A Martingale Approach and Time-Consistent Sampling-based Algorithms for Risk Management in Stochastic Optimal Control

Vu Anh Huynh   Leonid Kogan   Emilio Frazzoli

Abstract—In this paper, we consider a class of stochastic optimal control problems with risk constraints that are expressed as bounded probabilities of failure for particular initial states. We present here a martingale approach that diffuses a risk constraint into a martingale to construct time-consistent control policies. The martingale stands for the level of risk tolerance that is contingent on available information over time. By augmenting the system dynamics with the controlled martingale, the original risk-constrained problem is transformed into a stochastic target problem. We extend the incremental Markov Decision Process (iMDP) algorithm to approximate arbitrarily well an optimal feedback policy of the original problem by sampling in the augmented state space and computing proper boundary conditions for the reformulated problem. We show that the algorithm is both probabilistically sound and asymptotically optimal. The performance of the proposed algorithm is demonstrated on motion planning and control problems subject to bounded probability of collision in uncertain cluttered environments.

I. INTRODUCTION

Controlling dynamical systems in uncertain environments is a fundamental and essential problem in several fields, ranging from robotics [1] to management [2]. Given a system with dynamics described by a controlled diffusion process, a stochastic optimal control problem is to find an optimal feedback policy to optimize an objective function. Risk management has always been an important part of stochastic optimal control problems to guarantee safety during the execution of control policies. For instance, in critical applications such as self-driving cars and robotic surgery, regulatory authorities can impose a threshold of failure probability during operation of these systems. Thus, finding control policies that fully respect this type of constraint is important in practice.

There has been intensive literature on stochastic optimal control without risk constraints. Even in this setting, it is well-known that closed-form or exact algorithmic solutions for general continuous-time, continuous-space stochastic optimal control problems are computationally challenging [3]. Traditional approaches such as such as discrete Markov Decision Process approximation [4], [5] and solving associated Hamilton-Jacobi-Bellman (HJB) PDEs [6], [7] scale poorly with the dimension of the state space. Recently, in [8], [9], a new computationally-efficient sampling-based algorithm called the incremental Markov Decision Process (iMDP) algorithm has been proposed to provide asymptotically-optimal solutions.

Risk management in stochastic optimal control have also been received extensive attention by researchers in several fields. In robotics, a common risk management problem is chance-constrained optimization [10], [11]. Chance constraints specify that starting from a given initial state, the time-0 probability of success must be above a given threshold where success means reaching goal areas safely. Alternatively, we call these constraints risk constraints if we concern more about failure probabilities. The Lagrangian approach [12], [13] is a possible method for solving the mentioned constrained optimization. However, this approach requires numerical procedures to compute Lagrange multipliers before obtaining a policy, which is computationally demanding for high dimensional systems and unsuitable for online robotics applications. Other previous works in robotics [14]–[16] do not solve the continuous-time problems directly and often modify the problem formulation. As a result, available methods are either computationally intractable or only able to provide approximate but time-inconsistent solutions. Time-inconsistent policies lead to inconsistent behaviors in which risk preferences change in an irrational manner between periods. Recognizing this issue, in [17], the authors used Markov decision time-consistent risk measures [18] to assess the risk of future cost stream in a consistent manner and established a dynamic programming equation for this modified formulation. Solving the resulting dynamic programming equation is, however, computationally difficult as it involves functionals as control variables.

In mathematical finance, closely-related problems have been studied in the context of hedging with portfolio constraints where constraints on terminal states are enforced almost surely (a.s.), yielding so-called stochastic target problems [19], [20]. Research in this field focuses on deriving HJB equations for this class of problems.

In this paper, we investigate stochastic optimal control problems with risk constraints that are expressed in terms of time-0 bounded probabilities of failure for particular initial states. We present here a martingale approach to solve these problems such that obtained control policies are time-consistent with the initial threshold of failure probability. The martingale represents the level of risk tolerance that is contingent on available information over time. Thus, the martingale approach transforms a risk-constrained problem into a stochastic target problem. By sampling in the augmented state space and computing proper boundary conditions of the reformulated problem, we extend the iMDP algorithm to compute anytime solutions after a small number of iterations. When more computing time is allowed, the proposed algorithm refines the solution quality in an efficient manner.

The main contribution of this paper is a martingale approach that fully respects the considered risk constraints for systems with continuous-time dynamics in a time-consistent manner. The approach is suitable to manage risk in practical robotics applications without directly deriving HJB equations and guarantees probabilistically-sound and asymptotically-optimal solutions in an incremental procedure.

This paper is organized as follows. A formal problem definition is given in Section II. In Section III, we discuss the martingale approach and the key transformation. The extended iMDP algorithm and analysis are described in Sections IV-V. We present experimental results in Section VI and conclude the paper in Section VII.

II. PROBLEM DEFINITION

We first present a generic stochastic optimal control formulation with definitions and technical assumptions as shown in [8], [9], [21]. We then formulate risk constraints.
Stochastic Dynamics: Let $d_x$, $d_u$, and $d_w$ be positive integers. Let $S$ be a compact subset of $\mathbb{R}^{d_x}$, which is the closure of its interior $S^o$ and has a smooth boundary $\partial S$. Let a compact subset $U$ of $\mathbb{R}^{d_u}$ be a control set. The state is $x(t) \in S$, which is fully observable at all times.

