}^3$ workspace. We explicitly consider non-Gaussian noise distributions at each time step by including a noise term that acts directly on the controls.

The primary contribution of this paper is the demonstration of an adaptive importance sampling framework (whereby the sampling distribution is refined through the course of simulating sample trajectories) which enables accelerated Monte Carlo collision probability estimation in general, non-linear problem settings with rigid-body collision checking that admits a high degree of correlated collision modes. Unlike any aforementioned approximate approach, this method allows for *certified error margins* through confidence intervals constructed using a standard error estimate accompanying the CP estimate. We demonstrate its effectiveness on an 8-state, 3-input airplane model where it uses fewer MC samples than any alternative to accurately evaluate a policy with low associated CP (0.4%). A single-threaded MC implementation achieves usable estimates with certified accuracy within 1-2 seconds (< 500 samples). Other contributions of this paper include an improved method for identifying likely collision points based on a Newton method from [8] and a demonstration of why adaptive techniques are necessary for mixture importance sampling for robot CP. Given the successes of [6] in incorporating MC collision probability estimation within a near-real-time safe motion planning algorithm, and especially in light of this MC method's parallelization potential [15], we believe that the adaptive mixture importance sampling algorithm described in this paper represents a tool worthy for consideration in a wide variety of robotic applications that plan with safety constraints in operation.

Organization: The remainder of this paper is organized as follows: in Section II we review background material on importance sampling relevant to robot collision probability estimation including an adaptive technique, in Section III we define the problem dynamics, LQG controller model, and collision model. In Section IV we present how to tailor adaptive mixture IS to CP computation, and present a modified method of computing close obstacle points for constructing likely collision modes. Section V provides illustrative experiments with a rigid-body non-linear airplane model for the algorithms detailed in the previous sections, and Section VI contains conclusions and speculation on possible directions of future research.

II. Background Material

In this section we review the basics of importance sampling, a technique for reducing the variance of a Monte Carlo estimator, as well as an adaptive variant for mixture importance sampling that frames the selection of mixture weights as a convex optimization problem. Our discussion is tailored to the problem of computing trajectory CPs, where the high-dimensional nature of the noise (a joint distribution spanning the entire length of a trajectory) increases the challenge of selecting a good distribution.

A. Importance Sampling

Consider a random variable $\mathbf{X} \in \mathbb{R}^n$ distributed according to the probability density function (pdf) $P : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$. The expectation $\mathbb{E}[f(\mathbf{X})] = \int_{R^n} f(x)P(x)\,\mathrm{d}x$ of a function $f(\mathbf{X})$ may be rewritten as

$$\begin{aligned}\mathbb{E}^P[f(\mathbf{X})] &= \int_{R^n} f(x)P(x)\,\mathrm{d}x \\ &= \int_{R^n} \left(f(x)\frac{P(x)}{Q(x)}\right) Q(x)\,\mathrm{d}x \\ &= \mathbb{E}^Q\left[f(\mathbf{X})\tfrac{P(\mathbf{X})}{Q(\mathbf{X})}\right],\end{aligned} \tag{4}$$

where Q is an alternative pdf satisfying $Q(x) > 0$ for all $x \in \mathbb{R}^n$ where $f(x)P(x) \neq 0$, and $\mathbb{E}^P$ and $\mathbb{E}^Q$ denote expectations computed under the distributions $\mathbf{X} \sim P$ and $\mathbf{X} \sim Q$, respectively. The quantity $w(x) = P(x)/Q(x)$ is referred to as the likelihood ratio of the pdfs at x.

In the context of Monte Carlo estimation, the above computation implies that given independent and identically distributed (i.i.d.) samples $\{\mathbf{X}^{(i)}\}_{i=1}^m$ drawn from Q, the expectation $p := \mathbb{E}[f(\mathbf{X})]$ may be estimated by

$$\hat{p}^Q := \frac{1}{m}\sum_{i=1}^m f(\mathbf{X}^{(i)})w(\mathbf{X}^{(i)}). \tag{5}$$

The variance of the estimator $\hat{p}^Q$ may be estimated by

$$\hat{V}^Q := \frac{1}{m^2}\sum_{i=1}^m \left(f(\mathbf{X}^{(i)})w(\mathbf{X}^{(i)}) - \hat{p}^Q\right)^2. \tag{6}$$

In this paper, $\mathbf{X}$ will denote a vector of noise samples that act upon a robot at a sequence of discrete time increments, and $f(\mathbf{X})$ will denote the indicator function that the robot, guided by a trajectory-tracking controller, collides into the obstacle set given the trajectory noise sample $\mathbf{X}$. The robot's collision probability is $p = \mathbb{E}[f(\mathbf{X})]$.

We see from (4) that $\hat{p}^Q$ defined in (5) is an unbiased estimator of the CP p, that is, as long as the support of Q contains the support of P, the estimator $\hat{p}^Q$ converges in probability to p as $m \to \infty$ regardless of the choice of Q. Thus the selection criterion for the IS pdf Q is to minimize the variance of $\hat{p}^Q$. This has the practical effect of reducing the number of samples required before the estimator yields a useable result, which, as argued in [6], is critical for real-time accurate CP estimation for robotic applications with a very low collision chance constraint. Motivated by the sum in (5), we call $\mathbf{Var}^Q\left[f(\mathbf{X})\frac{P(\mathbf{X})}{Q(\mathbf{X})}\right]$ the per-sample variance

contributed by each i.i.d. sample $\mathbf{X}^{(i)} \sim Q$. Choosing Q to minimize the per-sample variance is equivalent [17] to minimizing the Rényi divergence

$$D_2\left(\pi^* \| Q\right) := \log \int_{R^n} \frac{\pi^*(x)^2}{Q(x)} \, \mathrm{d}x \tag{7}$$

where $\pi^*(x) := |f(x)|P(x) / \int_{R^n} |f(x)|P(x) \, \mathrm{d}x$ is itself the minimizer, provided the optimization is unconstrained.

