Probabilistically safe controllers based on control barrier functions and scenario model predictive control

Allan Andre do Nascimento, Antonis Papachristodoulou, Kostas Margellos

***Abstract*— Control barrier functions (CBFs) in continuous or discrete time, offer a theoretically principled and computationally efficient framework to construct real-time controllers that are safe by design. In the presence of uncertainty/disturbances, however, control barrier function based controllers are often myopic or very conservative; this has motivated research on safety filters based on model predictive control (MPC) and stochastic variations of those. MPC deals with safety constraints in a direct manner, however, it is computationally more demanding depending on the length of the prediction horizon. In this work we propose a safety formulation that involves at every time instance solving a finite horizon optimization problem like MPC, but rather than imposing constraints explicitly at all instances, in view of a closed-loop implementation, we rather enforce probabilistic safety constraints by means of CBFs only at the first step of the horizon. To deal with the probabilistic CBF constraints we use a scenario based methodology, where we require satisfaction of a finite number of CBF constraints corresponding to scenarios. Capitalizing on results on scenario based MPC, we provide distribution-free, *a priori* guarantees on the expected value of the average number of safety violations of the closed-loop MPC control input sequence. Our results are demonstrated on a case study involving an unmanned aerial vehicle position swapping and collision avoidance problem, and are also compared numerically with recent stochastic CBF formulations.**

I. Introduction

Ensuring safe interactions between autonomous systems and humans in shared environments is increasingly relevant. In the control community a typical real-time safe formulation involves defining safety via forward set invariance. This typically entails designing control actions to keep the system within a predefined safe region as time progresses. An emerging method that adopts this concept is Control Barrier Functions [2], which has captured considerable attention and demonstrated interesting results for both continuous [3] and discrete-time systems [1]. CBFs provide a principled framework to guarantee safety, while being suitable across various applications, including real-time deterministic settings for robotic systems [2], guaranteeing safety of learning methods [23], and providing safety assurances under stochastic conditions [8], which is highly important for uncertain environments.

AAdN, AP and KM acknowledge funding support by MathWorks. AP was supported in part by UK's Engineering, Physical Sciences Research Council projects EP/X017982/1 and EP/Y014073/1.

All authors are with the Department of Engineering Science, University of Oxford, Parks Road, Oxford OX13PJ, United Kingdom. {allan.adn, antonis, kostas.margellos}@eng.ox.ac.uk

A typical safety formulation of the problem uses quadratic programming (QP) optimization, aiming to determine least restrictive control actions. In particular, an optimization problem is solved at each time instance, and is subject to CBF constraints, while minimizing the difference of the control effort from some nominal controller [3]. The use of CBFs has also been investigated in uncertain environments under the robust [21] and stochastic lenses [8].

Safety analysis based on CBFs, despite being computationally efficient may be myopic and/or lead to conservative results as far as control invariance calculation is concerned. To alleviate these issues, predictive safety filters have emerged as a promising alternative, often treated as an "add-on" to a pre-existing stabilizing or learned control action [18]. Such approaches inherit the benefits of model predictive control (MPC), and by propagating the system dynamics forward in a receding horizon fashion lead to smoother trajectories, less safety filter interventions and lower control effort [16], [4], [17]. However, this comes at the expense of being increasingly computationally demanding according to the prediction horizon length.

In the presence of uncertainty/disturbance, these benefits of MPC which offers a direct way of enforcing safety and dealing with uncertainty appear more prominent. In light of this, we propose a safety formulation that attempts to combine the relative merits of MPC and CBFs. Our approach involves at every time instance solving a finite horizon optimization problem like MPC, but rather than imposing constraints explicitly at all instances, in view of a closed-loop implementation, we rather enforce probabilistic safety constraints by means of CBFs only at the first step of the horizon. This alleviates the need of pre-stabilizing controllers.

To deal with the probabilistic CBF constraints we use a scenario based methodology, where we require satisfaction of a finite number of CBF constraints corresponding to scenarios. This formulation is in line with data-driven considerations which allow us not to make assumptions on the underlying distribution of the uncertainty or the geometry of the uncertainty set. Our results are distribution-free and capitalize on results on scenario-based MPC [15]. We provide *a priori* guarantees on the expected value of the average number of safety violations of the closed-loop MPC control input sequence. To demonstrate the efficacy of the proposed approach, we evaluate our method on a case study involving a multi-UAV position swapping and collision avoidance problem. We also compare its performance against a related state-of-the-art approach that utilizes super martin-

gale principles [9] to establish a bound on the probability of exiting the safe set.