Suppose that a stochastic process $\{w(t); t \geq 0\}$ is a $d_w$-dimensional Brownian motion on some probability space. We define $\{F_t; t \geq 0\}$ as the augmented filtration generated by $w(t)$. Let a control process $\{u(t); t \geq 0\}$ be a $U$-valued, measurable random process defined on the same probability space such that the pair $(u(\cdot), w(\cdot))$ is admissible [8]. Let the set of all such control processes be $\mathcal{U}$. Let $\mathbb{R}^{d_x \times d_u}$ be the set of all $d_x$ by $d_u$ real matrices. We consider system dynamics described by a controlled diffusion process:

$$
dx(t) = f(x(t), u(t)) \, dt + F(x(t), u(t)) \, dw(t), \quad \forall t \geq 0 \tag{1}
$$

where $f : S \times U \to \mathbb{R}^{d_x}$ and $F : S \times U \to \mathbb{R}^{d_x \times d_w}$ are bounded measurable and continuous functions as long as $x(t) \in S^o$. The initial state $x(0)$ is a random vector in $S$. We assume that the matrix $F(\cdot, \cdot)$ has full row rank. The continuity requirement of $f$ and $F$ can be relaxed with mild assumptions [8], [22] such that we still have a weak solution to Eq. (1) that is unique in the weak sense [22].

Cost-to-go Function and Risk Constraints: We define the first exit time $T^u_x : \mathcal{U} \times S \to [0, +\infty]$ under a control process $u(\cdot) \in \mathcal{U}$ starting from $x(0) = z \in S$ as $T^u_x = \inf \{ t : x(t) = z, \ x(t) \notin S^o, \text{ and Eq.(1)} \}$. In other words, $T^u_x$ is the first time that the trajectory of the dynamical system given by Eq. (1) starting from $x(0) = z$ hits the boundary $\partial S$ of $S$. The expected cost-to-go function under a control process $u(\cdot)$ is a mapping from $S$ to $\mathbb{R}$ defined as

$$
J_u(z) = E^z_0 \left[ \int_0^{T^u_x} \alpha^x g(x(t), u(t)) \, dt + \alpha^{T^u_x} h(x(T^u_x)) \right]. \tag{2}
$$

where $\mathbb{E}^z_0$ denotes the conditional expectation given $x(0) = z$, and $g : S \times U \to \mathbb{R}^n$, $h : S \to \mathbb{R}$ are bounded measurable and continuous functions, called the cost rate function and the terminal cost function, respectively, and $\alpha \in [0, 1]$ is the discount rate. We further assume that $g(x, u)$ is uniformly H"older continuous in $x$ with exponent $2p \in (0, 1]$ for all $u \in \mathcal{U}$. The discontinuity of $g, h$ is treated as in [8], [22].

Let $\Gamma \subset \partial S$ be a failure set, and $\eta \in [0, 1]$ be a risk threshold given as a parameter. A risk constraint for an initial state $x(0) = z$ under a control process $u(\cdot)$ is defined as $P^u_0(x(T^u_x) \in \Gamma) \leq \eta$ where $P^u_t$ denotes the conditional probability at time $t$ given $x(0) = z$. That is, controls that drive the system from time 0 until the first exit time must be consistent with the choice of $\eta$ and initial state $z$ at time 0.

Intuitively, the constraint enforces that starting from a given state $z$ at time $t = 0$, if we execute a control process $u(\cdot)$ for $N$ times, when $N$ is very large, there are at most $N\eta$ executions resulting in failure. Control processes $u(\cdot)$ that satisfy this constraint are called time-consistent.

Let $\mathbb{R}$ be the extended real number set. The optimal cost-to-go function $J^* : S \to \mathbb{R}$ is defined in $\mathcal{O}PT^{1,2}$:

$$
J^*(z; \eta) = \inf_{u(\cdot) \in \mathcal{U}} J_u(z) \ \text{s.t.} \ P^u_0(x(T^u_x) \in \Gamma) \leq \eta. \tag{3}
$$

A control process $u(\cdot)$ is called optimal if $J_u(z) = J^*(z; \eta)$. For any $\epsilon > 0$, a control process $u(\cdot)$ is called an $\epsilon$-optimal policy if $|J_u(z) - J^*(z; \eta)| \leq \epsilon$.

We call a sampling-based algorithm probabilistically sound if the probability that a solution returned by the algorithm is feasible approaches one as the number of samples increases. We also call a sampling-based algorithm asymptotically-optimal if the sequence of solutions returned from the algorithm converges to an optimal solution in probability as the number of samples approaches infinity. Solutions returned from algorithms with such properties are called probabilistically-sound and asymptotically-optimal.

In this paper, we consider the problem of computing the optimal cost-to-go function $J^*$ and an optimal control process $u^*$ if obtainable. Our approach, outlined in Section IV, approximates $J^*$ and $u^*$ in any time horizon using an incremental sampling-based algorithm that is both probabilistically-sound and asymptotically-optimal.

III. Martingale Approach

The following lemma diffuses risk constraints to transform the considered problem into a stochastic target problem.

Lemma 1 (see [24], [25]) From $x(0) = z$, a control process $u(\cdot)$ is feasible for $\mathcal{O}PT^{1,2}$ if there exists a square-integrable (but possibly unbounded) stochastic process $c(\cdot)$ in $\mathbb{R}^{d_w}$ and a martingale $q(\cdot)$ in $[0, 1]$ a.s. satisfying:

1. $q(0) = \eta$, and $dq(t) = c^T(t)dw(t)$,
2. $1_{1}(x(T^u_x)) \leq q(T^u_x) \ a.s.$

where $1_{1}(x) = 1$ for $x \in \Gamma$ and 0 otherwise. The martingale $q(t)$ stands for the level of risk tolerance at time $t$. We call $c(\cdot)$ a martingale control process.

