UAV Trajectory Planning with Probabilistic Geo-Fence via Iterative Chance-Constrained Optimization

Bin Du[1], Jun Chen[2], Dengfeng Sun[1], Satyanarayana Manyam[3], and David W. Casbeer[4],

***Abstract*—Chance-constrained optimization provides a promising framework for solving control and planning problems with uncertainties, due to its modeling capability to capture randomness in real-world applications. In this paper, we consider an UAV trajectory planning problem with probabilistic geo-fence, building on the chance-constrained optimization approach. In the considered problem, randomness of the model, such as the uncertain boundaries of geo-fences, is incorporated in the formulation. By solving the formulated chance-constrained optimization with a novel sampling based solution method, the optimal UAV trajectory is achieved while limiting the probability of collision with geo-fences to a prefixed threshold. Furthermore, to obtain a totally collision-free trajectory, i.e., avoiding the collision not only at the discrete time-steps but also in the entire time horizon, we build on the idea of an iterative scheme. That is, to iterate the solving of the chance-constrained optimization until the collision with probabilistic geo-fence is avoided at any time within the time horizon. At last, we validate the effectiveness of our method via numerical simulations.**

***Index Terms*—UAV, Trajectory Planning, Geo-Fence, Chance-Constrained Optimization.**

I. Introduction

Recently, trajectory planning has been a fundamentally crucial problem for applications of autonomous vehicles [1]–[4], especially the unmanned aerial vehicles (UAVs) [5]–[7]. The UAV trajectory planning usually corresponds to generating a geometric path in the specific space, while ensuring some conditions of the generated path for the UAV, such as the safety (obstacle/collision avoidance), minimum time or fuel consumption, etc. Among those underlying conditions, the assurance of safety is of particular importance, since failure to guarantee safety could lead to critical consequences. Meanwhile, the geospatial restriction, also known as *geo-fence*, helps to create the virtual boundaries of restricted areas where the UAV cannot pass through. The technique of geo-fencing undoubtedly provided aid to enforce the safety of UAV operations.

In fact, the problem of UAV trajectory planning with geospatial constraints has attracted increasingly significant attention in the past decades. To solve such trajectory planning problems, a large amount of approaches have been proposed in the literature, including the mixed-integer linear programming (MILP) [8], model predictive control (MPC) [9], potential field method (PFM) [10], etc. Among those various approaches, typical challenges that are primarily focused consist of the following three aspects.

1) *Non-convexity*: Due to the presence of restricted areas in the considered planning space, the safety region for UAVs is inherently non-convex, and so is the resulting optimization model.
2) *Uncertainty*: Consider that a number of sources of randomness, such as the modeling stochasticity and and environmental uncertainty, exist in the UAV trajectory planning problem. Such uncertainties have to be taken into account, in order to obtain the more reliable and robust UAV trajectory.
3) *Discretization issue*: When discrete-time methods are typically adopted to approach the trajectory planning problem, the obstacle/collision avoidance can be guaranteed only at the discrete time-steps. Therefore, a reasonable concern is how to ensure the UAV's safety between these discrete time-steps and obtain a so-called totally collision-free trajectory, i.e., avoiding obstacles at any time within the time horizon.

In order to deal with the non-convexity of the optimization problem, a common approach is to introduce a group of binary variables and reformulate the non-convex feasible region into a set of mixed-integer linear constraints by leveraging the technique of big-M constraint [11]–[13]. Although the resulting optimization model is still non-convex due to the presence of binary variables, however, such non-convexity can be easily handled by some commercial off-the-shelf solvers like CPLEX [14] and GUROBI [15].

In terms of the uncertainties, the majority of existing works focus on the stochasticity in the modeling of UAV's dynamics [16]–[21]. Taking advantages of assumptions that the system is linear and noise is white Gaussian, the distribution of UAV's states at any time can be represented by a propagation of Gaussian statistics. Subsequently, the linear chance constraints are reformulated into a totally deterministic form with respect to the mean and covariance of the UAV's random states [20], [21]. On this account, the standard optimization approaches can be applied to solve the problem. Nevertheless, considering that such reformulation introduces high non-linearity into the deterministic constraints, thus the computational complexity is still the concern for this type of methods. To further reduce the complexity, [22] particu-

[1]Bin Du is Ph.D. student, and Dengfeng Sun is Associate Professor, with School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN 47907, {du185,dsun}@purdue.edu

[2]Jun Chen is Assistant Professor, with Aerospace Engineering Department, San Diego State University, San Diego, CA 92182, {jun.chen}@sdsu.edu

[3]Satyanarayana Gupta Manyam is Research Scientist with Infoscitex Corp., a DCS company, and contractor for Autonomous Control Branch, Air Force Research Laboratories, WPAFB, OH 45433, msngupta@gmail.com

[4]David Casbeer is Research Scientist with Control Science Center of Excellence, Air Force Research Laboratory, WPAFB, OH 45433, david.casbeer@us.af.mil

DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited. Case number: 88ABW-2020-1820

larly proposed to enlarge the spherical collision regions into half spaces; however, it is easy to see that the introduced conservatism is the potential drawback of this scheme. On the other hand, among a few works that investigate the environmental uncertainties [23]–[25], the chance-constrained rapidly-exploring random trees method is studied to solve the trajectory planning problem. In particular, [23] combines the localization and motion uncertainties together, by assuming that both of them are Gaussian distributed. In this paper, we particularly focus on the environmental uncertainty, i.e., uncertain boundaries of the geo-fence, but consider it to follow a generic probability distribution, rather than restricting it to be the white Gaussian.

In addition, aiming at obtaining the totally collision-free trajectory, several paths are followed to address the discretization issue in the literature. First, notice that such issue only arises when the discrete-time approaches are adopted, thus a natural idea would be to take the finer discretization by increasing the number of discrete time-steps. However, there is no theoretic guarantee that how many time-steps are sufficient to ensure a collision-free trajectory. Besides, considering that the size of optimization problem quickly grows as the number of time-steps getting larger, thus the efficiency is the critical concern in this case. Second, an alternative solution is to expand the actual sizes of the restricted areas [8], [26], so that safety of the UAV can be also ensured between those discrete time-steps. Similarly, there is still no theoretic guarantee that how large the size of restricted areas should be expanded; or even worse, the optimization model could be infeasible if too much conservatism is imposed. Third, as suggested in recent works [20], [27], the collision-free trajectory can be also generated by ensuring the same side condition, when only the polygon is considered as the restricted areas. Nevertheless, as stated in [20] itself, the optimality of the obtained solution might be compromised in some basic cases.

In this paper, we study the UAV trajectory planning problem with all the aforementioned challenges considered. Concretely, our goal here is to generate the optimal UAV trajectory with the conditions specified as follows. Suppose that the UAV needs to plan its own trajectory that takes the UAV from a origin position to a destination, and meanwhile, the trajectory should be optimal in the sense of minimum control effort. In addition, it is assumed that the geo-fences are present in the planning space. To guarantee the safety, it should be ensured that the UAV can avoid the collision with those geo-fences when planning its trajectory. Moreover, the collision avoidance is supposed to be considered in a probabilistic manner, given that the geo-fence boundaries are subject to uncertainties.

It is worthy pointing out that the considered UAV trajectory planning problem, with all three aforementioned challenges involved, is essentially difficult to solve. Corresponding to these difficulties, our ideas here are i) introducing binary decision variables to deal with the non-convexity of feasible sets; ii) using chance constraints to capture the uncertainty of geo-fence boundaries; and iii) iterating the solving of chance-constrained optimization until a collision-free trajectory is obtained. More specifically, we model the uncertainty of geo-fence into the randomness of distance of boundaries, and ensure that the probability of collision avoidance has to be above a prefixed threshold. In terms of the non-convex feasible set, we first represent the geo-fence as a generic polygon, defined by a group of linear inequalities, and then adopt the idea of big-M constraint to enforce that at least one inequality is violated. This will give us a chance-constrained mixed-integer quadratic programming (CC-MIQP) problem. To approach this CC-MIQP problem, we develop a new sampling based solution method. As distinct from the existing sampling or scenario based approaches, our method leverages the specific structure of the considered chance-constrained problem and builds on a novel samples selecting procedure. By designing an appropriate selecting criteria which picks only a part of the sampled scenarios, we guarantee that 1) the solution obtained by solving an approximated deterministic problem has to be feasible to the original chance-constrained optimization; and 2) the approximated solution can be arbitrarily close to the true optimal one as the sample size increases. Furthermore, in order to guarantee the UAV's safety between discrete time-steps, we build on the idea of iterative optimization scheme [8]. That is, to solve the CC-MIQP problems and check whether the resulting trajectory is collision-free. If not, we iterate the solving-and-checking procedure until the desired trajectory is achieved. As a result, such a chance-constrained optimization based method is less conservative than some robust approaches [28], [29], since only the worst-case realization of uncertainty is usually considered in the latter approaches. Meanwhile, by the iterative scheme, the collision avoidance is guaranteed within the entire trajectory rather than on the discrete time-steps.