Constructing and sampling from π^* is usually not possible in practice (for positive f, the normalization factor $\mathbb{E}^P[f(\mathbf{X})]$ is precisely the quantity we wish to estimate), but its form yields some insights. In the case that f is an indicator function, π^* has support on only the "important" parts of P where $f(x) = 1$; for our purposes, to sample from π^* is to sample only noise trajectories that lead to collision and weight them in the computation of $\hat{p}^{\pi^*}$ by their relative likelihood. This motivates the search for IS distributions Q that artificially inflate the occurrence of the rare event $f(\mathbf{X}) = 1$, but this should be accomplished while maintaining relative probability according to P lest the likelihood ratio P/Q be very large for some likely realization of the event, corresponding to a large value in the variance integral (7). Mathematically, we attempt to minimize $D_2\left(\pi^* \| Q_\theta\right)$ over a family of distributions $\mathcal{Q} = \{Q_\theta \mid \theta \in \Theta\}$, described by a finite vector of parameters θ, capable of capturing this aim.

B. Mixture Importance Sampling

As recognized in [6], there are typically multiple ways in which a noise-perturbed robot trajectory can collide with its surroundings. Although the noise pdf P is usually unimodal (corresponding to a robot centered on the nominal trajectory), an effective IS distribution may be multimodal (corresponding to the many ways the robot can drift into obstacles). This motivates the use of mixture IS distributions with pdfs of the form

$$Q_{(\alpha,\eta)}(x) = \sum_{d=1}^{D} \alpha_d q_d(x; \eta_d), \tag{8}$$

parameterized by $\theta = (\alpha, \eta)$; the α_d are nonnegative mixture weights such that $\sum_{d=1}^{D} \alpha_d = 1$ and the η_d are internal parameters of the component densities q_d. A special case of mixture IS relevant to robotic applications is defensive importance sampling, where the nominal distribution P is selected as one of the component distributions. If the pdfs q_d are selected as noise likely to lead to certain collision scenarios, including P serves as a catch-all to ensure that no unforeseen collision mode, e.g., due to the complex evolution of uncertainty distributions through non-linear dynamics, is left out.

C. Adaptive Mixture Importance Sampling

For robot collision probability estimation, optimizing the per-sample variance over a full family of distributions (8) is in general computationally intractable, especially if the component trajectory noise parameterization η_d is high-dimensional. Thus we consider instead the problem of selecting the weights α_d for fixed components q_d. As shown in [17], the objective $D_2\left(\pi^* \| Q_\alpha\right)$ is convex with respect to α, and is therefore amenable to online stochastic optimization methods with convergence guarantees. We reproduce in Algorithm 1 a stochastic mirror descent procedure from [17], designed for the simultaneous adaptation of mixture distribution weights alongside IS estimation. Algorithm 1 performs stochastic gradient descent on a set of mirrored variables $\tilde{\alpha}$ in order to enforce the probability constraints $\sum_{d=1}^{D} \alpha_d = 1, \ \alpha_d \geq 0 \ \forall d$. We present here the self-normalized versions of the final estimators, where in computing $\hat{p}$ we normalize by the sum of the sampled likelihood ratios. This is an alternative to normalizing, as in (5), by the reciprocal of the sample count $1/m$. Self-normalized importance sampling yields an asymptotically unbiased estimator, and in practice may further reduce variance when applying importance sampling techniques. We note that normalizing by the sample count $m = k\ell$ would only change the form of expressions in Alg. 1, Line 8.

Algorithm 1 Adaptive Mixture IS (Section 3.4.4, [17])

Require: Component densities $q_1(x), \ldots, q_D(x)$, initial mixture weights α^1, step size parameter C, batch size k, number of iterations ℓ

1: Set mirrored weights: $\widetilde{\alpha^1} = \log(\alpha^1)$
2: **for** $i = 1 : \ell$ **do**
3: Sample $\{\mathbf{X}_{i,j}\}_{j=1}^{k}$ from current IS distribution Q_{α^i}
4: Compute gradient:

$$g_i = -\frac{1}{k} \sum_{j=1}^{k} \frac{\left(f(\mathbf{X}_{i,j}) \frac{P(\mathbf{X}_{i,j})}{Q_{\alpha^i}(\mathbf{X}_{i,j})}\right)^2}{Q_{\alpha^i}(\mathbf{X}_{i,j})} \begin{bmatrix} q_1(\mathbf{X}_{i,j}) \\ \vdots \\ q_D(\mathbf{X}_{i,j}) \end{bmatrix}$$

5: Update mirrored weights: $\widetilde{\alpha^{i+1}} = \widetilde{\alpha^i} - (C/\sqrt{i}) g_i$
6: Set new mixture weights: $\alpha^{i+1} \propto \exp\left(\widetilde{\alpha^{i+1}}\right)$
7: **end for**
8: **return** Estimator, estimated variance of estimator [18]:

$$\hat{p}^{\text{AIS}} = \frac{\sum_{i=1}^{\ell} \sum_{j=1}^{k} f(\mathbf{X}_{i,j}) \frac{P(\mathbf{X}_{i,j})}{Q_{\alpha^i}(\mathbf{X}_{i,j})}}{\sum_{i=1}^{\ell} \sum_{j=1}^{k} \frac{P(\mathbf{X}_{i,j})}{Q_{\alpha^i}(\mathbf{X}_{i,j})}}$$

$$\hat{V}^{\text{AIS}} = \frac{1}{k\ell} \frac{\sum_{i=1}^{\ell} \sum_{j=1}^{k} \left(\frac{P(\mathbf{X}_{i,j})}{Q_{\alpha^i}(\mathbf{X}_{i,j})} (f(\mathbf{X}_{i,j}) - \hat{p}^{\text{AIS}})\right)^2}{\left(\sum_{i=1}^{\ell} \sum_{j=1}^{k} \frac{P(\mathbf{X}_{i,j})}{Q_{\alpha^i}(\mathbf{X}_{i,j})}\right)^2}$$