The remainder of this paper is structured as follows: Section II covers relevant results on stochastic and scenario-based Model Predictive Control. In Section III, we discuss the notion of probabilistic safety we consider and establish probabilistic safety guarantees. Section IV illustrates our findings on a multi-UAV position swapping problem, and compares it with a state-of-the-art stochastic CBF method [9]. Section V provides some concluding remarks and directions for future research.

II. Related results on MPC

We will consider the design of a real-time safe controller for discrete time linear systems. At each time-step a finite horizon stochastic model predictive control (SMPC) problem will be formulated. The first component of the associated optimal input sequence will be applied to the system and the horizon will be rolled as typically performed in predictive control. The first input component of each sequence will be subject to probabilistic safety constraints that are encoded by means of chance constraints. As such, the closed-loop input sequence (the concatenation of the first input components of each finite horizon problem) will be (probabilistically) safe by construction. To deal with the chance constraints present in the SMPC setting we follow a scenario based approach. To formalize this procedure we first revisit some results on stochastic and scenario based model predictive control.

A. Stochastic Model Predictive Control

Consider a discrete time linear system of the form

$$x_{t+1} = A(d_t)x_t + B_u(d_t)u_t + B_d(d_t)d_t, \tag{1}$$

where for each time instance $t \in \mathbb{N}$, $x_t \in \mathbb{R}^n$ denotes its state and $u_t \in \mathbb{R}^p$ the control input. The system is subject to a disturbance $d_t \in \mathbb{R}^d$ that affects the system dynamics in an affine manner. Matrices $A(d_t), B_u(d_t)$ and $B_d(d_t)$ are of appropriate dimension. For each t, we assume that $d_t \in D$, where D is endowed with a σ-algebra $\mathscr{D}$. Let $\mathbb{P}$ denote a probability measure over $\mathscr{D}$ according to which d_t is distributed, and let $\mathbb{E}$ denote the associated expected value operator.

We consider the following SMPC formulation [7], [13]. In particular, for time instance t, SMPC requires formulating the finite horizon optimization problem, with horizon N:

$$\begin{aligned}
\min_{u_{0|t},\dots,u_{N-1|t}} \ & \sum_{k=0}^{N-1} \mathbb{E}[l(x_{k|t}, u_{k|t})] \\
\text{subject to } \ & x_{k+1|t} = A(d_{k|t})x_{k|t} + B_u(d_{k|t})u_{k|t} + B_d d_{k|t}, \\
& x_{0|t} = x_t, \\
& u_{k|t} \in \mathbb{U}, \ \forall k = 0,\dots,N-1, \\
& \mathbb{P}[x_{k+1|t} \notin \mathbb{X}, \ \forall k = 0,\dots,N-1] \leq \varepsilon.
\end{aligned} \tag{2}$$

Note that $x_{k|t}$ and $u_{k|t}$ denote the state and input, respectively, k steps ahead within the prediction horizon at time t.

Our objective is to minimize the expected value of the running cost $l(x_{k|t}, u_{k|t})$ over the prediction horizon $k = 0,\dots,N-1$. The first constraint is the system dynamics constraint, depicting the system evolution throughout the horizon. Since these are linear equality constraints it is to be understood that they are substituted recursively to eliminate all state variables; once this is performed the chance constraint will only involve input variables. We do not perform this substitution to ease notation. The second constraint outlined in (2) embodies the "initialization constraint" which introduces the system feedback at each sampling instance t. In the third constraint of (2), we consider bounded inputs, where $\mathbb{U} \subset \mathbb{R}^p$.

Finally, the fourth constraint of (2) is the chance constraint which introduces a bound ε on the probability that the system state will escape a prespecified state constraint set $\mathbb{X} \subset \mathbb{R}^n$ on any of the k step ahead instances of the prediction horizon. Another perspective on ε is its role as a tuning parameter, offering a trade-off between robustness and performance, meaning, the higher the value of ε, the less likely the system is to stay within the set $\mathbb{X}$ but the higher performance (lower cost) the system will achieve.

B. Scenario based Model Predictive Control

Even if all functions and constraint sets involved are convex, solving (2) can still be challenging, as chance constraints render the problem in general non-convex. To overcome this issue, as well as to avoid making any assumptions on the underlying disturbance's distribution or its domain geometry, we adopt a data driven perspective where we suppose that $D, \mathbb{P}$ are fixed but potentially unknown. The only information we assume to have are scenarios (potentially historical data) for d_t.