The rest of this paper is organized as follows. Sec. II-A first presents the basic CC-MIQP formulation, i.e., the safety of the UAV is guaranteed only at the discrete time-steps. Sec. II-B develops a novel sampling based solution method to solve the basic CC-MIQP problem. Feasibility and optimality of the obtained solution are discussed in this Sec. II-C. The iterative scheme to obtain the collision-free trajectory is provided in Sec. III. Sec. IV validates the proposed method by a set of simulation results. Sec. V finally concludes this paper. For the reader's convenience, the proofs of two propositions are provided in Appendix.

II. The Basic CC-MIQP: Collision Avoidance at Discrete Time-Steps

For the sake of presentation, we start with the basic CC-MIQP formulation, in which the collision avoidance is ensured only at discrete time-steps. Behaviors of the UAV between the discrete time-steps will be investigated later.

A. The Basic CC-MIQP

To simplify the problem formulation, let us assume that the UAV is located in a 2-D plane. The position of the UAV at each discrete time-step t is then represented as $\mathbf{x}(t) \in \mathbb{R}^2$. We study the UAV trajectory planning problem within a finite-time horizon T, i.e., totally T time-steps are considered and thus each discrete time-step t has $t \in \mathcal{T} := \{0, 1, \cdots, T\}$. Note that the subsequently developed trajectory planning approach

can be extended to the 3-D case without much effort. In addition, we denote the control input of the UAV as $\mathbf{u}(t) \in \mathbb{R}^2$. The constraints and objective function of the UAV trajectory planning problem are stated as follows.

1) UAV Dynamics: Besides the UAV's position $\mathbf{x}(t)$, we further take its velocity, denoted as $\mathbf{v}(t) \in \mathbb{R}^2$, into consideration. The dynamics of the UAV is modeled as the following linear time-invariant (LTI) system,

$$\begin{bmatrix}\mathbf{x}(t+1)\\ \mathbf{v}(t+1)\end{bmatrix} = A\begin{bmatrix}\mathbf{x}(t)\\ \mathbf{v}(t)\end{bmatrix} + B\mathbf{u}(t), \tag{1}$$

where the matrices $A \in \mathbb{R}^{4\times 4}$ and $B \in \mathbb{R}^{4\times 2}$ are system parameters. Concretely, in the 2-D case, we denote the UAV's position and velocity as

$$\mathbf{x}(t) = [p_x(t), p_y(t)]^\top \quad \text{and} \quad \mathbf{v}(t) = [v_x(t), v_y(t)]^\top, \tag{2}$$

where the subscripts x and y represent the position or velocity along x- and y-axis, respectively. In this paper, it is assumed that the dynamics of the UAV follows a simple double-integrator [20], i.e., the control input $\mathbf{u}(t)$ represents the acceleration of the UAV. Thus, the system parameters are given as,

$$A = \begin{bmatrix}1 & 0 & \Delta t & 0\\ 0 & 1 & 0 & \Delta t\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}, \text{ and } B = \begin{bmatrix}\Delta t^2/2 & 0\\ 0 & \Delta t^2/2\\ \Delta t & 0\\ 0 & \Delta t\end{bmatrix}, \tag{3}$$

where Δt is the time interval between two consecutive discrete time-steps.

2) Origin-Destination Assignment: Considering that the UAV is required to depart from a specific starting position $\mathbf{x}^{\text{orig}} \in \mathbb{R}^2$, and arrive at the destination $\mathbf{x}^{\text{dest}} \in \mathbb{R}^2$ at the end of time horizon, thus the origin-destination assignment constraint reads

$$\mathbf{x}(0) = \mathbf{x}^{\text{orig}}, \quad \text{and} \quad \mathbf{x}(T) = \mathbf{x}^{\text{dest}}. \tag{4}$$

3) Bounded Planning Space: In addition, it is assumed that the entire trajectory of the UAV is restricted to be inside of a bounded planning space. We define such space, denoted as a set $\mathcal{K} \subset \mathbb{R}^2$, by a group of linear inequality constraints, i.e., $\mathcal{K} := \{\mathbf{x} \,|\, L\mathbf{x} \le \mathbf{b}\}$. Therefore, the bounded planning space constraint is represented as

$$\mathbf{x}(t) \in \mathcal{K}, \ \forall t \in \mathcal{T}. \tag{5}$$

4) Geo-Fence Restriction: Suppose that there are totally I geo-fences considered and each of them is represented as a generic polygon $\mathcal{S}_i$ where the index $i \in \mathcal{I} := \{1, 2, \cdots, I\}$. We model each of the geo-fences $\mathcal{S}_i$ as follows. Let us represent the center of $\mathcal{S}_i$ as a fixed point $\mathbf{p}_i \in \mathbb{R}^2$, then each side, indexed by $k \in \{1, 2, \cdots, K_i\}$, corresponds to a boundary of the geo-fence; see the solid lines in Fig. 1. Note that each side k can be uniquely determined by its perpendicular distance r_i^k to the point $\mathbf{p}_i$ together with the corresponding angle θ_i^k, as shown in Fig. 1. With the help of the perpendicular distance r_i^k and angle θ_i^k, each boundary k of the polygon can be analytically represented as,

$$\begin{bmatrix}\cos\left(\theta_i^k\right)\\ \sin\left(\theta_i^k\right)\end{bmatrix}^\top \left(\mathbf{x} - \mathbf{p}_i\right) - r_i^k = 0, \ \forall k \in \{1, 2, \cdots, K_i\}. \tag{6}$$

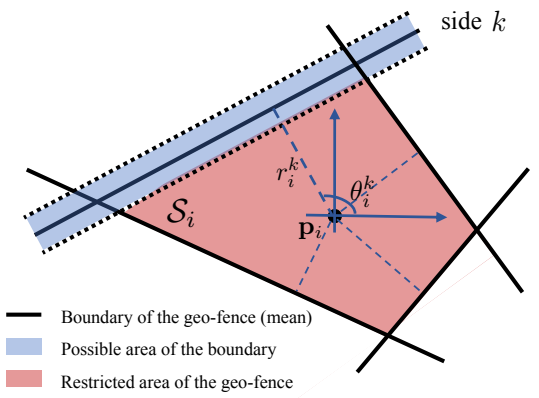


Fig. 1: Illustration of the geo-fence $\mathcal{S}_i$

Consequently, the restricted area defined by the geo-fence $\mathcal{S}_i$ with K_i boundaries is,

$$\mathcal{S}_i := \left\{\mathbf{x} \;\middle|\; \begin{bmatrix}\cos\left(\theta_i^k\right)\\ \sin\left(\theta_i^k\right)\end{bmatrix}^\top \left(\mathbf{x} - \mathbf{p}_i\right) < r_i^k, \right. \\ \left. \forall k \in \{1, 2, \cdots, K_i\}, \, \mathbf{x} \in \mathbb{R}^2 \right\}. \tag{7}$$

In order to guarantee the safety, the UAV has to avoid these restricted areas when planning its trajectory, i.e.,

$$\mathbf{x}(t) \notin \mathcal{S}_i, \quad \forall i \in \mathcal{I}, \ t \in \mathcal{T}. \tag{8}$$

Next, we incorporate uncertainties into the model of geo-fence. Here, instead of assuming the distances r_i^k's to be deterministic constants, we consider them as a set of random variables, following some known probability distributions. As shown in Fig. 1, we use the shadowed area to denote all possible positions of the boundary; given this definition, the solid line given in (6) represents the mean value of the random distance. In addition, it should be noted that we only illustrate the uncertainty of the k-th boundary in Fig. 1, but all boundaries are subject to uncertainties in our problem setup. Subsequently, the geo-fence restriction is ensured by the following chance constraint, with a prefixed probability threshold $\alpha \in (0, 1)$,

$$\mathbf{Pr}\Big(\mathbf{x}(t) \notin \mathcal{S}_i^{\mathbf{r}}, \ \forall i \in \mathcal{I}\Big) \ge \alpha, \ \forall t \in \mathcal{T}. \tag{9}$$

Note that here we use the notation $\mathcal{S}_i^{\mathbf{r}}$, that follows a definition similar to (7), to represent the uncertainty of the geo-fence due to the random distances r_i^k's. $\mathbf{Pr}(\cdot)$ is the probability measure of these random variables. The intuition behind these chance constraints (9) are quite straight-forward. If there are uncertainties in the position of the geo-fence $\mathcal{S}_i^{\mathbf{r}}$, it is too conservative (or even infeasible) to enforce the UAV's position to be outside of $\mathcal{S}_i^{\mathbf{r}}$ for any realization of the random variables. An alternative to the conservative approach is to let the constraint violation be tolerated, but the probability of violation is constrained to be no larger than a specified threshold, $1 - \alpha$.