III. Problem Formulation

A. State Space Dynamics

As in the introduction, let the state space dynamics of a robot be given by

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_{t-1} + \mathbf{v}_t^u) + \mathbf{v}_t^x, \quad \mathbf{z}_t = \mathbf{h}(\mathbf{x}_t) + \mathbf{w}_t, \tag{9}$$

where $\mathbf{x}_t \in \mathbb{R}^{d_x}$ is the state, $\mathbf{u}_t \in \mathbb{R}^{d_u}$ is the control input, $\mathbf{z}_t \in \mathbb{R}^{d_z}$ is the measurement, $\mathbf{v}_t = [\mathbf{v}_t^u; \mathbf{v}_t^x]$ is the process noise (comprising an explicit control uncertainty $\mathbf{v}_t^u \sim \mathbf{V}_t^u$ in addition to a propagation uncertainty $\mathbf{v}_t^x \sim \mathbf{V}_t^x$), and $\mathbf{w}_t \sim$

$\mathbf{W}_t$ is the measurement noise at time t. We restrict our attention to independent noise distributions $\mathbf{V}_t^u$, $\mathbf{V}_t^x$, and $\mathbf{W}_t$ with zero mean and finite second moments; we note that cases of colored noise may be addressed through state augmentation. Let V_t^u, V_t^x, and W_t denote the covariance matrices of $\mathbf{V}_t^u$, $\mathbf{V}_t^x$, and $\mathbf{W}_t$, respectively. We are interested in estimating the collision probability that arises from tracking a nominal path $\mathcal{P}^* = \{\mathbf{x}_0^*, \mathbf{u}_0^*, \mathbf{x}_1^*, \mathbf{u}_1^*, \ldots, \mathbf{x}_T^*\}$ where $\mathbf{x}_t^* = \mathbf{f}(\mathbf{x}_{t-1}^*, \mathbf{u}_{t-1}^*)$ for $t = 1, \ldots, T$. The true initial state $\mathbf{x}_0$ satisfies $\mathbf{x}_0 = \mathbf{x}_0^* + \mathbf{p}_0$, where the initial state uncertainty $\mathbf{p}_0 \sim \mathbf{P}_0$ is drawn from a distribution with zero mean and covariance matrix P_0, and the controller uses an understanding of the dynamics and information from observations $\mathbf{z}_1, \cdots, \mathbf{z}_T$ to choose actions $\mathbf{u}_0, \ldots, \mathbf{u}_{T-1}$ from which the true state $\mathbf{x}_t$ evolves according to (9).

Similar to [3] and [7] we consider an LQR state-feedback controller equipped with an extended Kalman Filter for state estimation (together, LQG control) to control the deviation of a robot from a nominal trajectory. With deviation variables from $\mathcal{P}^*$ defined as $\bar{\mathbf{x}}_t = \mathbf{x}_t - \mathbf{x}_t^*$, $\bar{\mathbf{u}}_t = \mathbf{u}_t - \mathbf{u}_t^*$, and $\bar{\mathbf{z}}_t = \mathbf{z}_t - \mathbf{h}(\mathbf{x}_t^*)$, we linearize (9) according to:

$$\begin{aligned} \bar{\mathbf{x}}_t &= A_t \bar{\mathbf{x}}_{t-1} + B_t \bar{\mathbf{u}}_{t-1} + (B_t \mathbf{v}_t^u + \mathbf{v}_t^x) \\ &\quad + O(\|\bar{\mathbf{x}}_{t-1}\|^2 + \|\bar{\mathbf{u}}_{t-1}\|^2), \\ \bar{\mathbf{z}}_t &= H_t \bar{\mathbf{x}}_t + \mathbf{w}_t + O(\|\bar{\mathbf{x}}_t\|^2), \end{aligned} \tag{10}$$

where $A_t = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}_{t-1}^*, \mathbf{u}_{t-1}^*)$, $B_t = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}(\mathbf{x}_{t-1}^*, \mathbf{u}_{t-1}^*)$, and $H_t = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\mathbf{x}_t^*)$. The LQG controller maintains an estimate $\hat{\mathbf{x}}_t$ of the true state deviation $\bar{\mathbf{x}}_t$ using an extended Kalman filter

$$\hat{\mathbf{x}}_t = K_t \bar{\mathbf{z}}_t + (I - K_t H_t)(A_t \hat{\mathbf{x}}_{t-1} + B_t \bar{\mathbf{u}}_{t-1}),$$

where K_t is the Kalman gain matrix at time t corresponding to the linearized system (10) with noise covariance matrices P_0, $V_t = \begin{bmatrix} B_t V_t^u B_t^T & 0 \\ 0 & V_t^x \end{bmatrix}$, and W_t [19]. Then at each time $t = 0, \ldots, T-1$ the LQG controller applies the input $\mathbf{u}_t = \mathbf{u}_t^* + \bar{\mathbf{u}}_t$ with

$$\bar{\mathbf{u}}_t = \mathbf{u}_t^* + L_{t+1} \hat{\mathbf{x}}_t,$$

where L_t is the finite time horizon LQR feedback gain matrix corresponding to (10) with appropriately chosen state regulation and control effort penalties for the robotic application.