To this end, consider the scenario based model predictive control problem analyzed in [15]:

$$\begin{aligned}
\min_{u_{0|t},\dots u_{N-1|t}} \ & \sum_{i=1}^{m} \sum_{k=0}^{N-1} J_k^i(x_{k|t}^i, u_{k|t}) + J_N^i(x_{N|t}^i) \\
\text{subject to } \ & x_{k+1|t}^i = A(d_{k|t}^i)x_{k|t}^i + B_u(d_{k|t}^i)u_{k|t} + B_d(d_{k|t}^i)d_{k|t}^i, \\
& x_{0|t} = x_t, \\
& x_{k+1|t}^i \in \mathbb{X}, \\
& u_{k|t} \in \mathbb{U}, \\
& \forall k = 0,\dots,N-1, \quad \forall i = 1,\dots,m.
\end{aligned} \tag{3}$$

For each t,k, we assume that $d_{k|t}^i$, $i = 1,\dots,m$, are scenarios/samples of $d_{k|t}$. In this regime we are seeking a sequence of inputs that is consistent with the dynamics and satisfies input and state constraints for all scenarios $i = 1,\dots,m$. However, for a given input, for each scenario a different state trajectory is generated. To reflect this, we introduce superscript i in $x_{k|t}^i$ and $d_{k|t}^i$.

In the sequel we capitalize on the scenario based reformulation of SMPC and show how safe-by-design controllers can be generated within this paradigm. In particular, rather than enforcing constraints that encode a safe part of the state space explicitly within $\mathbb{X}$, we opt for a computationally more efficient solution, where we enforce safety by means of control

barrier functions [2], [19]. Moreover, scenario based SMPC offers a tractable albeit approximate reformulation of SMPC. As such, we aim at accompanying the constructed closed-loop input sequence with (probabilistic) guarantees regarding the satisfaction of state constraints; as such constraints in our setting are related to safety we offer guarantees on safe performance for the closed loop input.

III. Problem statement and main results

A. Scenario based safety

1) Safety constraints: We consider state constraints encoding safety, as this is represented via control barrier functions. To this end, consider a set S, to represent the zero-superlevel set of a function [1] denoted as $h : \mathscr{P} \subset \mathbb{R}^n \to \mathbb{R}$:

$$S = \{x_t \in \mathscr{P} \subset \mathbb{R}^n : h(x_t) \geq 0\}.$$

Guaranteeing system trajectories to be within S, implies the system is safe. For a given disturbance, this is equivalent to enforcing invariance of S along the trajectories of (1). An explicit condition for system safety under the scenario approach can be derived based on the original discrete time barrier function conditions presented in [1]. To this end, we consider that $\mathbb{X}$ is defined by the following constraints:

$$\begin{aligned} &(i)\ h(x_{0|t}) \geq 0, \\ &(ii)\ \exists\, u_{k|t} \text{ such that } \forall t \in \mathbb{N} \cup \{0\}, \\ &\quad h(x^i_{k+1|t}) - h(x^i_{k|t}) \geq -\gamma h(x^i_{k|t}),\ \forall i = 1,\dots,m. \end{aligned} \tag{4}$$

In this context, we assume that γ lies within the interval (0, 1). However, a more general approach outlined in [22], could be adopted, whereby a class κ functional is considered instead, ensuring $0 < \gamma(h(x^i_{k|t})) \leq h(x^i_{k|t})$.

Notice that item (i) in (4) does not involve the superscript i since, at the current time t, the system remains unaffected by $d^i_{0|t}$, $i = 1,\dots,m$. Moreover, to satisfy the condition in item (ii) of (4), there must exist at least one $u_{k|t}$ such that the inequality is satisfied for all scenarios indexed by $i = 1,\dots,m$ at a given look ahead instance k to be considered. It is worth highlighting that while these conditions mirror the original definition in [1], the requirement for an input sequence to satisfy these across all scenarios significantly restrict the solution space. This trade-off is an unavoidable "price of robustness" for a distribution-free setting. Finally, when both conditions in (4) are satisfied, it is stated that $h : \mathscr{P} \to \mathbb{R}$ serves as a discrete-time exponential control barrier function for the generated scenarios.