It is worthy pointing out that the restriction in the chance constraint (9), i.e., $\mathbf{x}(t) \notin \mathcal{S}_i^{\mathbf{r}}$, inherently defines a non-convex feasible region for decision variables, since the set $\mathcal{S}_i^{\mathbf{r}}$ itself is

convex for any realization of r_i^k's. In order to address this non-convexity issue, we further adopt the idea of big-M constraint and build on the technique of mixed-integer programming. Recall that each geo-fence is modeled by the polygon $\mathcal{S}_i^{\mathbf{r}}$, as defined in (7). Then, the feasible region for the UAV with respect to the geo-fence $\mathcal{S}_i^{\mathbf{r}}$ can be represented as the outside of the polygon, i.e.,

$$\mathcal{C}_i^{\mathbf{r}} := \Bigg\{ \mathbf{x} \;\Bigg|\; \text{there exists } k \in \{1,2,\cdots,K_i\} \text{ such that } \begin{bmatrix} \cos\left(\theta_i^k\right) \\ \sin\left(\theta_i^k\right) \end{bmatrix}^\top \Big(\mathbf{x} - \mathbf{p}_i\Big) \geq r_i^k,\ \mathbf{x} \in \mathbb{R}^2 \Bigg\}. \tag{10}$$

To further tackle with the existence condition of k in the definition of $\mathcal{C}_i^{\mathbf{r}}$, we introduce a group of binary variables $z_i^k \in \{0,1\}$ with $k \in \{1,2,\cdots,K_i\}$. Consequently, the set $\mathcal{C}_i^{\mathbf{r}}$ is equivalent to

$$\mathcal{C}_i^{\mathbf{r}} = \Bigg\{ \mathbf{x} \;\Bigg|\; \begin{bmatrix} \cos\left(\theta_i^k\right) \\ \sin\left(\theta_i^k\right) \end{bmatrix}^\top \Big(\mathbf{x} - \mathbf{p}_i\Big) + M\big(1 - z_i^k\big) \geq r_i^k, \sum_{k=1}^{K_i} z_i^k \geq 1,\ z_i^k \in \{0,1\},\ \mathbf{x} \in \mathbb{R}^2 \Bigg\}. \tag{11}$$

where M is a sufficiently large constant. As a result, the chance constraint (9) is now rewritten as,

$$\mathbf{Pr}\Big(\mathbf{x}(t) \in \mathcal{C}_i^{\mathbf{r}},\ \forall i \in \mathcal{I}\Big) \geq \alpha,\ \forall t \in \mathcal{T}. \tag{12}$$

5) Control Effort Minimization: In this paper, we consider to minimize the total cost of control effort and meanwhile to ensure the safety of the UAV. Thus, the overall optimization problem is formulated as,

$$\begin{aligned} \underset{\{\mathbf{u}(t)\}_{t\in\mathcal{T}}}{\text{minimize}} \quad & \sum_{t=0}^{T-1} \|\mathbf{u}(t)\|^2, \\ \text{subject to} \quad & (1), (4), (5), \text{ and } (12). \end{aligned} \tag{13}$$

Note that the problem (13) is basically a mixed-integer program involved with quadratic objective function and chance constraints. With a slight abuse of the name, we refer to it as the chance-constrained mixed-integer quadratic programming (CC-MIQP) problem. Next, we will develop a sampling based solution method to solve such CC-MIQP problem.

B. Sampling Based Solution Method

Before proceeding to the development of our method, let us make a few remarks on the basic CC-MIQP formulation (13). First, it should be noted that the probability of chance constraint is jointly applied to all geo-fences $\mathcal{S}_i, i \in \mathcal{I}$, as shown in (12). Alternatively, one could also consider each geo-fence individually, and thus the chance constraint reads,

$$\mathbf{Pr}\Big(\mathbf{x}(t) \in \mathcal{C}_i^{\mathbf{r}}\Big) \geq \alpha_i,\ \forall i \in \mathcal{I},\ t \in \mathcal{T}. \tag{14}$$

In this case, the resulting problem becomes much easier to solve, since each individual chance constraint in (14) can be equivalently rewritten into the following deterministic form,

$$\begin{bmatrix} \cos\left(\theta_i^k\right) \\ \sin\left(\theta_i^k\right) \end{bmatrix}^\top \Big(\mathbf{x}(t) - \mathbf{p}_i\Big) + M\big(1 - z_i^k(t)\big) \geq q_{i,k}(\alpha_i), \sum_{k=1}^{K_i} z_i^k(t) \geq 1,\ z_i^k(t) \in \{0,1\},\ \forall k \in \{1,2,\cdots,K_i\}, t \in \mathcal{T}. \tag{15}$$

Here, $q_{i,k}(\alpha_i)$ is the α_i-quantile of the individual random variable r_i^k, i.e., $\mathbf{Pr}\big(r_i^k \leq q_{i,k}(\alpha_i)\big) = \alpha_i$. Nevertheless, in order to ensure the joint probability α by enforcing the individual chance constraints, the corresponding α_i's should be considerably greater than α, especially when the number of geo-fences I is large. Second, notice that another approach is to utilize the convexity of feasible regions when the distribution of random variables enjoys some specific properties [30]–[32]. In our setup, however, it is unlikely to maintain the convexity property or even build the convex approximations of the non-convex chance constraint, since binary variables z_i^k's are involved in the formulation. Here, we consider an approximation of the basic CC-MIQP problem, in which the continuous distribution of random variables is replaced by some specifically selected samples based on the Monte Carlo sampling. The resulting approximation problem is formulated as a deterministic mixed-integer quadratic program and thus can be easily handled by commercial solvers, such as CPLEX [14] and GUROBI [15]. More importantly, such a scheme also provides promising candidate solutions with feasibility guaranteed to the original chance-constrained problem, as shown later on.

For the sake of simplicity of notation, here we stack the uncertain distance r_i^k's all together and denote it as a random vector $\mathbf{r} = [r_1^1, r_1^2 \cdots, r_1^{K_1}, \cdots, r_I^1, r_I^2, \cdots, r_I^{K_I}] \in \mathbb{R}^K$ with the dimension $K = \sum_{i=1}^I K_i$. It is assumed that the random vector $\mathbf{r}$ follows a known probability distribution and its joint cumulative distribution function (CDF) is given by,

$$F(\mathbf{y}) := \mathbf{Pr}\Big(r_i^k \leq y_i^k,\ \forall k \in \{1,2,\cdots,K_i\}, i \in \mathcal{I}\Big). \tag{16}$$

Suppose that a group of N independent samples, also called scenarios, is drawn based on a uniform distribution $\mathbf{U}(\mathbf{0}, \mathbf{r}^{\max})$. Let us denote each of the samples as $\mathbf{r}(n)$ and thus the set of N samples is $\mathcal{R} = \{\mathbf{r}(1), \mathbf{r}(2), \cdots, \mathbf{r}(N)\}$. Note that here we require each sample n to satisfy with $0 \leq \mathbf{r}(n) \leq \mathbf{r}^{\max}$ element-wise, since too large samples are meaningless for solving the problem, as we can see later on. In fact, the upper bound of samples $\mathbf{r}^{\max}$ can be calculated before hand, according to the center of geo-fences together with the bounded planning space as defined in (5).

Starting from the samples which are independently and identically distributed (*i.i.d.*), a number of scenario based approaches are proposed in the literature [33]–[36], where the true distribution of $\mathbf{r}$ is replaced by the N sampled scenarios, and consequently, the chance constraint becomes N individual deterministic constraints. The idea behind these scenario-based approaches is quite intuitive. That is, when the number of samples is large enough and the obtained

solution is feasible for all of these samples, it is highly possible (with probability greater than α) that the solution is also feasible to the original chance-constrained problem. However, as pointed out in [36], finding the solution satisfying all sampled scenarios might be conservative, given that the constraint violation can be tolerated with probability $1-\alpha$ in the original problem setup. Motivated by this, we next develop the solution method with reduced conservatism, compared to the existing scenario-based approaches. Our main idea here is, instead of imposing all N sampled scenarios from the set $\mathcal{R}$, we specifically select a part of them such that the following two conditions are satisfied. First, each of the selected samples can individually result in a solution with feasibility guaranteed to the original chance-constrained problem. Second, all the selected samples can jointly reflect the constraint violation tolerance with probability $1-\alpha$. By doing so, we are expected to develop an approach which is capable of picking only one of the selected samples such as the obtained solution is feasible and also with the least conservatism. Note that the feasibility of solution can be automatically guaranteed by the first aforementioned condition of the selected samples. The problem now becomes how to enforce the least conservative solution by involving the second condition.

Consider that the random variable r_i^k appears separately on the right-hand side of inequalities in the chance constraint; see the definition (11). This implies that the probability of obtaining a feasible trajectory, i.e., located outside of the uncertain geo-fences, will be greater when the distance r_i^k has the large value. Such an observation prompts us to select the sampled scenario $\mathbf{r}(n)$ which has larger $r_i^k(n)$'s. Precisely, we determine whether a sampled scenario should be selected by the following criteria: a sampled scenario $\mathbf{r}(n)$ is said to be selected, if its joint CDF satisfies $F\big(\mathbf{r}(n)\big) \geq \alpha$. Note that we here assume the joint CDF, as shown in (16), can be obtained as prior knowledge, since the distribution of random vector $\mathbf{r}$ is known. Even though the random vector could follow any generic distribution, yet we remark that its CDF can be still approximated before hand through the Monte Carlo simulation [37]. By following such a selecting criteria of samples, it will be proved in the next subsection that each selected sample can yield a feasible solution to the original chance-constrained optimization, i.e., the first condition is satisfied. To consider the second condition, we leverage the nature of minimization problem.