The proof based on the martingale representation theorem can be found in the full version of this paper [25].

A. Stochastic Target Problem

Using the above lemma, we augment the original system dynamics with the martingale $q(t)$ into the following form:

$$
d\left[ \begin{array}{c} x(t) \\ q(t) \end{array} \right] = \left[ \begin{array}{c} f(x(t), u(t)) \\ 0 \end{array} \right] dt + \left[ \begin{array}{c} F(x(t), u(t)) \\ c^T(t) \end{array} \right] dw(t), \tag{4}
$$

where $(u(\cdot), c(\cdot))$ is the control process of the above dynamics. The initial value of the new state is $(x(0), q(0)) = (z, \eta)$. We will refer to the augmented state space $S \times [0, 1]$ as $\tilde{S}$ and the augmented control space $U \times \mathbb{R}^{d_w}$ as $\tilde{U}$. We also refer to the nominal dynamics and diffusion matrix of Eq. (4) as $\tilde{f}(x, q, u, c)$ and $\tilde{F}(x, q, u, c)$ respectively.

It is well known that in the following reformulated problem, optimal control processes are Markov control policies [24]. Thus, let us now focus on the set of Markov controls that depend only on the current state, i.e., $(u(t), c(t))$ is a function only of $(x(t), q(t))$, for all $t \geq 0$. A function $\varphi : \tilde{S} \to \tilde{U}$ represents a Markov or feedback control policy, which is known to be admissible with respect to the process noise $w(t)$. Let $\Psi$ be the set of all such policies $\varphi$. Let $\mu : \tilde{S} \to U$ and $\kappa : \tilde{S} \to \mathbb{R}^{d_u}$ so that $\varphi = (\mu, \kappa)$. We rename $T^u_x$ to $T^\varphi_x$ for the sake of notation clarity. Using these notations, $\mu(\cdot, 1)$ is thus a Markov control policy that maps from $S$ to $U$ for the problem without risk constraints. Henceforth, we will use $\mu(\cdot)$ to refer to $\mu(\cdot, 1)$. Let $\Pi$ be the set of all such Markov control policies $\mu(\cdot)$ on $S$.

Now, let us rewrite cost-to-go function $J_u(z)$ in Eq. (2):

$$
J_\mu(z; \eta) = E^z_0 \eta \left[ \int_0^{T^\varphi_x} \alpha^x g(x(t), \mu(x(t), q(t))) \, dt + \alpha^{T^\varphi_x} h(x(T^\varphi_x)) \right].
$$

3Compared to other sampling-based algorithms, e.g. [23], we have relaxed this concept from a.s. convergence to convergence in probability.
We therefore transform the risk-constrained problem in Eq. (3) into a stochastic target problem OPT2 as follows:\(^4\):
\[
J^*(z, \eta) = \inf_{\varphi \in \Psi, Q \in Q_0} J_\varphi(z, \eta) \text{ s.t. } 1_{\Gamma}(x(T_n^\varphi)) \leq q(T_n^\varphi) \text{ a.s.}
\] (5)

The constraint in the above formulation specifies the relationship of random variables at the terminal time as target, and hence the name of this formulation [24]. Here, we solve for feedback control policies $\varphi$ for all $(z, \eta) \in S$, and the boundary conditions are not fully specified a priori. We subsequently discuss how to construct its boundary and compute the boundary conditions to remove the constraint.

B. Characterization and Boundary Conditions

The domain of the stochastic target problem is:
\[
D = \{(z, \eta) \in S \mid \exists \varphi \in \Psi \text{ s.t. } 1_{\Gamma}(x(T_n^\varphi)) \leq q(T_n^\varphi) \text{ a.s.}\}.
\]

By the definition of the risk-constrained problem, we can see that if $(z, \eta) \in D$ then $(z', \eta') \in D$ for any $\eta < \eta' \leq 1$. Thus, for each $z \in S$, we define $\gamma(z) = \inf \{ \eta \in [0, 1] \mid (z, \eta) \in D \}$ as the infimum of risk tolerance at $z$. Therefore, we have:
\[
\gamma(z) = \inf_{u \in U(z)} P^0_n(\{x(T_n^u) \in \Gamma\}) = \inf_{u \in U(z)} E^0_n \left[ 1_{\Gamma}(x(T_n^u)) \right].
\] (6)

Thus, the boundary of $D$ is
\[
\partial D = S \times \{1\} \cup \{(z, \gamma(z)) \mid z \in S\} \cup \{(z, \eta) \mid z \in \partial S, \eta \in [\gamma(z), 1]\}.
\]

For states in $\{(z, \eta) \mid z \in \partial S, \eta \in [\gamma(z), 1]\}$, the system stops on $\partial S$ and takes terminal values according to $h(\cdot)$.

Now, let $\eta = 1$, we notice that $J^*(z, 1)$ is the optimal cost-to-go from $z$ for the stochastic optimal problem without the risk constraint. An optimal control process that solves the unconstrained problem is given by a Markov policy $\mu^*(\cdot, 1) \in \Pi$. This policy leads to the failure probability function $\Upsilon: S \rightarrow [0, 1]$ defined as $\Upsilon(z) = E[1_{\Gamma}(x(T_n^\mu^\star)) \mid \forall z \in S$. By the definitions of $\gamma$ and $\Upsilon$, we can recognize that $\Upsilon(z) \geq \gamma(z)$ for all $z \in S$.