2) Cost function: For each scenario $i = 1,\dots,m$, the corresponding term in the objective function of (3) is computed as

$$\begin{aligned} J^i_k(x^i_{k|t}, u_{k|t}) &= (x^i_{k|t})^T Q x^i_{k|t} + (u_{k|t})^T R u_{k|t}, \\ J^i_k(x^i_{N|t}) &= \eta (x^i_{N|t})^T Q_N x^i_{N|t} \end{aligned} \tag{5}$$

where $Q \succeq 0$ and $R \succ 0$ are matrices of appropriate dimension used to penalize the state and the control input along the prediction horizon, respectively. Matrix $Q_N \succeq 0$ is used to encode a terminal state penalty, which can be further scaled by an appropriately chosen high weight coefficient $\eta \geq 0$, alleviating the need of a terminal set [11]. For multi-agent systems, one could also include an aggregative or "team goal" in the objective as considered in [10]. This extra term can serve different purposes. For instance, it could incorporate into the optimization process a metric assessing the fleet's ability to follow a shared target point.

B. Closed loop safe set violation guarantees

Following the developments of [15], we impose the following assumptions (3).

Assumption 1:
1) *Uncertainty:* Consider the product probability space $(\Delta^m, \mathbb{P}^m)$. Assume that for each t, the scenarios $[d^i_{0|t}, \dots, d^i_{N-1|t}]$, $i = 1,\dots,m$ are independent and identically distributed elements of the product probability space.
2) *Problem structure:* The objective function and state constraint set in (3) are convex. The input constraint set is convex and bounded, while full state measurement is assumed.
3) *Feasibility:* At each instance t, it is assumed that problem (3) admits almost surely (with respect to the choice of the scenarios) a feasible solution.

The first part of this assumption requires all scenarios generated to be independent. However, along a prediction horizon of a specific scenario i, temporal correlation is still allowed, e.g., for each fixed value of $i = 1,\dots,m$, the elements of $[d^i_{0|t}, \dots, d^i_{N-1|t}]$ can be correlated. The second part of this assumption is rather mild; it is worth noting that this approach does not enforce any specific structure regarding the system's dependency on uncertainty (beyond the generic system dynamics' structure) or the distribution support of disturbances.

The third part of this assumption refers to a requirement of our problem to remain (recursively) feasible, for almost all realizations of the scenarios. In practice, this is difficult to be satisfied; however, this requirement can be relaxed. In such cases, with the confidence that we will establish in the sequel, our probabilistic safety guarantees will still hold whenever the problem is found to be feasible [5], [12].

Let $\omega^i_t = \{d^i_{k|t}\}_{k=0}^{N-1}$ and $\omega_t = \{\omega^i_t\}_{i=1}^m$ be the collection of all finite-horizon samples. Given a current state $x_t(\omega_t)$ (notice the dependence on all scenarios up to that time), denote the next-step safety violation probability as $\mathbb{P}[x_{t+1} \notin \mathbb{X} | x_t(\omega_t)]$. Moreover, define ρ as the so called support rank of the constraint $\mathbb{X}$ [15], given by $\rho = p - \dim\mathscr{L}$, where $\dim\mathscr{L}$ represents the dimension of the largest unconstrained subspace within the decision variable space, which has dimension p (equal to the dimension of the input vector). In particular, if the safety constraints are encoded, or can be approximated by means of linear inequalities (as in the numerical examples considered in the sequel), then we can represent them as $A_t u_{0|t} \leq b(d_t)$, where A_t is solely dependent on t, and focuses exclusively on the "next-step" safety. We can determine $\dim\mathscr{L}$ and subsequently ρ based on the structure of A_t. If $A_t \in \mathbb{R}^{z \times m}$ has a rank r for all d_t and all t, then $\dim\mathscr{L} = p - r$. For this specific case, note that $\rho \leq p$ as $0 \leq \dim\mathscr{L} \leq p$.

Following [15, Lemma 13], we can then bound this probability as summarized in the following lemma.

Lemma 1: Consider Assumption 1. Fix $\varepsilon \in (0,1)$, and a confidence level $\beta \in (0,1)$. Select the number of scenarios m such that

$$\min\left\{1, \sum_{j=0}^{\rho-1} \binom{m}{j} \varepsilon^j (1-\varepsilon)^{m-j}\right\} \leq \beta,$$

where ρ denotes the support rank of the safety constraints, as defined above. We then have that for each t,

$$\mathbb{P}^{mN}[\omega_t : \ \mathbb{P}[x_{t+1} \notin \mathbb{X} | x_t(\omega_t)] \leq \varepsilon] \geq 1-\beta. \tag{6}$$

In words, for a properly chosen number of scenarios m, we guarantee that with confidence at least equal to $1-\beta$, the probability of being unsafe at the next step is at most equal to a prespecified level ε.