Suppose that totally S samples are selected based on the criteria, then we denote the selected subset of samples as $\tilde{\mathcal{R}}^S = \{\tilde{\mathbf{r}}(1), \tilde{\mathbf{r}}(2), \cdots, \tilde{\mathbf{r}}(S)\} \subseteq \mathcal{R}$. Subsequently, we substitute the original random vector $\mathbf{r}$ in the chance constraint by the selected samples $\tilde{\mathcal{R}}^S$ and enforce that at least one sample from the set $\tilde{\mathcal{R}}^S$ is satisfied when solving the problem. Therefore, the chance constraint is replaced by the following deterministic one,

$$\mathbf{x}(t) \in \mathcal{C}_i^{\tilde{\mathbf{r}}(s)},\ \exists s \in \{1, 2, \cdots, S\},\ \forall i \in \mathcal{I},\ t \in \mathcal{T}. \tag{17}$$

In order to deal with the existence condition in (17), we follow the same path as before, i.e., introducing a group of binary variables $w(s) \in \{0,1\},\ s \in \{1, 2, \cdots, S\}$. As a result, the explicit form of geo-fence constraint becomes, for $t \in \mathcal{T}$,

$$\begin{aligned}
&\begin{bmatrix}\cos\left(\theta_i^k\right)\\ \sin\left(\theta_i^k\right)\end{bmatrix}^\top \Big(\mathbf{x}(t) - \mathbf{p}_i\Big) + M\big(1 - z_i^k(t)\big) \geq \sum_{s=1}^{S} w(s)\tilde{r}_i^k(s),\\
&\sum_{k=1}^{K_i} z_i^k(t) \geq 1,\ \sum_{s=1}^{S} w(s) = 1,\ z_i^k(t),\ w(s) \in \{0,1\},\ \forall i \in \mathcal{I}.
\end{aligned} \tag{18}$$

Here, we use $\tilde{r}_i^k(s)$ to denote the corresponding component of the selected scenario $\tilde{\mathbf{r}}(s)$. Next, we alternatively solve the following deterministic optimization to approach the solution of the original CC-MIQP (13),

$$\begin{aligned}
\underset{\{\mathbf{u}(t)\}_{t\in\mathcal{T}}}{\text{minimize}} \quad & \sum_{t=0}^{T-1} \|\mathbf{u}(t)\|^2,\\
\text{subject to} \quad & (1),\ (4),\ (5),\ \text{and}\ (18).
\end{aligned} \tag{19}$$

We remark that, since a minimization problem is considered and only one of the selected samples will be active, it can be verified that the optimization tends to activate the selected sample $\tilde{\mathbf{r}}(s)$ which has relatively smaller $\tilde{r}_i^k(s)$'s. On this basis, the second of the aforementioned conditions is also expected to be satisfied, since less conservatism will be imposed when the distance r_i^k has the smaller value.

C. Performance Analysis

It is worthy noting that, with the help of the selected subset of samples $\tilde{\mathcal{R}}^S$, the original chance constraint (12) has been suitably approximated by a set of deterministic mixed-integer constraints (18). In order to validate the effectiveness of such approximation, we here provide the guarantee of its performance by showing the following two propositions. Note that those two propositions also correspond to the two aforementioned conditions of the selected samples. Before proceeding to the propositions, let us first introduce two additional sets which are defined by the chance constraint (12) and its deterministic approximate (18), respectively. We denote the feasible set $\mathcal{X}$ induced by the chance constraint (12) as

$$\mathcal{X} := \left\{\mathbf{x} \;\middle|\; \mathbf{Pr}\Big(\mathbf{x} \in \mathcal{C}_i^{\mathbf{r}},\ \forall i \in \mathcal{I}\Big) \geq \alpha\right\}, \tag{20}$$

and denote its approximation set $\tilde{\mathcal{X}}(\tilde{\mathcal{R}}^S)$ induced by (18) as

$$\tilde{\mathcal{X}}(\tilde{\mathcal{R}}^S) := \left\{\mathbf{x} \;\middle|\; (18) \text{ is satisfied, by given } \tilde{\mathcal{R}}^S \right\}. \tag{21}$$

Proposition 1: For any non-empty set of selected samples $\tilde{\mathcal{R}}^S$, i.e., $S \geq 1$, it holds that $\tilde{\mathcal{X}}(\tilde{\mathcal{R}}^S) \subseteq \mathcal{X}$.

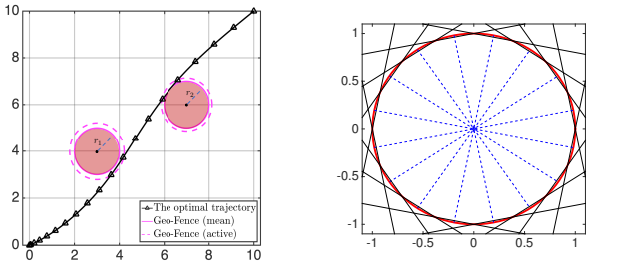
Proof: See Appendix VI-A. ■

It should be remarked that the above Proposition 1 guarantees that the solution obtained by solving (19) is also feasible to the original chance-constrained optimization (13). To further investigate the optimality of the obtained solution, we show the next proposition.

Proposition 2: Suppose that the samples $\mathcal{R}$ are independently and identically distributed, and let the number of selected samples be $S \to \infty$, then it holds that

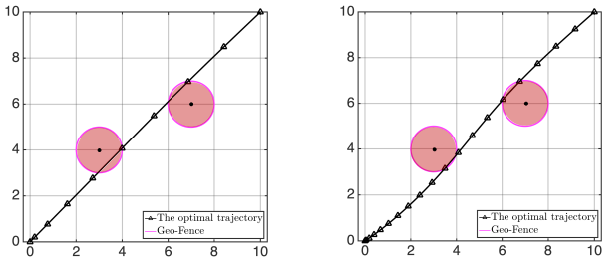
$$\lim_{S\to\infty} \mathbf{Pr}\left(\mathcal{X} \subseteq \tilde{\mathcal{X}}(\tilde{\mathcal{R}}^S)\right) = 1. \tag{22}$$

Proof: See Appendix VI-B. ■



(a) The UAV's trajectory (b) 16-sided polygon approximate

Fig. 2: Trajectory of the UAV with probabilistic geo-fence



(a) Trajectory with $T = 10$ (b) Trajectory with $T = 20$

Fig. 3: Demonstration of the discretization issue

To sum up, considering the challenge to deal with uncertainties involved in the probabilistic geo-fences, we here proposed to solve the approximated deterministic optimization (19) instead of the chance-constrained problem (13) directly. It is theoretically guaranteed by Proposition 1 that the obtained solution must be feasible to the original CC-MIQP (13). In addition, it is expected by Proposition 2 that the approximated solution can be arbitrarily close to the true optimal one, as the sample size increases. Furthermore, we call the sample s active scenario when its corresponding binary variable has $w(s) = 1$ after solving the deterministic problem (19). The determination of this active scenario helps to identify the actual restricted area of the geo-fence that is considered in the obtained solution.

With the formulation of CC-MIQP specified and the solution method presented, we now provide an illustrative example obtained by the preliminary simulation. Let us here consider the trajectory planning problem from the origin $\mathbf{x}^{\text{orig}} = [0, 0]^T$ to the destination $\mathbf{x}^{\text{dest}} = [10, 10]^T$. The finite-time horizon is uniformly discretized into $T = 20$ time-steps with the time interval $\Delta t = 1$. Suppose that there are two geo-fences in the 2-D plane which are centering at points $\mathbf{p}_1 = [3, 4]^\top$ and $\mathbf{p}_2 = [7, 6]^\top$, respectively. Their restricted areas are modeled as two disks which have the uncertain radii (r_1, r_2), as shown in Fig. 2(a). In our simulation, it is assumed that the random variable pair (r_1, r_2) follows the Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean $\boldsymbol{\mu} = [1, 1]^\top$ and covariance $\boldsymbol{\Sigma} = [0.01, 0; 0, 0.01]$. The prefixed probability threshold of chance constraint is set as $\alpha = 0.9$. In addition, the UAV dynamics follows the LTI system as specified in (1) and (3). Note that we use the GUROBI 9.0 package to solve the resulting deterministic mixed-integer programs (19).