Since following the policy $\mu^*(\cdot, 1)$ from an initial state $z$ yields a failure probability $\Upsilon(z)$, we infer that $J^*(z, 1) = J^*(z, \Upsilon(z))$. We can show that $J^*(z, \eta)$ is non-increasing, and thus $\forall \eta \in [\Upsilon(z), 1] \Rightarrow J^*(z, \eta) = J^*(z, 1)$. As a consequence, from an initial state $z$, if $\eta \geq \Upsilon(z)$, it is optimal to execute an optimal control policy of the corresponding unconstrained problem from the initial state $z$. In addition, we can also infer that for augmented states $(x(t), q(t))$ where $q(t) = 1.0$, the optimal martingale control $c^*(t)$ is 0.

It is clear that $J^*(z, \eta) = +\infty$ for all $0 \leq \eta < \gamma(z)$.

The following lemma characterizes the optimal martingale control $c^*(t)$ for augmented states $(x(t), q(t) = \gamma(x(t)))$.

The proof is presented in [25].

**Lemma 2** To have feasible solutions to the problem in Eq. (5), when $q(t) = \gamma(x(t))$ and $u(t)$ are chosen, we must have:
\[
c(t)^T = \frac{\partial S}{\partial x(t)}(T) F(x(t), u(t)).
\]

In addition, if a control process that solves Eq. (6) is obtainable, say $u_\star$, the cost-to-go due to that control process is $J_{u_\star}(z)$. We will conveniently refer to $J_{u_\star}(z)$ as $J^*(z)$.

Under the mild assumption that $u_\star$ is unique, it follows that $J^*(z) = J^*(z, \gamma(z))$. We also emphasize that $(x(t), q(t))$ is inside the interior $D^\star$ of $D$; the usual dynamic programming principle holds.

**IV. ALGORITHM**

Now, we briefly overview the Markov chain approximation technique. We then present the extended IMDP algorithm that incrementally constructs the boundary conditions and computes solutions.

A. Markov Chain Approximation

A discrete-state Markov decision process (MDP) is a tuple $M = (X, A, P, G, H)$ where $X$ is a finite set of states, $A$ is a set of actions that is possibly a continuous space, $P(\cdot|\cdot, \cdot) : X \times X \times A \rightarrow \mathbb{R}_{\geq 0}$ is the transition probability function, $G(\cdot|\cdot) : X \times A \rightarrow \mathbb{R}$ is an immediate cost function, and $H : X \rightarrow \mathbb{R}$ is a terminal cost function. From an initial state $\xi_0$, under a sequence of controls $\{\xi_i; i \in \mathbb{N}\}$, the induced trajectory $\{\xi_i; i \in \mathbb{N}\}$ is generated by following the transition probability function $P$.

On the state space $S$, we want to approximate $J^*(z, 1)$, $\gamma(z)$ and $J^*(z)$, and it is sufficient to consider optimal Markov controls as shown in [8], [9]. The Markov chain approximation method approximates the continuous dynamics in Eq. (1) using a sequence of MDPs $\{\bar{M}_n = (S_n, U_n, P_n, G_n, H_n)\}_{n=0}^\infty$ and a sequence of holding times $\{\Delta n_{n=0}^\infty\}$ that are locally consistent. In particular, we construct $G_n(z, v) = g(z, v)\Delta n_{n=0}^\infty$, $H_n(z) = h(z)$ for each $z \in S_n$ and $v \in U_n$. We also require that $\lim_{n, m \rightarrow \infty} \sum_{n \in \mathbb{N}, \in \mathbb{N}} \|\Delta n_{n=0}^\infty || = 0$ where $\Omega_n$ is the sample space of $M_n$, $\Delta n_{n=0}^\infty = \xi_{i+1} - \xi_i$ for all $z \in S_n$ and $v \in U_n$.

The main idea of the Markov chain approximation approach for solving the original continuous problem is to solve a sequence of control problems defined on $\{M_n\}_{n=0}^\infty$ as follows. A Markov or feedback policy $\mu_n$ is a function that maps each state $z \in S_n$ to a control $u(z, \eta) \in U_n$. The set of all such policies is $\Pi_n$. We define $t_i = \sum_{o=1}^{i-1} \Delta n_{o=0}^\infty$ for $i \geq 1$ and $t_0 = 0$. Given a policy $\mu_n$ that approximates a Markov control process $u(\cdot)$ in Eq. (2), the corresponding cost-to-go due to $\mu_n$ on $M_n$ is:
\[
J_{n, \mu_n}(z) = E_{\mu_n} P_n \left[ \sum_{i=0}^{t_i-1} \alpha^i_n G_n(\xi_i, \mu_n(\xi_i)) + \alpha^i_n H_n(\xi_i) \right],
\]
where $E_{\mu_n}$ denotes the conditional expectation given $\xi_0 = z$ under $P_0$, and $\{\xi_i; i \in \mathbb{N}\}$ is the sequence of states of the controlled Markov chain under the policy $\mu_n$ and $t_i$ is termination time defined as $t_i = \min\{i : \xi_i \in S_n\}$ where $\partial S_{\mu_n} = \partial S \cap S_n$.