We emphasize that when simulating trajectories in this paper, we compute the exact state evolution (using an RK4 integration scheme) according to the true dynamics (9). However, we note here that with $\mathbf{y}_t = \left[\bar{\mathbf{x}}_t; \hat{\mathbf{x}}_t\right]$ we may write down an approximate system with linearized dynamics and Gaussian noise (moment-matched to the true noise up to second order):

$$\mathbf{y}_t = F_t \mathbf{y}_{t-1} + G_t \mathbf{q}_t, \quad \mathbf{q}_t \sim \mathcal{N}(\mathbf{0}, Q_t), \tag{11}$$

for appropriate choices of matrices F_t, G_t and Q_t. See [3] for a full derivation. In addition to evolving approximate state trajectories, this system may be used to evolve approximate pointwise uncertainty distributions of $\bar{\mathbf{x}}_t$ parameterized as multivariate Gaussians, as in [3], [6], [7]. Let Σ_t denote the a priori covariance of $\bar{\mathbf{x}}_t$ thus derived.

B. Configuration Space and Workspace Representations

In this work we consider rigid-body robots whose configuration $\mathbf{q}_t = \mathbf{q}(\mathbf{x}_t) \in SE(3)$, consisting of a 3D rotation and translation, is a deterministic function of the state. We represent both the robot, configured at $\mathbf{q}$, and the static obstacle set in the workspace as unions of convex components $\mathcal{R}(\mathbf{q}) = \bigcup_{i=1}^{r} \mathcal{R}_i(\mathbf{q}) \subset \mathbb{R}^3$ and $\mathcal{E} = \bigcup_{j=1}^{e} \mathcal{E}_j \subset \mathbb{R}^3$, respectively. Then the state space obstacle set is $\mathcal{X}_{\text{obs}} = \{\mathbf{x} \in \mathbb{R}^{d_x} \mid \mathcal{R}(\mathbf{q}(\mathbf{x})) \bigcap \mathcal{E} \neq \varnothing\}$. For a given robot state $\mathbf{x}$ and associated configuration $\mathbf{q}$, we assume access to a distance function $d_{i,j}(\mathbf{q})$ measuring the Euclidean separation between $\mathcal{R}_i(\mathbf{q})$ and $\mathcal{E}_j$. In the case that $\mathcal{R}_i(\mathbf{q})$ intersects $\mathcal{E}_j$, $d_{i,j}(\mathbf{q})$ returns a negative value corresponding to the maximum extent of penetration. We also assume access to the distance gradient $\partial d_{i,j}(\mathbf{q})/\partial \mathbf{q}$, either analytically or through finite differencing. We may also compute $\partial d_{i,j}(\mathbf{x})/\partial \mathbf{x} = \frac{\partial \mathbf{q}}{\partial \mathbf{x}} \frac{\partial d_{i,j}}{\partial \mathbf{q}}$.

C. Problem Statement

The problem we wish to solve in this paper is to devise an accurate, computationally-efficient algorithm equipped with an error estimate to estimate the CP

$$\mathbb{P}\left(\overline{\mathbf{x}_0, \ldots, \mathbf{x}_T} \cap \mathcal{X}_{\text{obs}} \neq \varnothing\right)$$

where $\overline{\mathbf{x}_0, \ldots, \mathbf{x}_T}$ denotes a continuous interpolation between states, and the state trajectory $\mathbf{x}_t$ is controlled via the control law $\mathbf{u}_t = \mathbf{u}_t^* + \bar{\mathbf{u}}_t$. As discussed in the introduction, the primary motivation of this problem is to enable belief space planning with closed-loop predictions for general non-linear problems with possibly non-Gaussian noise models.

A few comments are in order. First, in this paper we are not proposing a new planning algorithm, rather an algorithm that addresses one of the key bottlenecks for planning under uncertainty. Second, the method proposed here can be used in combination with a variety of planning frameworks, e.g., exhaustive evaluation of RRT plans [3], [8] or meta-algorithms as in [6]. Third, although for clarity Subsection III-B assumes an $SE(3)$ configuration space and $\mathbb{R}^3$ workspace, the methods in this paper may be readily generalized to other rigid-body robots, e.g., manipulators [8]. Finally, in stark contrast with alternative methods (with the exception of [6], which this work extends), we provide a computable error estimate that can be used as a certificate of accuracy for the trajectory's estimated CP.

IV. Adaptive Importance Sampling for Collision Probability Estimation

In this section we present an algorithm for the accurate, computationally-efficient estimation of a trajectory's tracking CP under non-linear dynamics and non-Gaussian noise models. Our approach is to select importance sampling distributions for CP estimation as a mixture of reparameterized copies of the actual process and measurement noise, corresponding to different modes of failure, similar to [6]. In the notation of Section II,

$$\begin{aligned} \mathbf{X} &= \left(\mathbf{p}_0, \mathbf{v}_1^u, \mathbf{v}_1^x, \mathbf{w}_1, \ldots, \mathbf{v}_T^u, \mathbf{v}_T^x, \mathbf{w}_T\right), \\ P(x) &= \mathbf{P}_0(p_0) \cdot \mathbf{V}_1^u(v_1^u) \cdot \ldots \cdot \mathbf{W}_T(w_T), \end{aligned}$$

Algorithm 2 Close Pairwise $\mathcal{X}_{\text{obs}}$ Point (adapted from [8])