To suit the model of CC-MIQP formulated in Sec. II-A, we first approximate the disk-like geo-fence by a 16-sided polygon, shown in Fig. 2(b). Note that each polygon approximation can be represented as the set $\mathcal{S}_i$, defined in (7), with the angles $\theta_i^k = (2k - 1)\pi/16$, $k \in \{1, 2, \cdots, 16\}$. Next, to apply the sampling based solution method, we draw $N = 1000$ *i.i.d.* samples on the random variable (r_1, r_2) and $S = 614$ of them are selected according to the joint CDF. The obtained optimal trajectory of the UAV is shown as Fig. 2. We note that the boundaries with solid lines have radii $r_1 = r_2 = 1$, which is the mean of distribution, while the ones with dashed lines represent the active scenario which has radii $r_1 = 1.21$ and $r_2 = 1.14$. It is easy to verify that the actual radius of the geo-fences is expected to be larger than the mean, since the chance constraint is required to be satisfied with probability at least $\alpha = 0.9$. This is consistent with the observation as shown in Fig. 2(a).

III. The Iterative CC-MIQP: Achieving the Collision-Free Trajectory

Recall that, in the basic CC-MIQP formulation, the UAV's safety is ensured only at the discrete time-steps $t \in \mathcal{T}$, no matter in the original chance constraint (12) or its deterministic approximation (18). In this case, however, the UAV might be still under risk between the time-steps, due to the potential discretization issue. In this section, we aim to address such discretization issue and obtain a totally collision-free trajectory, i.e., avoiding the geo-fence not only at the discrete time-steps but also in the entire time horizon.

Before continuing, let us first demonstrate the potential discretization issue of the UAV's trajectory, by two simple examples. For the sake of illustration, here we assume that each geo-fence $\mathcal{S}_i$ in both examples has the deterministic boundary. As shown in Fig. 3, the geo-fences are two disks which are centering at $[3, 4]^\top$ and $[7, 6]^\top$, respectively, and have the fixed radius $r_i = 1$. The finite-time horizon is uniformly discretized with $T_1 = 10$ and $T_2 = 20$ time-steps, and the time interval is $\Delta t = 1$. It can be observed from both Fig. 3(a) and Fig. 3(b) that, although the UAV's position at each discrete time-step is out of the disks, yet the obtained trajectories are still unsafe as both of them traverse the restricted area of geo-fence. To further resolve the illustrated discretization issue, we here adopt the idea of iterative scheme, as suggested in [8]. The guideline of this approach is to consider the control time-steps and collision avoidance time-steps separately, so that the collision avoidance time-steps can be independently added iteratively, until the obtained trajectory is totally collision-free.

To differentiate the discretization for the control of the UAV and also the collision avoidance, we denote T_u as the number of control time-steps and T_s as the number of collision avoidance time-steps. Note that, while the control time-steps can be specified as same as before, the collision avoidance time-steps are no longer assumed to be uniformly discretized. Instead, we denote $t_s[k]$ with $k \in \{1, 2, \cdots, T_s\}$ as the collision avoidance time-steps and use $\mathcal{T}_s := \{t_s[k], k \in \{1, 2, \cdots, T_s\}\}$ to represent the set of these time-steps.

With these two types of discrete time-steps considered independently, the basic CC-MIQP formulation becomes,

$$\begin{aligned} \underset{\mathbf{u}(t)}{\text{minimize}} \quad & \sum_{t=0}^{T_u-1} \|\mathbf{u}(t)\|^2 \\ \text{subject to} \quad & (1),\ (4),\ (5),\ \text{and} \\ & \mathbf{Pr}\Big(\mathbf{x}(t_s[k]) \in \mathcal{C}_i^{\mathbf{r}}\Big) \geq \alpha,\ \forall t_s[k] \in \mathcal{T}_s. \end{aligned} \tag{23}$$

In the formulation (23), we control the UAV by determining the input $\mathbf{u}(t)$ at control time-steps $t \in \{0, 1, \cdots, T_u - 1\}$, and meanwhile ensure the safety of the UAV at collision avoidance time-steps $t_s[k] \in \mathcal{T}_s$. It should be noted that, with the UAV's dynamics modeled as the LTI system (1), its state at any time τ (including the position $\mathbf{x}(\tau)$ and velocity $\mathbf{v}(\tau)$) can be exactly computed as follows. Without loss of generality, we assume that the time τ lies in the interval $[t, t+1]$ and the state of UAV at time t is known, then according to the UAV's dynamics, $\mathbf{x}(\tau)$ and $\mathbf{v}(\tau)$ can be obtained as

$$\mathbf{x}(\tau) = \mathbf{x}(t) + (\tau - t)\mathbf{v}(t) + 0.5(\tau - t)^2\mathbf{u}(t); \tag{24}$$

$$\mathbf{v}(\tau) = \mathbf{v}(t) + (\tau - t)\mathbf{u}(t). \tag{25}$$

Therefore, once the set of collision avoidance time-steps $\mathcal{T}_s$ is fixed, the optimization problem (23) can be solved by applying the solution method proposed in Sec. II-B.

We are now in the position to address the aforementioned discretization issue. The idea here is to solve the optimization problem (23) with just a few collision avoidance time-steps at the very beginning, and then check whether the obtained trajectory is collision-free or not. If not, we identify the unsafe traverses by a safety check process, and compute the corresponding time interval $[t_l^i, t_u^i]$ for each unsafe traverse i. Note that here t_l^i and t_u^i represent the time when the UAV enters and leaves the geo-fence, respectively. Next, we augment the previous optimization problem by adding the new collision avoidance time-step $t_s^{\text{new}} = (t_l^i + t_u^i)/2$ into the set $\mathcal{T}_s$ and solve the augmented optimization again. Such a procedure is repeated until a totally collision-free trajectory is obtained. To sum up, we outline such iterative scheme as the following Algorithm 1 and provide a few remarks on the algorithm.

Remark 1: It should be noted that the detection of unsafe traverses and computation of traverse time intervals can be accomplished efficiently by using a simple bisection routine. Although such procedure still relies on the discretization of the UAV's continuous time dynamics, yet we argue that the aforementioned discretization issue is no longer the concern in this case. In addition, considering that no optimization problem is involved in the detection and computation of unsafe traverses, the finer discretization by increasing the number of sampling times is implementable here since only a computing-and-checking procedure is needed. For the details of schemes on how to detect the unsafe traverses and also compute the traverse time intervals, we refer the interested reader to [8].

Remark 2: We emphasize that, in our probabilistic setting, the detection and computation of unsafe traverses are also subject to the uncertainty of geo-fences, since they are inherently dependent on the random distances of boundaries.

Algorithm 1: Trajectory planning via iterative CC-MIQP

Data: Initialize the set of collision avoidance time-steps as $\mathcal{T}_s = \emptyset$. Solve the CC-MIQP problem (23), and obtain the trajectory.

while the obtained trajectory is NOT collision-free **do**

(S.1) For each unsafe traverse i, compute traverse time interval $[t_l^i, t_u^i]$;

(S.2) Augment the basic CC-MIQP formulation by adding the new time-steps into $\mathcal{T}_s$,

$$\mathcal{T}_s \longleftarrow \mathcal{T}_s \bigcup \{\frac{t_l^i + t_u^i}{2}\},\ \forall i; \tag{26}$$

(S.3) Solve the augmented CC-MIQP problem by the sampling based solution method, and obtain the new trajectory;

(S.4) Continue.

end

Recall the sampling based solution method that we use to solve the CC-MIQP problem, there are only one active scenario which determines the obtained trajectory and such scenario is indicated by the value of binary variables $w(s)$ after solving the optimization. On this account, we can realize the detection of the unsafe traverse and computation of the traverse time intervals by only considering the obtained active scenario.

IV. Numerical Example

In this section, we first validate the proposed iterative CC-MIQP method, by showing the collision-free trajectory planning result. Furthermore, we evaluate how the performance of our method will be affected by the choices of probability threshold α and sample size N.

A. Demonstration of the Collision-Free Trajectory

The simulation setup is the same as before. We consider to plan the UAV's trajectory from the origin $\mathbf{x}^{\text{orig}} = [0, 0]^\top$ to the destination $\mathbf{x}^{\text{dest}} = [10, 10]^T$. The UAV dynamics follows the LTI system (1) with system parameters specified as (3). The control time-steps are uniformly discretized with $T_u = 20$ and the time interval is $\Delta t = 1$. In this simulation, we consider two geo-fences centering at points $\mathbf{p}_1 = [2.0, 5.5]^\top$ and $\mathbf{p}_2 = [7.0, 6.0]^\top$ in the 2-D plane, and each geo-fence is assumed to have a rectangular restricted area, as shown in Fig. 4. Concretely, we use h_i and w_i to denote the distances from the point $\mathbf{p}_i$ to the horizontal and vertical boundaries, respectively. Suppose that the random variable pair (h_1, w_1) follows a joint Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ with mean $\boldsymbol{\mu}_1 = [3.5, 0.5]^T$ and covariance $\boldsymbol{\Sigma}_1 = [0.01, 0; 0; 0.01]$, and the second pair has $(h_2, w_2) \sim \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ with mean $\boldsymbol{\mu}_2 = [2.0, 0.4]^T$ and covariance $\boldsymbol{\Sigma}_2 = [0.01, 0; 0; 0.01]$. The probability threshold of chance constraints is set as $\alpha = 0.9$. In order to apply the sampling based solution method to solve the chance-constrained optimization problems, here we draw $N = 1000$ *i.i.d.* sample on the random variables h_i and w_i; and based on the values of their joint CDF, totally $S = 332$ samples are selected. Note that, according to the analysis in Sec. II-B, each of them must provide the feasible solution.