The optimal cost-to-go function $J^*_n : S \rightarrow \mathbb{R}$ that approximates $J^*(z, 1)$ is defined as
\[
J^*_n(z, 1) = \inf_{\mu_n \in \Pi_n} J_{n, \mu_n}(z) \forall z \in S_n.
\] (7)

An optimal policy, denoted by $\mu^*_n$, satisfies $J_{n, \mu^*_n}(z) = J^*_n(z)$ for all $z \in S_n$. For any $\epsilon > 0$, $\mu_n$ is an $\epsilon$-optimal policy if $\|J_{n, \mu_n} - J^*_n\|_{\infty} \leq \epsilon$. We also define the failure probability function $\Upsilon_n : S_n \rightarrow [0, 1]$ due to an optimal policy $\mu^*_n$, as follows:
\[
\Upsilon_n(z) = E_{\mu^*_n} \left[ 1_{\Gamma(\xi_{t_i}^\mu_n)} \right] x(0) = z ; \mu^*_n \right] \forall z \in S_n.
\] (8)

In addition, the min-failure probability $\gamma_n$ on $M_n$ that approximates $\gamma(z)$ is defined as:
\[
\gamma_n(z) = \inf_{\mu_n \in \Pi_n} E_{\mu_n} \left[ 1_{\Gamma(\xi_{t_i}^\mu_n)} \right] \forall z \in S_n.
\] (9)
The optimization programs in Eqs. (7)-(9) usually have two different optimal feedback control policies. Let \( \nu_n \in \Pi_n \), be a control policy on \( \mathcal{M}_n \), that achieves \( \gamma_n \), then we define the cost-to-go due to \( \nu_n \) as \( J_n^* \), which approximates \( J^* \).

Similarly, in the augmented state space \( \overline{S}_n \), we use a sequence of MDPs \( \{ \overline{M}_n \} \), \( \{ \overline{P}_n \} \), \( \{ \overline{G}_n \} \), \( \{ \overline{H}_n \} \) and a sequence of holding times \( \{ \Delta t_n \} \) that are locally consistent with the augmented dynamics in Eq. (4). In particular, \( \overline{S}_n \) is a random subset of \( S \subseteq \overline{S}_n \), \( \overline{G}_n \), and \( \overline{H}_n \) are equal to \( G_n \) if \( \eta \in \{ \gamma_n\{1,\} \) and \(+\infty \) otherwise. We also construct the transition probabilities \( \overline{P}_n \) on \( \overline{M}_n \) and holding time \( \Delta \) that satisfy the local consistency conditions for nominal dynamics \( \overline{J}(x, q, u, c) \) and diffusion matrix \( \overline{F}(x, q, u, c) \).

A trajectory on \( \overline{M}_n \) is denoted as \( \{ \overline{\xi}_n^i \} \) where \( \overline{\xi}_n^i \) is a Markov policy \( \overline{\nu}_n(\cdot, \cdot) \) that maps each value pair \( (x, q) \in \overline{S}_n \) to a control \( (\mu_n(\cdot, \cdot), \kappa_n(\cdot, \cdot)) \). Admissible \( \kappa_n \) at \( (z, 1) \in \overline{S}_n \) is 0 and at \( (z, \gamma_n(z)) \in \overline{S}_n \) is a function of \( \mu_n(z, \gamma_n(z)) \) as shown in Lemma 2. Admissible \( \kappa_n \) for other states in \( \overline{S}_n \) is such that the martingale-component process \( \{ \overline{\xi}_n^i \} \) belongs to \([0,1]\) almost surely. We can show that equivalently, each control component of \( \kappa_n(z, \eta) \) belongs to \([0,1]\) almost surely. The set of all such policies \( \overline{\nu}_n \), is \( \mathfrak{P}_n \).

Under the cost-to-go \( J_n^* \), \( \overline{M}_n \) that approximates \( J^* \) is defined as:

\[
J_n^*(\cdot, \cdot) = \mathbb{E}_{\cdot, \cdot}^{\overline{P}_n} \left[ \sum_{t=0}^{n-1} \alpha_t \overline{G}_n(\overline{\xi}_t, \mu_n(\overline{\xi}_t)) + \alpha_n \overline{H}_n(\overline{\xi}_n) \right],
\]

where \( \overline{\xi}_t = \sum_{i=0}^{t-1} \Delta t_i \overline{\xi}_i \) for \( i \geq 1 \) with \( \overline{\xi}_0 = 0 \), and \( \Delta t_i \) is index when the x-component of \( \overline{\xi}_i \) first arrives at \( \partial S \). The approximating optimal cost \( J_n^* \) is \( \mathbb{E}_n \) for \( J_i^* \) is:

\[
J_n^*(\cdot, \cdot) = \inf_{\nu_n \in \mathfrak{P}_n} J_n(\cdot, \cdot) \forall (\cdot, \cdot) \in \overline{S}_n.
\]

To solve this optimization, we compute approximate boundary values for states on the boundary of \( D \) using the sequence of MDP \( \{ \overline{M}_n \} \) on \( S \), the normal dynamic programming principle holds.

The extension of iMDP outlined below is designed to compute the sequence of optimal cost-to-go \( \{ J_n^* \} \) associated failure probability function \( \{ \gamma_n \} \), mini-max failure probabilities \( \{ \gamma_n \} \), mini-max failure cost value \( \{ \gamma_n \} \), the sequence of anytime control policies \( \{ \mu_n \} \) and \( \{ \kappa_n \} \) in an efficient iterative procedure.