Require: Nominal mean $\mathbf{x}^* \in \mathbb{R}^{d_x}$, covariance matrix $\Sigma \in \mathbb{R}^{d_x \times d_x}$, workspace distance function $d_{i,j}(\mathbf{q}(\mathbf{x}))$ between robot/environment pair of convex components $\mathcal{R}_i(\mathbf{q})$ and $\mathcal{E}_j$, linesearch parameter γ, tolerance $\epsilon > 0$

1: Set $\mathbf{x}_0 = \mathbf{x}^*, k = 0$
2: **repeat**
3: Newton step (derivatives evaluated at $\mathbf{x}_k, \mathbf{q}(\mathbf{x}_k)$):

$$\mathbf{x}_{k+1} = \mathbf{x}_k - d_{i,j}(\mathbf{x}_k)\Sigma\frac{\partial d_{i,j}}{\partial \mathbf{x}}(\mathbf{x}_k) \Big/ \left(\frac{\partial d_{i,j}}{\partial \mathbf{x}}(\mathbf{x}_k)^T \Sigma \frac{\partial d_{i,j}}{\partial \mathbf{x}}(\mathbf{x}_k)\right)$$

4: $k = k + 1$
5: **until** $\|\mathbf{x}_k - \mathbf{x}_{k-1}\| < \epsilon$
6: $m_k = (\mathbf{x}_k - \mathbf{x}^*)^T \Sigma^{-1} (\mathbf{x}_k - \mathbf{x}^*)$
7: **repeat**
8: With $\mathbf{g} = \frac{\partial d_{i,j}}{\partial \mathbf{x}}(\mathbf{x}_k)$, compute search direction:

$$\mathbf{s} = \Sigma^{-1}\mathbf{g} - \left(\frac{\mathbf{g}^T\Sigma^{-1}(\mathbf{x}_k - \mathbf{x}^*)}{\mathbf{g}^T\mathbf{g}}\right)\mathbf{g}$$

9: $\alpha = 1$
10: **repeat**
11: $\mathbf{x}_{k+1} = \text{project}_{i,j}\left(\mathbf{x}_k - \frac{\alpha\gamma m_k \mathbf{s}}{\mathbf{s}^T\Sigma^{-1}(\mathbf{x}_k - \mathbf{x}^*)}\right)$
12: $m_{k+1} = (\mathbf{x}_{k+1} - \mathbf{x}^*)^T\Sigma^{-1}(\mathbf{x}_{k+1} - \mathbf{x}^*)$
13: $\alpha = \alpha/2$
14: **until** $m_{k+1} \leq m_k$ or $\|\mathbf{x}_{k+1} - \mathbf{x}_k\| < \epsilon$
15: $k = k + 1$
16: **until** $m_k \geq m_{k-1}$ or $\|\mathbf{x}_k - \mathbf{x}_{k-1}\| < \epsilon$
17: **return** $\mathbf{x}_k$ with minimum corresponding m_k

(where we have assumed the independence of the noise distributions in the construction of this joint pdf) and $f(\mathbf{X})$ is the event that the noise random variable $\mathbf{X}$ gives rise to a colliding trajectory under LQG control. We consider IS distributions of the form

$$Q_{(\alpha,\eta)}(x) = \sum_{d=1}^{D} \alpha_d q_d(x; \eta_d)$$
$$q_d(x; \eta_d) = \mathbf{P}_0(p_0; \eta_d) \cdot \mathbf{V}_1^u(v_1^u; \eta_d) \cdot \ldots \cdot \mathbf{W}_T(w_T; \eta_d)$$

where η_d encodes all of the parameters required to specify the process and measurement noise distributions. For example, in the case of Gaussian noise, η_d consists of $(3T + 1)$ mean vectors and covariance matrices: one pair for the initial state uncertainty and for each process/measurement noise distribution at each time step $t = 1, \ldots, T$.

We choose each η_d to represent a likely tracking collision mode (see Figure 1). In particular, as in [6], we consider η_d derived as a shift in noise means resulting in an expected collision (under the linearized dynamics (11)) at an obstacle point $\mathbf{x}_{\text{obs}} \in \mathcal{X}_{\text{obs}}$ close to $\mathbf{x}_t^*$, measured by the Mahalanobis distance $\sqrt{(\mathbf{x}_t^* - \mathbf{x}_{\text{obs}})^T \Sigma_t^{-1} (\mathbf{x}_t^* - \mathbf{x}_{\text{obs}})}$. The problem of computing close obstacle points for a robot consisting of convex rigid-body components was previously considered in [8], which, for a pair $(\mathcal{R}_i, \mathcal{E}_j)$, proposes a Newton method for identifying a close point $\mathbf{x}_{\text{obs}}$ satisfying $d_{i,j}(\mathbf{q}(\mathbf{x}_{\text{obs}})) = 0$. We apply that method in Algorithm 2 to first identify a feasible point satisfying the zero-distance constraint, and then follow it with a standard non-linear constrained minimization phase to find a local optimum in Mahalanobis distance. This minimization phase employs a linesearch with a constraint projection subroutine ($\text{project}_{i,j}$ Alg. 2, Line 11) to ensure $d_{i,j}(\mathbf{q}(\mathbf{x}_{\text{obs}})) = 0$; we use a Newton method to implement that projection as well. In our experiments we find that local optimization can reduce $\mathbf{x}_{\text{obs}}$ Mahalanobis distance by $\sim 5\%$. In order to select the mean shifts for η_d corresponding to collision at $\mathbf{x}_{\text{obs}}$, we apply the maximum likelihood method described in [6], which amounts to a simple least-squares solve. We note that this choice admits an alternative interpretation of minimizing $D_2(P\|Q)$ subject to the constraint that $\mathbb{E}^Q[\bar{\mathbf{x}}_t] = \mathbf{x}_{\text{obs}} - \mathbf{x}_t^*$.