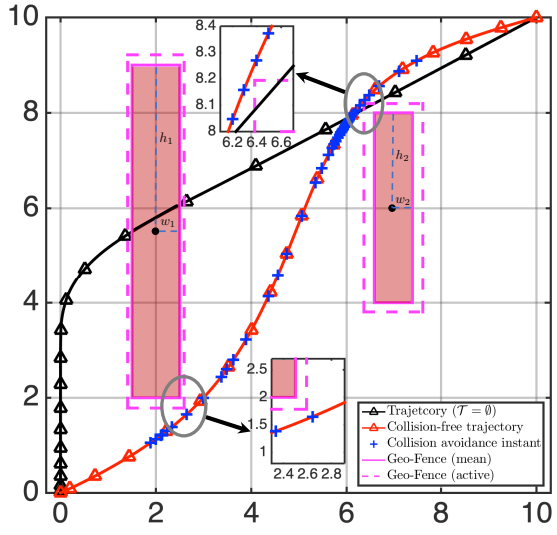
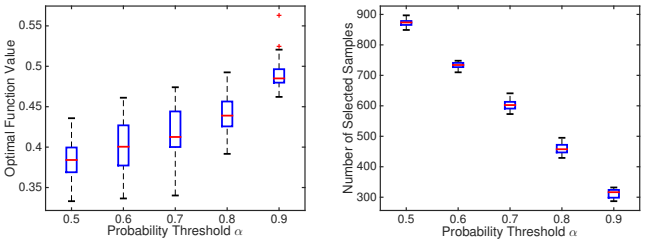


Fig. 4: Collision-Free Trajectory of the UAV

Fig. 4 plots the trajectory planning results simulated by our iterative method. In this figure, the black line shows the obtained trajectory when the collision avoidance step-times are considered as same as the control time-steps, i.e. $\mathcal{T}_s = \{1, 2, \cdots, T_u - 1\}$. In this case, it can be observed that, whereas the UAV's position at each control time-step (marked as triangle) is safe, the trajectory traverses both geo-fences and thus the UAV is actually under risk. Such observation is also consistent with the two examples introduced at the beginning of Sec. III. Next, in order to achieve the totally collision-free trajectory, we iteratively augment the optimization model by adding collision avoidance time-steps. The red line plots the obtained trajectory when such iterative procedure terminates after adding 44 collision avoidance time-steps. Note that the UAV's position at each collision avoidance time-step is marked as the blue plus sign. It can be seen from the zoomed figures that the UAV just passes around the geo-fences without traverses. It is confirmed that the trajectory is collision-free. We also remark that the rectangle in solid lines shows the boundaries of geo-fence in mean values, while the dashed one represents the active scenario with $h_1 = 3.77, w_1 = 0.64$ and $h_2 = 2.24, w_2 = 0.67$. Recall that the detection of unsafe traverse is realized by only considering the active scenarios; see Remark 2, the simulation results are obtained as expected.

At last, to evaluate the effectiveness of the obtained trajectory under chance constraints, we further perform the validation of solution with a sampling data set on the random variables. In detail, we generate the data set by sampling 10^5 *i.i.d.* scenarios on h_i and w_i, following the Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $\mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$. Next, the obtained trajectory is tested by checking if it is collision-free with respect to each sampled scenario. To order to ensure the feasibility under chance constraints with $\alpha = 0.9$, it is expected that the trajectory should be collision-free on more than 90% of sampled scenarios. In fact, we observe that, with 95227 out of 10^5 samples, the trajectory passes the safety check. This confirms the feasibility of solution under the chance constraints.



(a) Optimal function values (b) Number of selected samples

Fig. 5: Trajectory of the UAV with probabilistic geo-fence

B. Performance on the Choice of Probability Threshold

In this subsection, we compare the performance of our method with different choices of the probability threshold α. The simulation setup is the same as in Sec. IV-A, except that the value of α varies from 0.5 to 0.9. In particular, we are concerned with the following performance of the obtained solutions: 1) the optimal objective function value; and 2) the number of selected samples S. Consider that the smaller probability threshold α inherently implies more tolerance of the constraint violation. Thus ideally, with the choice of a smaller α, it should be observed that, 1) the smaller objective function value will be obtained, since the UAV has more freedom to plan its trajectory; and 2) more samples will be selected, since the selection criteria $F\big(\mathbf{r}(n)\big) \geq \alpha$ is less restrictive. In fact, both the implications are confirmed by the obtained simulation results, as shown in Fig. 5. Note that the data reported in Fig. 5 is achieved by running 20 independent simulations. Moreover, it is also observed from Fig. 5(a) that, as the probability threshold α increases, the variance of optimal function values is reduced in general. That is because the obtained solution is less likely to be conservative, when the larger α is specified.

C. Performance on the Choice of Sample Size

In addition, we further evaluate the influence of choices of sample size N on the trajectory planning result. According to the previous analysis in Sec. II-B, when the sample size is getting larger, the solution generated by our sampling based method will be closer to the true optimal one obtained by solving the stochastic problem directly. Next, we expect to verify such implication via the simulation results. Here, we run the previous simulation ($\alpha = 0.8$) with the sample size N being 500, 1000, 2000, and 50000, respectively. As similar to the analysis in Sec. IV-B, we focus on the following performance of the obtained solutions: 1) the optimal objective function value; 2) the number of selected samples S; and additionally, 3) the execution time of our algorithm; 4) the number of collision avoidance time-steps that are added for obtained the collision-free trajectory. Note that the needed number of collision avoidance time-steps is not necessarily proportional to the number of iterations, since multiple time-steps might added at the same iteration of Algorithm 1. Similarly, Table. I provides the results obtained by averaging the data in 20 independent simulation runs. The first number in

TABLE I: Simulation results on different choices of sample size N

N	optimal values	# of selected samples	execution time (sec)	# of added time-steps
500	$\mathbf{0.452} \pm 0.06$	$\mathbf{232} \pm 13$	$\mathbf{21.7} \pm 11.9$	$\mathbf{35.8} \pm 7.9$
1000	$\mathbf{0.439} \pm 0.03$	$\mathbf{460} \pm 19$	$\mathbf{28.6} \pm 18.4$	$\mathbf{39.1} \pm 7.4$
2000	$\mathbf{0.435} \pm 0.05$	$\mathbf{925} \pm 19$	$\mathbf{36.9} \pm 21.2$	$\mathbf{38.7} \pm 7.9$
5000	$\mathbf{0.434} \pm 0.04$	$\mathbf{2328} \pm 29$	$\mathbf{184.8} \pm 143.2$	$\mathbf{33.6} \pm 6.9$

each cell represents the mean of corresponding values, whereas the second one is the standard deviation.

It is observed from the table that, when the sample size N increases, 1) the obtained objective function value is smaller; 2) more sample are selected; 3) more execution time is needed for the algorithm; and 4) however, the needed number of collision avoidance time-steps stays in the same level. We remark that the explanation for 2) is, in fact, trivial, since more samples are drawn before hand. Combining both Table I and Fig. 5(b), it can concluded that the number of selected samples can be affected by both factors; more samples will be selected when more scenarios are sampled and/or the smaller α is specified. Given that the more selected samples are involved in the computing, the algorithm execution time is undoubtedly increased. The observation 1) inherently helps verify the correctness of Proposition 2; the tighter bound for chance constraints is more likely to be sampled when the sample size N increases. Moreover, the data in Table I also shows the trade-off between the efficiency and accuracy for our method. That is, the larger sample size helps to find the more optimal solution, however, the computational complexity is increased on the other hand. Interestingly, it is observed that the optimal objective function value is reduced more drastically when the sample size increases from 500 to 1000 than from 1000 to 5000. In fact, once we increase the sample size from 1000 to 5000, there are 1868 more selected samples (on average) involved in the computing process and 156.2 more seconds (on average) are needed to accomplish the algorithm execution. Nevertheless, the decreasing in optimal function values does not reflect the effort that is made in computing. Such observation implies that an extremely large sample size may not help so much, especially when taking the computational complexity into account.

V. Conclusion

This paper studies the UAV trajectory planning problem with probabilistic geo-fences. A basic CC-MIQP problem is formulated to capture the uncertain boundaries of the geo-fences and also deal with the non-convexity of feasible sets. A novel sampling based solution method is proposed to solve the basic CC-MIQP problem. It is proved that the solution obtained by solving an approximated deterministic optimization has to be feasible to the original CC-MIQP problem. In addition, the approximated solution can be arbitrarily close to the true optimal one as the sample size increases. Moreover, in order to address the potential discretization issue, the iterative CC-MIQP method is further developed, which helps to obtain the totally collision-free trajectory. Numerical simulations finally validate the effectiveness of the proposed method.