### B. Extension of iMDP

Before presenting the details of the algorithm, we discuss a number of primitive procedures. More details about these procedures can be found in [8, 9, 25]. Briefly, the procedure \texttt{Sample}(X) samples states independently and uniformly in \( X \). \texttt{Nearest}(x, y, k) returns the k nearest states \( \hat{y} \) that are closest to \( x \) in terms of the \( d_Y \)-dimensional Euclidean norm. \texttt{ComputeHoldingTime}(c, k, d) returns a holding time \( \Theta((\log k)/k) \) for \( c \in (0, 1) \) and \( \theta = (0, 1) \). The procedure \texttt{ComputeTranProb}(z, v, \tau, K) returns locally consistent transition probabilities for nominal dynamics \( K \) and dispersion matrix \( K \) from a state \( z \) under a control \( v \) within a holding time \( \tau \). Below, we discuss in more detail backward extension and control construction.

**Backward Extension:** Given \( T > 0 \) and two states \( z, z' \in S \), the procedure \texttt{ExtBackwards}(z, z', T) returns a triple \((x, v, \tau)\) such that \( (i)\ x(t) = f(x(t), u(t))dt \) and \( u(t) = v \in U \) for all \( t \in [0, \tau] \), \( (ii)\ \tau = T \), \( (iii)\ x(t) \in S \) for all \( t \in [0, \tau] \), \( (iv)\ x(\tau) = z \), and \( (v)\ x(0) \) is close to \( z' \). If no such trajectory exists, the procedure returns failure. We can solve for the triple \((x, v, \tau)\) by sampling several controls \( v \) and choosing the control resulting in \( x(0) \) that is closest to \( z' \).

When \((z, \eta, \eta') \) are in \( \{ S \} \), the procedure \texttt{ExtBackwards}(z, \eta, \eta', T) returns \((x, q, v, \tau)\) in which \((x, q, v, \tau)\) is output of \texttt{ExtBackwards}(z, \eta, \eta', T) and \( q \) is sampled according to a Gaussian distribution \( N(\eta' \cdot q, \sigma) \) where \( \sigma \) is a parameter.

**Sampling and Discovering Controls:** For \( z \in S \) and \( y \subseteq S \), the procedure \texttt{ConstructControls}(k, z, y, T) returns a set of \( k \) controls in \( U \). We can uniformly sample \( k \) controls in \( U \). Alternatively, for each state \( z' \) \texttt{Nearest}(z, y, k), we solve for a control \( v \in U \) such that \((i)\ x(t) = f(x(t), u(t))dt \) and \( u(t) = v \in U \) for all \( t \in [0, T] \). (ii) \( x(t) \in S \) for all \( t \in [0, T] \). (iii) \( x(0) = z \) and \( x(T) = z' \).

For \((z, \eta, \eta') \in \{ S \} \) and \( y \subseteq S \), the procedure \texttt{ConstructControls}(k, z, \eta, y, T) returns a set of \( k \) controls in \( U \) such that the \( U \)-component of these controls are computed as in \texttt{ConstructControls}, and the martingale-control-components are sampled in admissible sets.

The extended iMDP algorithm is presented in Algorithms 1-5. The algorithm incrementally refines two MDP sequences, namely \( \{ \mathcal{M}_n \} \) and \( \{ \mathcal{M}_n \} \), and two holding time sequences, namely \( \{ \Delta t_n \} \) and \( \{ \Delta t_n \} \), that consistently approximate the original system in Eq. (1) and the augmented system in Eq. (4) respectively. We associate with \( z \in S \) a cost value \( J_n(z, 1) \), a control \( \mu_n(z, 1) \), a failure probability \( \gamma_n(z) \) due to \( \mu_n(z, 1) \), a min-max failure probability \( \gamma_n(z) \), a cost-to-go value \( J_n(z, 1) \) induced by the obtained min-failure policy. Similarly, we associate with \( z \in S \), a cost value \( J_n(z, 1) \), a control \( \mu_n(z, \eta) \).

As shown in Algorithm 1, initially, empty MDP models \( \mathcal{M}_0 \) and \( \mathcal{M}_0 \) are created. The algorithm then executes a
number of iterations in which it samples states on the pre-specified part of the boundary $\partial D$, constructs the unspecified part of $\partial D$ using ConstructBoundary and processes the interior of $D$ using ProcessInterior.

In Algorithm 2, we show the implementation of the procedure ConstructBoundary. We construct a finer MDP model $M_\nu$ based on the previous model as follows. A state $z_{\nu}$ is sampled from the interior of the space $S$ (Line 1). The nearest state $z_{\text{near}}$ to $z_{\nu}$ (Line 2) in the previous model is used to construct an extended state $z_{\text{ext}}$ by using the procedure ExtendBackwardsS at Line 3. The extended states $z_{\text{ext}}$ and $(z_{\nu}, 1)$ are added into $S_{\text{t}}$ and $S_{\text{n}}$ respectively. The associated cost value $J_{\nu}(z_{\text{ext}})$, failure probability $\gamma_{\nu}(z_{\text{ext}})$, min-failure probability $\gamma_{\text{min}}(z_{\text{ext}})$, min-failure cost value $J_{\text{min}}(z_{\text{ext}})$ and control $\mu_{\text{opt}}(z_{\text{ext}})$ are initialized at Line 8.