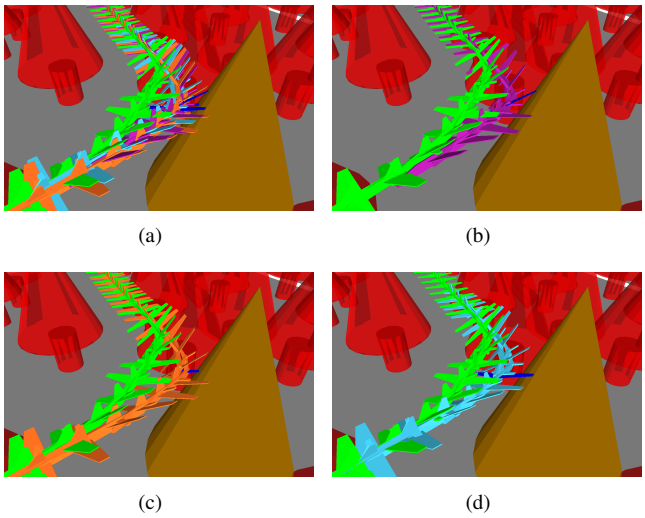


Fig. 2. The three most likely collision points, measured by Mahalanobis distance, for the nominal trajectory displayed in Figure 1. Figure 2(a) displays the high degree of correlation between the likely collisions 2(b) wing strike at $t = 82$, 2(c) stabilizer strike at $t = 83$, and 2(d) wing strike at $t = 83$. Convex robot components in collision are highlighted in blue.

The mixture distribution $Q_{(\alpha,\eta)}$ contains components corresponding to each of the top $D - 1$ most likely collisions computed over tuples $(\mathbf{x}_t, \mathcal{R}_i, \mathcal{E}_j)$, ordered by Mahalanobis distance (we set the final term $q_D = P$ to enable defensive importance sampling). The initial mixture weights α^1 may be set uniformly, or in proportion to the probability that a state sampled from a Gaussian estimate of the marginal deviation distribution at the relevant trajectory time step crosses a half-space associated with each close obstacle point [7], [8]. Figure 2 illustrates why adaptive mixture IS, Algorithm 1, is a necessary addition for refining these weights online to achieve the highest degree of variance reduction. For this plane trajectory, there is a high degree of correlation between similar collision events at successive time steps, or at the same time step between collisions involving different robot components. Unlike any heuristic method to address these correlations, which also grow worse with finer time discretization, stochastically solving the convex optimization problem of mixture weight selection is guaranteed to con-

verge to the optimal α^* as the sample size $m \to \infty$.

V. Numerical Experiments

A. Dynamics and Noise Model

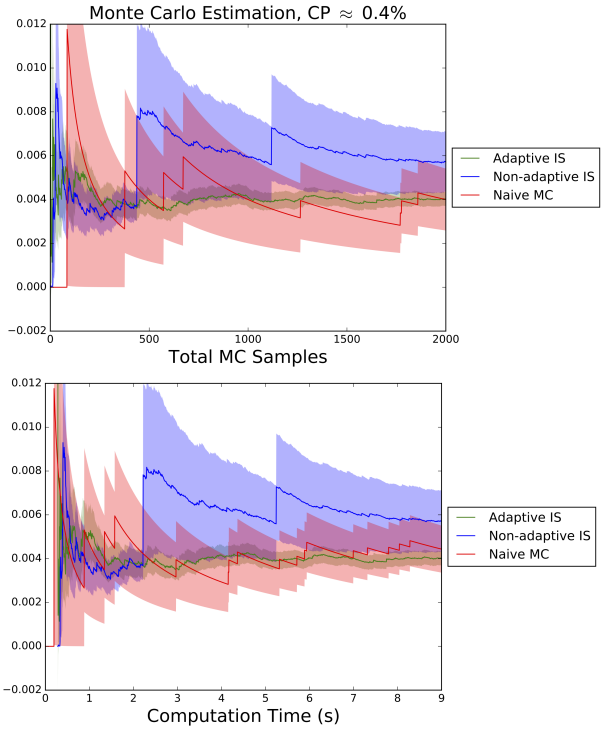
In this paper we consider discrete-time nominal dynamics for an airplane integrated from a simple continuous-time model [20] propagated under zero-order hold control inputs with a time step Δt. The continuous-time model is:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{v} \\ \dot{\psi} \\ \dot{\gamma} \\ \dot{\phi} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} v\cos(\psi)\cos(\gamma) \\ v\sin(\psi)\cos(\gamma) \\ v\sin(\gamma) \\ u_a - F_{\text{drag}}(v,\alpha)/m - g\sin(\gamma) \\ -F_{\text{lift}}(v,\alpha)\sin(\phi)/(mv\cos(\gamma)) \\ F_{\text{lift}}(v,\alpha)\cos(\phi)/(mv) - g\cos(\gamma)/v \\ u_{\dot{\phi}} \\ u_{\dot{\alpha}} \end{bmatrix}, \tag{12}$$