VI. Appendix

A. Proof of Proposition 1

Let us introduce an additional set $\tilde{\mathcal{X}}^s$, which has the similar definition as (21), but is only determined by one single selected sample $\tilde{\mathbf{r}}(s)$, i.e.,

$$\tilde{\mathcal{X}}^s := \Bigg\{ \mathbf{x} \;\Big|\; \begin{bmatrix} \cos\left(\theta_i^k\right) \\ \sin\left(\theta_i^k\right) \end{bmatrix}^\top \Big(\mathbf{x} - \mathbf{p}_i\Big) + M\big(1 - z_i^k\big) \geq \tilde{r}_i^k(s), \\ \sum_{k=1}^{K_i} z_i^k \geq 1, \;\; z_i^k \in \{0,1\}, \; \forall i \in \mathcal{I} \Bigg\}. \tag{27}$$

Note that, with a slight abuse of notation, we have absorbed the information of selected sample $\tilde{\mathbf{r}}(s)$ into the index s in the representation of $\tilde{\mathcal{X}}^s$. Recall that the constraint (18) ensures exactly one sample to be activated by enforcing the condition of binary variables, i.e., $\sum_{s=1}^{S} w(s) = 1$. By the definition of $\tilde{\mathcal{X}}(\tilde{\mathcal{R}}^S)$, it can be immediately verified that the set $\tilde{\mathcal{X}}(\tilde{\mathcal{R}}^S)$ is the union of $\tilde{\mathcal{X}}^s$'s, i.e.,

$$\tilde{\mathcal{X}}(\tilde{\mathcal{R}}^S) = \bigcup_{i=1}^{S} \tilde{\mathcal{X}}^s. \tag{28}$$

Consequently, in order to prove $\tilde{\mathcal{X}}(\tilde{\mathcal{R}}^S) \subseteq \mathcal{X}$ when $S \geq 1$, it suffices to show that $\tilde{\mathcal{X}}^s \subseteq \mathcal{X}, \; \forall \tilde{\mathbf{r}}(s) \in \tilde{\mathcal{R}}^S$.

Notice that every selected sample $\tilde{\mathbf{r}}(s)$ has to meet the selection criteria $F\big(\tilde{\mathbf{r}}(s)\big) \geq \alpha$, i.e.,

$$\mathbf{Pr}\Big(r_i^k \leq \tilde{r}_i^k(s), \; \forall k \in \{1, 2, \cdots, K_i\}, i \in \mathcal{I}\Big) \geq \alpha. \tag{29}$$

Therefore, for any point $\mathbf{x} \in \tilde{\mathcal{X}}^s$, combining both (27) and (29) yields,

$$\mathbf{Pr}\left(\begin{array}{c} \begin{bmatrix} \cos\left(\theta_i^k\right) \\ \sin\left(\theta_i^k\right) \end{bmatrix}^\top \Big(\mathbf{x} - \mathbf{p}_i\Big) + M\big(1 - z_i^k\big) \geq r_i^k, \\ \sum_{k=1}^{K_i} z_i^k \geq 1, \;\; z_i^k \in \{0,1\}, \; \forall i \in \mathcal{I} \end{array} \right) \geq \alpha. \tag{30}$$

This implies that $\mathbf{x} \in \mathcal{X}$, and thus the proof is completed.

B. Proof of Proposition 2

We begin the proof by characterizing the arbitrary point in the considered planning space with some features. Let us assign each point $\bar{\mathbf{x}}$ in the planning space with a vector $\bar{\mathbf{r}} = [\bar{r}_1^1, \bar{r}_1^2 \cdots, \bar{r}_1^{K_1}, \cdots, \bar{r}_I^1, \bar{r}_I^2, \cdots, \bar{r}_I^{K_I}] \in \mathbb{R}^K$, where the element $\bar{r}_i^k$ is the distance from $\bar{\mathbf{x}}$ to a line that passes through the center of the geo-fence $\mathbf{p}_i$ and is parallel to boundary k. Let us denote such a line as k'. Fig. 6 shows a simple example of the feature assignment. Note that the entire space is divided into safe and unsafe half spaces by the line k' corresponding to each boundary of the geo-fence. For the k-th boundary related to the i-th geo-fence, if the point $\bar{\mathbf{x}}$ is located in the safe side, then $\bar{r}_i^k$ represents the distance to $\mathbf{p}_i$ along with θ_i^k; otherwise $\bar{r}_i^k = \infty$ if $\bar{\mathbf{x}}$ is located in the unsafe side. On this account, it can be verified that, for any point $\bar{\mathbf{x}} \in \mathcal{X}$, its feature $\bar{\mathbf{r}}$ must satisfy with $F(\bar{\mathbf{r}}) \geq \alpha$, since the chance constraint (12) has to be satisfied. Furthermore, we have shown, in the proof of Proposition 1, that the set $\tilde{\mathcal{X}}(\tilde{\mathcal{R}}^S)$ is the union of sets $\tilde{\mathcal{X}}^s$'s,

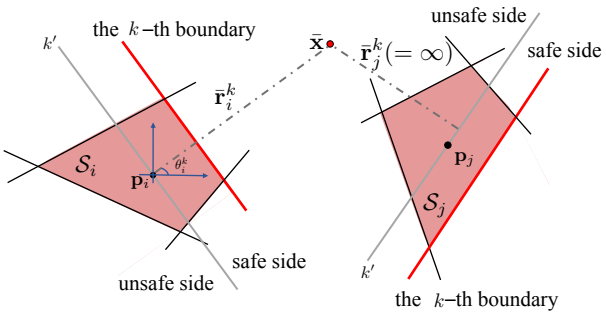


Fig. 6: Characterization of each point $\bar{\mathbf{x}}$ in the planning space

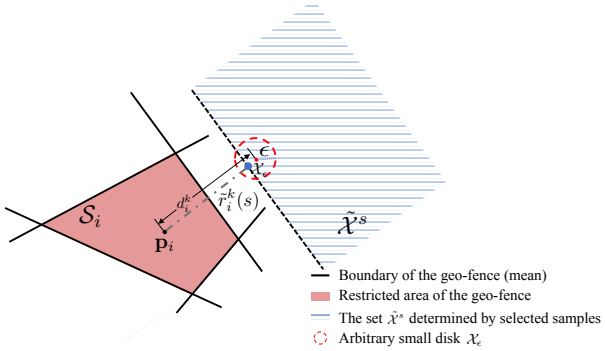


Fig. 7: Demonstration of the connection between $\mathcal{X}^s$ and $\mathcal{X}_\epsilon$

where $\tilde{\mathcal{X}}^s$ is defined as (27). Meanwhile, for any point $\tilde{\mathbf{x}} \in \tilde{\mathcal{X}}^s$, its feature $\tilde{\mathbf{r}}$ must have $\tilde{\mathbf{r}} \geq \tilde{\mathbf{r}}(s)$ element-wise, where $\tilde{\mathbf{r}}(s)$ is the selected sample corresponding to the set $\tilde{\mathcal{X}}^s$.

We next investigate the connection between a subset of $\mathcal{X}$ and the set $\tilde{\mathcal{X}}^s$ determined by the selected sample $\tilde{\mathbf{r}}(s)$. Suppose that the set $\mathcal{X}_\epsilon \subseteq \mathcal{X}$ is an arbitrary small disk which has radius $\epsilon > 0$ and is centering at point $\mathbf{p}_\epsilon \in \mathbb{R}^2$. According to the feature assignment we specified previously, let us denote the feature of point $\mathbf{p}_\epsilon$ as $\mathbf{d} \in \mathbb{R}^K$ where each of the elements d_i^k represents the distance. With the help of definition of the distance d_i^k, it can be verified that, as long as the selected scenario $\tilde{\mathbf{r}}(s)$ has the corresponding distance satisfying $d_i^k - \epsilon < \tilde{r}_i^k(s) < d_i^k$, $\forall k \in K_i, i \in \mathcal{I}$, it implies that $\tilde{\mathcal{X}}^s \cap \mathcal{X}_\epsilon \neq \emptyset$. Fig. 7 provides a simple demonstration of the statement. In fact, since the center point $\mathbf{p}_\epsilon$ satisfies the condition $\mathbf{d} \geq \tilde{\mathbf{r}}(s)$ element-wise, one can immediately prove that $\mathbf{p}_\epsilon$ must be in the set $\tilde{\mathcal{X}}^s$ and thus $\tilde{\mathcal{X}}^s \cap \mathcal{X}_\epsilon \neq \emptyset$. On this basis, if we denote the probability of the event that the selected sample $\tilde{\mathbf{r}}(s)$ satisfies the condition as $\mathbf{Pr}\big(d_i^k - \epsilon < \tilde{r}_i^k(s) < d_i^k\big)$, then it holds that

$$\mathbf{Pr}\big(\tilde{\mathcal{X}}^s \cap \mathcal{X}_\epsilon \neq \emptyset\big) \geq \mathbf{Pr}\big(d_i^k - \epsilon < \tilde{r}_i^k(s) < d_i^k\big). \tag{31}$$

On the other hand, since the selected sample $\tilde{\mathbf{r}}(s)$ is independently drawn based on the uniform distribution $\mathbf{U}(0, \mathbf{r}^{\max})$, then we have

$$\mathbf{Pr}\big(d_i^k - \epsilon < \tilde{r}_i^k(s) < d_i^k\big) \geq \prod_{i=1}^{I}\prod_{k=1}^{K_i} \frac{\epsilon}{(\mathbf{r}^{\max})_i^k} > 0. \tag{32}$$

As a result, it holds that

$$\begin{aligned} &\lim_{S\to\infty} \mathbf{Pr}\left((\cup_{s=1}^{S}\tilde{\mathcal{X}}^s) \cap \mathcal{X}_\epsilon = \emptyset\right) \\ &= \lim_{S\to\infty} \left(1 - \mathbf{Pr}\left(\tilde{\mathcal{X}}^s \cap \mathcal{X}_\epsilon = \emptyset\right)\right)^S \\ &= 0. \end{aligned} \tag{33}$$

Note that the first equation follows from the fact that the samples are independently and identically distributed; and the second equation is due to the inequalities (31) and (32).