We then perform $L_{\nu} \geq 1$ updating rounds in each iteration (Lines 9-12). In particular, we construct the update-set $Z_{\text{update}}$ consisting of $K_{\nu} = \Theta(|S_{\nu}|^\theta)$ states and $z_{\text{ext}}$ where $|K_{\nu}| < |S_{\nu}|$. For each state $z$ in $Z_{\text{update}}$, the procedure UpdateS as shown in Algorithm 4 computes:

$$J_{\nu}(z, 1) = \min_{v \in U_{\nu}(z)} \{G_{\nu}(z, v) + \alpha \Delta_{\text{sm}}(z) E_{P_{\nu}}[J_{\nu-1}(y) | z, v]\}.$$  

To implement this Bellman update, a set of $U_{\nu}$ controls is constructed using the procedure ConstructControlsS where $|U_{\nu}| = \Theta(|S_{\nu}|)$ at Line 2 as done in the original iMDP algorithm [8], [25]. Moreover, we also update $\gamma_{\nu}$ and improve the min-failure probability $\gamma_{\text{min}}$ and its induced min-failure cost value $J_{\text{min}}$ in Lines 9-12.

Similarly, in Algorithm 3, we carry out the sampling and extending process in the augmented state space $S$ to refine the MDP sequence $M_{\nu}$ (Lines 1-3). In this procedure, we update the cost-to-go $J_{\nu}$ for states in the interior $D^{\nu}$ of $D$ using the procedure UpdateSM as shown in Algorithm 5. When a state $z \in S_{\nu}$ is updated in UpdateSM, we perform the following Bellman update:

$$J_{\nu}(z) = \min_{(x, v) \in C_{\nu}(z)} \{G_{\nu}(z, v) + \alpha \Delta_{n}(z) E_{P_{\nu}}[J_{\nu-1}(y) | z, v, c]\},$$

where the control set $C_{\nu}$ is constructed by the procedure ConstructControlsSM, and the transition probability $P_{\nu}(z, v, c)$ consistently approximates the augmented dynamics in Eq. (4). Using the characteristics in Section III-B, we implement the Bellman update at Line 5 in Algorithm 5 where the notation $1_A$ is 1 if the event $A$ occurs and 0 otherwise. That is, when the martingale state $s$ of a state $\mathcal{G}(z, v)$ in the support $Z_{\text{near}}$ is at least $\gamma_{\nu}(y)$, we substitute $J_{\nu}(y)$ with $J_{\nu}(y, 1)$. Similarly, when the martingale state $s$ is equal to $\gamma_{\nu}(y)$, we substitute $J_{\nu}(y)$ with $J_{\nu}(y, 1)$.

C. Feedback Control and Complexity

At the $n^{th}$ iteration, given a state $x \in S$ and a martingale component $q$, to find a policy control $(v, c)$, we perform a Bellman update based on the approximated cost-to-go $J_{\nu}$ for the augmented state $(z, q)$. In addition, as discussed above, starting from any augmented state $(z, q)$ where $q = \mathcal{G}(z)$, we can solve the problem as if the failure probability were 1.0 and start using optimal control policies of the unconstrained problem from the state $z$.

The time complexity per iteration of the implementation in Algorithms 1-5 is $O\left(|S_{\nu}|^\theta |\log |S_{\nu}|\right)^2$ where $\theta \in (0, 1]$. The space complexity of the iMDP algorithm is $O(|S_{\nu}|)$ where $|S_{\nu}| = \Theta(n)$ due to our sampling strategy.

V. ANALYSIS

We present here the main analysis of the extended iMDP algorithm. The proof is discussed in [8], [25].

Theorem 3 Let $M_{\nu}$ and $M_{\nu^*}$ be two MDPs with discrete states constructed in $S$ and $S^*$ respectively, and let $J_{\nu^*} : S_{\nu^*} \rightarrow \mathbb{R}$ be the cost-to-go function returned by the extended iMDP algorithm at the $n^{th}$ iteration. Let us define $|b|_X = sup_{z \in X} b(z)$ as the sup-norm over a set $X$ of a function $b$ with a domain containing $X$. We have the following random variables converge in probability:

1. $\text{plim}_{n \rightarrow \infty} |J_{\nu}(\cdot, 1) - J_{\nu^*}(\cdot, 1)| = 0,$
The first four events construct the boundary values on $\partial D$ in probability, which leads to the probabilistically sound property of the extended iMDP algorithm. The last event asserts the asymptotically optimal property through the convergence of the approximating cost-to-go function $J_n$ to the optimal cost-to-go function $J^*$ on the augmented state space $\mathcal{S}$.

VI. EXPERIMENTS

We controlled a system with stochastic single integrator dynamics to a goal region with free ending time in a cluttered environment. The dynamics is given by $\dot{x}(t) = u(t)dt + Fdw(t)$ where $x(t) \in \mathbb{R}^2$, $u(t) \in \mathbb{R}^2$, $F = 0.5I_2$. The system stops when it collides with obstacles or reach the goal region. The cost function is the weighted sum of total energy spent to reach the goal $G$ at $(8, 8)$, which is measured as the integral of square of control magnitude, and a terminal cost, which is $-1000$ for the goal region $G$ and $10$ for the obstacle region $\Gamma$, with a discount factor $\alpha = 0.9$. The maximum velocity of the system in the $x$ and $y$ directions is one. The system starts from $(6.5, -3)$. Here, we use failure probability and collision probability interchangeably.

We first examine the boundary values for the stochastic target problem in Fig. 1. In particular, Figs. 1(a)-1(c) shows a policy map, cost value function $J_{4000,1.0}$ and the associated collision probability function $\gamma_{4000}$ for the unconstrained problem after 4000 iterations. Similarly, Figs. 1(d)-1(f) show a policy map, the associated cost function $J_{4000}$ and the min-collision probability function $\gamma_{4000}$ after 4000 iterations. For the unconstrained problem, the policy map encourages the system to go through the narrow corridors with low cost-to-go values and high probabilities of collision. In contrast, the policy map from the min-collision problem encourages the system to detour around the obstacles with high cost-to-go values and low probabilities of collision.