where x, y, z are position in a global frame, v is airspeed (we assume zero wind, aside from isotropic gusts represented by process noise), ψ is the course angle, γ is the flight path angle, ϕ is the roll angle, and α is the angle of attack. The control inputs $\mathbf{u} = (u_a, u_{\dot{\phi}}, u_{\dot{\alpha}})$ are longitudinal acceleration (due to engine thrust), roll rate, and pitch rate respectively. We assume a flat-plate airfoil model so that $F_{\text{lift}} = \pi\rho A v^2 \alpha$ and $F_{\text{drag}} = \rho A v^2 (C_{D_0} + 4\pi^2 K \alpha^2)$ [20] where gravity g, air density ρ, wing area A, plane mass m, drag coefficient C_{D_0}, and induced drag factor K are all constants. Using the Euler ZYX (yaw ψ, pitch θ, roll ϕ) rotation angle convention, the mapping from state space to configuration space is given by $\mathbf{q}(\mathbf{x}) = (x, y, z, \psi, \theta = \alpha_0 - \alpha - \gamma, \phi)$, where α_0 is the angle of attack at straight and level (zero pitch) flight. We assume that the state is fully observed (i.e. $\mathbf{h}(\mathbf{x}_t) = \mathbf{x}_t$) up to the measurement noise $\mathbf{w}_t$, and in our experiments we consider Gaussian noise distributions $\mathbf{V}_t^u$, $\mathbf{V}_t^x$, and $\mathbf{W}_t$. We note that the explicit consideration of control noise $\mathbf{v}_t^u$ ensures non-Gaussian uncertainty distributions at every time step, in addition to those arising from non-linear propagation.

B. Performance of Adaptive Mixture Importance Sampling

We implemented Algorithms 1 and 2 in Julia [21] using the Bullet physics engine [22] for continuous (swept) collision checking, and ran experiments on a Linux system equipped with a 3.0GHz 8-core Intel i7-5960X processor (although we note that the implementation presented in this work is only single-threaded). Figure 3 depicts estimation results for applying adaptive mixture IS (Alg. 1, $k = 20$, $\ell = 100$), non-adaptive IS (Alg. 1, $k = 2000$, $\ell = 1$), and naive MC ($Q = P$ in Equations (5) and (6)) to the nominal trajectory depicted in Figure 1. For discrete LQG control, $T = 100$ and the time discretization is $\Delta t = 0.129$ s. Both importance sampling methods use $D = 10$ mixture components, initialized with uniform weight aside from $q_{10} = P$ initialized with $\alpha_{10}^1 = 0.5$. Figure 3 is indicative of a problem that may arise from poorly chosen mixture weights. Non-adaptive IS (in blue) twice encounters positive collision samples with a very high likelihood ratio, indicating poor proportional representation by the IS distribution Q. The table in Figure 3 indicates that this type of event is relatively rare, as on average $\hat{V}_{\text{IS}}$ is much lower than $\hat{V}_{\text{NMV}}$ (owing to the construction of a good η) and near adaptive mixture IS in terms of error certificate.



Estimator means computed over 30 trials, $m = 1000$ samples					
$\hat{p}_{\text{AIS}}$	$\hat{p}_{\text{IS}}$	$\hat{p}_{\text{NMC}}$	$\hat{V}_{\text{AIS}}$	$\hat{V}_{\text{IS}}$	$\hat{V}_{\text{NMV}}$
0.436%	0.416%	0.471%	0.043%	0.048%	0.209%

Fig. 3. Example runs of Algorithm 1 with $k = 20, \ell = 100$ (adaptive mixture IS), with $k = 2000, \ell = 1$ (non-adaptive IS), and naive Monte Carlo for the nominal trajectory depicted in Figure 1. The dark lines represent the evolution of the MC estimators $\hat{p}$; the shadows around each line represent a confidence interval of ± 1 standard error, estimated as $\sqrt{\hat{V}}$. Both IS methods are shifted by 0.28 s in the lower plot to reflect the time required to derive the noise mean shifts η. Adaptive IS converges to an estimate with a usable level of certification within 500 samples (2 s).

On average both importance sampling methods process 1000 samples in ~ 4.5 s (specifically, AIS: 4.46 ± 0.13 s, IS: 4.51 ± 0.65 s). We can see from Figure 3, however, that only a few hundred samples are required to get a confident handle on trajectory CP, which we note may be processed in far less time than the airplane takes to fly its 13 s trajectory. The equivalent timings for both versions of IS are not surprising for the single-threaded implementation featured in this work, as the computational effort required by the stochastic mirror descent update is negligible compared to integrating dynamics, collision checking, and computing likelihood ratios. A parallel implementation might require larger batch sizes k in order to overcome the communication overhead inherent in coordinating adaptive IS and achieve the expected parallel MC speedup (see, e.g., [15] which implements Monte Carlo certification of point-quadrotor trajectory plans on a GPU with computation times on the order of 10 ms). Naive MC processes 1000 samples in $2.34 \pm .04$ s; the additional importance sampling time is spent entirely in

TABLE I

MIXTURE WEIGHTS α DERIVED THROUGH ALGORITHM 1 COMPARED TO HALF-SPACE VIOLATION PROBABILITY. ADAPTIVE IS BATCH SIZE $k = 20$, DEFENSIVE IMPORTANCE SAMPLING WEIGHT α_{10} LOWER-BOUNDED AT 10%.

Distribution component	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10} $(= P)$
Time step t	82	83	83	2	13	14	14	68	82	N/A
Plane component	Wing	Stabilizer	Wing	Tail	Wing	Wing	Stabilizer	Wing	Body	N/A
Half-space violation probability	0.177%	0.120%	0.111%	0.065%	0.041%	0.029%	0.026%	0.016%	0.012%	N/A
Example α^{100} $(m = 2000)$	0.165	0.097	0.091	0.101	0.038	0.079	0.062	0.047	0.069	0.247
Example α^{1000} $(m = 20000)$	0.208	0.157	0.191	0.115	0.043	0.057	0.032	0.037	0.058	0.1

evaluating probability density functions.

Table I gives a description of the mixture components used in importance sampling, as well as the final adaptive IS mixture weights from Figure 3. We can see from the last two rows that a few thousand samples is generally insufficient to converge to the optimal weights, but already at $m = 2000$ samples the relationship between good weights and half-space violation probability for the approximate marginal distributions defies heuristic approximation.