Finally, we prove the statement in Proposition 2 by contradiction. Suppose that $\lim_{S\to\infty} \mathbf{Pr}\left(\mathcal{X} \subseteq (\cup_{s=1}^{S}\tilde{\mathcal{X}}^s)\right) = 1$ does not hold. Since the probability $\mathbf{Pr}\left(\mathcal{X} \subseteq (\cup_{s=1}^{S}\tilde{\mathcal{X}}^s)\right)$ is bounded by one and non-decreasing with respect to S, then the limit must converge to a constant $\gamma < 1$, i.e.,

$$\lim_{S\to\infty} \mathbf{Pr}\left(\mathcal{X} \subseteq (\cup_{s=1}^{S}\tilde{\mathcal{X}}^s)\right) = \gamma < 1. \tag{34}$$

Subsequently, $\lim_{S\to\infty} \mathbf{Pr}\left(\mathcal{X} \nsubseteq (\cup_{s=1}^{S}\tilde{\mathcal{X}}^s)\right) = 1 - \gamma > 0$ implies that there must exist a small disk $\mathcal{X}_\epsilon \subseteq \mathcal{X}$ such that $\lim_{S\to\infty} \mathbf{Pr}\left(\mathcal{X}_\epsilon \cap (\cup_{s=1}^{S}\tilde{\mathcal{X}}^s) = \emptyset\right) > 0$. This contradicts the fact (33) and thus the proof is completed.

References

[1] David González, Joshué Pérez, Vicente Milanés, and Fawzi Nashashibi. A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 17(4):1135–1145, 2015.

[2] Wonteak Lim, Seongjin Lee, Myoungho Sunwoo, and Kichun Jo. Hierarchical trajectory planning of an autonomous car based on the integration of a sampling and an optimization method. *IEEE Transactions on Intelligent Transportation Systems*, 19(2):613–626, 2018.

[3] Jianyu Chen, Wei Zhan, and Masayoshi Tomizuka. Autonomous driving motion planning with constrained iterative LQR. *IEEE Transactions on Intelligent Vehicles*, 4(2):244–254, 2019.

[4] Laurène Claussmann, Marc Revilloud, Dominique Gruyer, and Sébastien Glaser. A review of motion planning for highway autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[5] Phillip Chandler, Steven Rasmussen, and Meir Pachter. UAV cooperative path planning. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, page 4370, 2000.

[6] Chuangchuang Sun, Yen-Chen Liu, Ran Dai, and David Grymin. Two approaches for path planning of unmanned aerial vehicles with avoidance zones. *Journal of Guidance, Control, and Dynamics*, 40(8):2076–2083, 2017.

[7] Yucong Lin and Srikanth Saripalli. Sampling-based path planning for UAV collision avoidance. *IEEE Transactions on Intelligent Transportation Systems*, 18(11):3179–3192, 2017.

[8] Matthew G Earl and Raffaello D'andrea. Iterative MILP methods for vehicle-control problems. *IEEE Transactions on Robotics*, 21(6):1158–1167, 2005.

[9] Mina Kamel, Javier Alonso-Mora, Roland Siegwart, and Juan Nieto. Robust collision avoidance for multiple micro aerial vehicles using nonlinear model predictive control. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 236–243. IEEE, 2017.

[10] Yadollah Rasekhipour, Amir Khajepour, Shih-Ken Chen, and Bakhtiar Litkouhi. A potential field-based model predictive path-planning controller for autonomous road vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1255–1267, 2016.

[11] Cedric S Ma and Robert H Miller. MILP optimal path planning for real-time applications. In *2006 American Control Conference*, page 6. IEEE, 2006.

[12] Arthur Richards and Jonathan P How. Aircraft trajectory planning with collision avoidance using mixed integer linear programming. In *Proceedings of the 2002 American Control Conference*, volume 3, pages 1936–1941. IEEE, 2002.

[13] Dimitris Bertsimas and John N Tsitsiklis. *Introduction to Linear Optimization*, volume 6. Athena Scientific Belmont, MA, 1997.

[14] IBM ILOG CPLEX. User's manual for CPLEX. *International Business Machines Corporation*, 46(53):157, 2009.

[15] LLC Gurobi Optimization. Gurobi optimizer reference manual, 2019.

[16] Adam Bry and Nicholas Roy. Rapidly-exploring random belief trees for motion planning under uncertainty. In *2011 IEEE international conference on robotics and automation*, pages 723–730. IEEE, 2011.

[17] Kaito Ariu, Cheng Fang, Marcio Arantes, Claudio Toledo, and Brian Williams. Chance-constrained path planning with continuous time safety guarantees. In *Workshops at the Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[18] Lars Blackmore, Masahiro Ono, Askar Bektassov, and Brian C Williams. A probabilistic particle-control approximation of chance-constrained stochastic predictive control. *IEEE transactions on Robotics*, 26(3):502–517, 2010.

[19] Lars Blackmore and Masahiro Ono. Convex chance constrained predictive control without sampling. In *AIAA Guidance, Navigation, and Control Conference*, page 5876, 2009.

[20] Marcio da Silva Arantes, Claudio Fabiano Motta Toledo, Brian Charles Williams, and Masahiro Ono. Collision-free encoding for chance-constrained nonconvex path planning. *IEEE Transactions on Robotics*, 35(2):433–448, 2019.

[21] Lars Blackmore, Masahiro Ono, and Brian C Williams. Chance-constrained optimal path planning with obstacles. *IEEE Transactions on Robotics*, 27(6):1080–1094, 2011.

[22] Hai Zhu and Javier Alonso-Mora. Chance-constrained collision avoidance for MAVs in dynamic environments. *IEEE Robotics and Automation Letters*, 4(2):776–783, 2019.

[23] Brandon Luders, Mangal Kothari, and Jonathan How. Chance constrained RRT for probabilistic robustness to environmental uncertainty. In *AIAA guidance, navigation, and control conference*, page 8160, 2010.

[24] Brandon D Luders, Ian Sugel, and Jonathan P How. Robust trajectory planning for autonomous parafoils under wind uncertainty. In *AIAA Infotech@ Aerospace (I@ A) Conference*, page 4584, 2013.

[25] Brandon Luders, Aaron Ellertson, Jonathan P How, and Ian Sugel. Wind uncertainty modeling and robust trajectory planning for autonomous parafoils. *Journal of Guidance, Control, and Dynamics*, 39(7):1614–1630, 2016.

[26] Tom Schouwenaars. *Safe trajectory planning of autonomous vehicles*. PhD thesis, Massachusetts Institute of Technology, 2006.

[27] Robin Deits and Russ Tedrake. Efficient mixed-integer planning for UAVs in cluttered environments. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 42–49. IEEE, 2015.

[28] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust Optimization*, volume 28. Princeton University Press, 2009.

[29] Dimitris Bertsimas, David B Brown, and Constantine Caramanis. Theory and applications of robust optimization. *SIAM review*, 53(3):464–501, 2011.

[30] Andras Prékopa. Numerical Solution of Probabilistic Constrained Programming Problems, 1988.

[31] András Prékopa. *Stochastic Programming*, volume 324. Springer Science & Business Media, 2013.

[32] Jun Chen, LiJian Chen, and Dengfeng Sun. Air traffic flow management under uncertainty using chance-constrained optimization. *Transportation Research Part B: Methodological*, 102:124–141, 2017.

[33] Shabbir Ahmed and Alexander Shapiro. Solving chance-constrained stochastic programs via sampling and integer programming. In *State-of-the-Art Decision-Making Tools in the Information-Intensive Age*, pages 261–269. Informs, 2008.

[34] Arkadi Nemirovski and Alexander Shapiro. Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, 17(4):969–996, 2006.

[35] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory*. SIAM, 2014.

[36] Marco C Campi and Simone Garatti. A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. *Journal of Optimization Theory and Applications*, 148(2):257–280, 2011.

[37] Michael B Giles, Tigran Nagapetyan, and Klaus Ritter. Multilevel Monte Carlo approximation of distribution functions and densities. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):267–295, 2015.