We now show how the extended iMDP algorithm constructs the sequence of approximating MDPs on the augmented state space $\mathcal{S}$. Figures 2(a)-2(c) show the corresponding anytime policies in $\mathcal{S}$ over iterations. In Fig. 2(c), we show the top-down view of a policy for states in $\mathcal{M}_{4000,3000}$. We observe that the system will try to avoid the narrow corridors when the risk tolerance is low. Next, Figs. 2(d)-2(f) show approximate cost-to-go $J_n$ when the probability threshold $\gamma_0$ is 1.0 for $n = 200$, 2000 and 4000. In the interior $D^n$, Figs. 2(d)-2(f) present the approximate cost-to-go $J_{4000}$ for augmented states where their martingale components are 0.1, 0.5 and 0.9. As we can see, the lower the martingale state is, the higher the cost value is.

In Fig. 3(a), we provide an example of controlled trajectories when the system starts from $(6.5, -3)$ with the failure probability threshold $\eta = 0.4$. In this figure, the min-collision probability function $\gamma_{4000}$ is plotted in blue, and the collision probability function $\gamma_{4000}$ in green. Starting from the augmented state $(6.5, -3, 0.40)$, the martingale state varies along controlled trajectories as a random parameter in a randomized control policy. When the martingale state is above $\gamma_{4000}$, the system follows a deterministic control policy of the unconstrained problem.

Similarly, in Fig. 4, we show controlled trajectories for different values of $\eta$ (0.01, 0.05, 0.40). In Figs. 4(a)-4(c), we show 50 trajectories resulting from a policy induced by $J_{4000}$ with different initial collision probability thresholds. In Figs. 4(d)-4(f), we show 5000 corresponding trajectories in the original state space $S$ with reported simulated collision probabilities and average costs in their captions. Trajectories that reach the goal region are plotted in blue, and trajectories that hit obstacles are plotted in red. The simulated collision probabilities and average costs for different $\eta$ are shown in Table I. As we expect, the lower the threshold is, the higher the average cost is.

More importantly, the simulated collision probabilities follow very closely the values of $\eta$ chosen at time 0. In Fig. 3(b), we plot these simulated probabilities for the first 4000 trajectories where $n \in [1, 5000]$ to show that the algorithm fully respects the bound failures probability. Thus, this observation indicates that the extended iMDP algorithm is able to manage the risk tolerance along trajectories in different executions to minimize the expected costs using feasible and time-consistent anytime policies.

VII. CONCLUSIONS

We have introduced and analyzed the extension of the incremental Markov Decision Process (iMDP) algorithm for stochastic optimal control in the presence of bounded probabilities of failure for initial states. We present a martingale approach to construct time-consistent control policies by diffusing the probability constraint into a martingale. We formulate and solve an equivalent stochastic target problem with sampled trajectories in the original and augmented state spaces. The algorithm guarantees the probabilistic soundness and asymptotic optimality of computed control policies as the number of iterations approaches infinity.

The future extension of the work is broad. We intend incorporate logical rules expressed as temporal logic constraints to achieve high degree of autonomy for systems to operate safely in uncertain environments with complex mission specifications. Extending the approach to solve stochastic games is another direction. We also plan to implement the proposed algorithm on several robotic platforms.

ACKNOWLEDGMENTS

This work was partially supported by the National Science Foundation grant CNS-1016213 and the Army Research Office MURI grant W911NF-11-1-0046.

REFERENCES

Fig. 1. Boundary values for the stochastic target problem. Figures 1(a)-1(c) shows a policy map, cost value function and the associated collision probability function for the unconstrained problem after 4000 iterations. Similar, Figures 1(d)-1(f) show a policy map, the associated value function, and the min-collision probability function after 4000 iterations.

Fig. 2. Sampling in the augmented state space $\mathcal{S}$. Figures 2(a)-2(c) show the anytime policies on $M_n$. In Fig. 2(c), we show the top-down view of a policy for states in $M_{3000}\backslash M_{3000}$. The system will try to avoid the narrow corridors when the risk tolerance is low. Figures 2(d)-2(f) present the approximate cost-to-go function $J_{4000}$ in $M_{4000}$ for augmented states where their martingale components are $0.1, 0.5$ and $0.9$ respectively.


In Fig. 3(a), the system starts from \((6.5, -3)\) with the failure-probability threshold \(\eta = 0.4\). When the martingale state is above \(\Upsilon\), the system follows a deterministic control policy obtained from the unconstrained problem. In Fig. 3(b), we show failure ratios for the first \(N\) trajectories \((1 \leq N \leq 5000)\) starting from \((6.5, -3)\) with different values of \(\eta\). Failure ratios follow very closely the values of \(\eta\), which indicates that the iMDP algorithm is able to provide solutions that are probabilistically sound.

In Figs. 4(a)-4(c), we show 50 trajectories resulting from a policy induced by \(J_{\text{MDP}}\) with different collision-probability thresholds \((\eta = 0.01, 0.05, 0.40)\). In Figs. 4(d)-4(f), we show 5000 corresponding trajectories in the original state space \(S\) with simulated collision probabilities and average costs in their captions. Successful and unsuccessful trajectories are plotted in blue and red respectively.