VI. CONCLUSIONS

We have presented an adaptive mixture importance sampling algorithm inspired by the statistics literature [17] and demonstrated its success in quantifying CP (with tight estimated error) for an LQG with EKF control policy applied to a non-linear system with a full rigid-body collision model. In particular, we note that this procedure succeeds in achieving a level of certifiable accuracy for which there are no comparable existing methods other than naive, non-variance-reduced, Monte Carlo. The adaptive nature of the procedure has been demonstrated as essential for selecting proper component weights for use in mixture IS, at negligible additional computational cost compared to non-adaptive mixture IS.

While this work may in its current form see direct application within a non-linear LQG control planner (for which we stress that a parallel implementation would be a key technology for enabling truly real-time use), we mention here a number of other future research avenues. First, we note that in this work a few thousand samples are sufficient to learn improved mixture weights. With a budget of tens of thousands of MC samples (possibly enabled by GPU), we might attempt to adaptively improve the high-dimensional distribution parameters η as well as the weights α. Second, estimating CP for control policies departing from the LQG approach of tracking a nominal trajectory (e.g., stochastic extended LQR [23]) may be considered using the same mixture IS techniques. Finally, we note that although the "rare event" considered throughout this paper has been obstacle collision, adaptive importance sampling as a Monte Carlo variance reduction technique may be applied to estimate a variety of other performance or safety requirements.

REFERENCES

[1] W. J. A. Dahm, "Technology Horizons: a Vision for Air Force Science & Technology During 2010-2030," *USAF HQ*, 2010.

[2] "Science and Technology Strategic Plan, C4ISR," Office of Naval Research, Tech. Rep., 2012, available at http://www.onr.navy.mil/Science-Technology/Departments/Code-31.aspx.

[3] J. V. D. Berg, P. Abbeel, and K. Goldberg, "LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information," *International Journal of Robotics Research*, 2011.

[4] J. A. Starek, B. Açıkmeşe, I. A. D. Nesnas, and M. Pavone, "Spacecraft Autonomy Challenges for Next Generation Space Missions," in *Advances in Control System Technology for Aerospace Applications*, ser. Lecture Notes in Control and Information Sciences, E. Feron, Ed., ch. 1.

[5] S. Lavalle, *Planning Algorithms*. Cambridge University Press, 2006.

[6] L. Janson, E. Schmerling, and M. Pavone, "Monte Carlo Motion Planning for Robot Trajectory Optimization Under Uncertainty," in *International Symposium on Robotics Research*, 2015.

[7] S. Patil, J. van den Berg, and R. Alterovitz, "Estimating probability of collision for safe motion planning under Gaussian motion and sensing uncertainty," in *Proc. IEEE Conf. on Robotics and Automation*, 2012.

[8] W. Sun, L. G. Torres, J. V. D. Berg, and R. Alterovitz, "Safe Motion Planning for Imprecise Robotic Manipulators by Minimizing Probability of Collision," in *International Symposium on Robotics Research*, 2013.

[9] F. Oldewurtel, C. 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.

[10] B. Luders, M. Kothari, and J. P. How, "Chance constrained RRT for probabilistic robustness to environmental uncertainty," in *AIAA Conf. on Guidance, Navigation and Control*, 2010.

[11] G. S. Aoude, B. D. Luders, J. M. Joseph, N. Roy, and J. P. How, "Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns," *Autonomous Robots*, 2013.

[12] M. Kothari and I. Postlethwaite, "A probabilistically robust path planning algorithm for UAVs using rapidly-exploring random trees," *Journal of Intelligent & Robotic Systems*, 2013.

[13] B. D. Luders, S. Karaman, and J. P. How, "Robust sampling-based motion planning with asymptotic optimality guarantees," in *AIAA Conf. on Guidance, Navigation and Control*, 2013.

[14] W. Liu and M. H. Ang, "Incremental Sampling-Based Algorithm for Risk-Aware Planning Under Motion Uncertainty," in *Proc. IEEE Conf. on Robotics and Automation*, 2014.

[15] B. Ichter, E. Schmerling, A. Agha-mohammadi, and M. Pavone, "Real-Time Stochastic Kinodynamic Motion Planning via Multiobjective Search on GPUs," *submitted to IEEE Conf. on Robotics and Automation*, 2017, available at http://arxiv.org/abs/1607.06886.

[16] J. Pan, S. Chitta, and D. Manocha, "Probabilistic collision detection between noisy point clouds using robust classification," 2011.

[17] E. Ryu, "Convex optimization for Monte Carlo: Stochastic optimization for importance sampling," Ph.D. dissertation, Stanford University, Institute for Computational & Mathematical Engineering, 2016. [Online]. Available: https://purl.stanford.edu/kb323fs8835

[18] A. B. Owen, *Monte Carlo theory, methods and examples*, 2013, available at http://statweb.stanford.edu/~owen/mc/.

[19] S. Haykin, *Kalman filtering and neural networks*. John Wiley & Sons, 2004.

[20] R. W. Beard and T. W. McLain, *Small Unmanned Aircraft: Theory and Practice*. Princeton University Press, 2012.

[21] J. Bezanson, S. Karpinski, V. B. Shah, and A. Edelman, "Julia: A Fast Dynamic Language for Technical Computing," 2012, available at http://arxiv.org/abs/1209.5145.

[22] E. Coumans. Bullet physics. [Online]. Available: http://bulletphysics.org

[23] W. Sun, J. van den Berg, and R. Alterovitz, "Stochastic extended lqr for optimization-based motion planning under uncertainty," *IEEE Transactions on Automation Sciences and Engineering*, 2016.