Lemma 1 provides a bound on the next-step probability of being safe. Despite this being interesting per se, when operating in closed-loop, we would rather have guarantees on the safety violation properties of the closed-loop input sequence. To achieve this, we follow the rationale of [15, Theorem 16] and consider the expected value of the average frequency of closed-loop safety constraints violation. Let $T \in \mathbb{N}$ denote the number of MPC horizon rolls. We then have that the average number of safety violations for the closed-loop input sequence is given by

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbf{1}_{\mathbb{X}^c}(x_{t+1}),$$

where $\mathbf{1}_{\mathbb{X}^c}$ is an indicator function, which takes the value 1 if x_{t+1} belongs to the complement of the safe set, namely, $\mathbb{X}^c$, and 0 otherwise. We then have the following result.

Theorem 1: Consider Assumption 1 and fix $\varepsilon \in (0,1)$. Select the number of scenarios $m \geq \frac{\rho}{\varepsilon} - 1$, where ρ denotes the support rank of the safety constraints. We then have that

$$\mathbb{E}^{T(mN+1)}\left[\frac{1}{T}\sum_{t=0}^{T-1} \mathbf{1}_{\mathbb{X}^c}(x_{t+1})\right] \leq \varepsilon. \tag{7}$$

We believe that Theorem 1 offers the first result that characterizes probabilistically, the safety violation properties for closed-loop input sequences generated by a combination of MPC and control barrier functions, while not demanding any knowledge of the distribution of the uncertainty.

We note that both Lemma 1 and Theorem 1 could be extended to allow for an *a priori* fixed number of samples to be removed in an *a posteriori* fashion. This allows trading feasibility to optimality and a better cost. We do not pursue this direction here but refer to such schemes [14], [6].

The overall procedure that involves combining CBFs and scenario based MPC is summarized in Algorithm 1. Step 3 updates the safety constraints based on the most recent x_t information from step 2. Here we consider that the support rank value of the chosen function $h(x_t)$ is unaffected by scenario i to be generated in subsequent steps, as assumed in [15]. Steps 4, 5 and 6 involve the calculation and generation of the minimum number of scenarios to be generated for the chosen ε and ρ. In the specific case where the safety constraints in Step 3 are fixed and ρ is constant for all t, Algorithm 1 can be solved faster by moving steps 3-6 out of the loop.

Algorithm 1 Probably Safe Scenario MPC - (PSS-MPC)

1: **for** $t = 0, 1, \ldots$ **do**
2: Measure state x_t.
3: Calculate safety constraints as in (4).
4: Fix the values of ε and calculate the support rank ρ.
5: Use Theorem 1 to obtain the number of scenarios m.
6: Generate (or collect) m scenarios.
7: Solve problem (3), using (4) and (5).
8: Apply $u_t = u_{0|t}$ in (1).
9: **end for**

In general, one must fix two among ε, ρ and m to obtain the third one. This means that even if a limited number of scenarios m is available, one could still make use of Theorem 1 to find a lower bound ε for the given number m of samples available and the support rank ρ of the constraints considered.

IV. Numerical examples

A. Position swapping and collision avoidance of UAVs

In [10] a distributed algorithm for a multi-agent UAV setting was proposed, and resulted in a control law that was collision free and allowed for position swapping. Here we are not concerned with a distributed implementation, even though the scenario program proposed is directly amenable to distributed schemes. We rather revisit the problem to account for the presence of disturbance affecting the UAVs' dynamics.

To this end, consider 4 UAVs, that execute planar trajectories. For each $j = 1, \ldots, 4$, denote by $p_{0_j} = [p_{x_j}, p_{y_j}]$ their initial position, with p_{x_j} and p_{y_j} representing their horizontal and vertical coordinate, respectively. With reference to Figure 1, UAVs are initially positioned at $p_{0_1} = [0,\ 1]^T$, $p_{0_2} = [0,\ -1]^T$, $p_{0_3} = [1,\ 0]^T$, and $p_{0_4} = [-1,\ 0]^T$. Their target positions are set as: $p_{d_1} = p_{0_2}$, $p_{d_2} = p_{0_1}$, $p_{d_3} = p_{0_4}$, and $p_{d_4} = p_{0_3}$.

Dynamics: All UAVs are represented as discrete-time double integrators; for each $j = 1, \ldots, 4$, each UAV's state is given by $x_j = [p_{x_j}, p_{y_j}, v_{x_j}, v_{y_j}]^T$, where v_{x_j} and v_{y_j} denote the horizontal and vertical velocity components, respectively. These dynamics are affected by an additive disturbance so that with reference to (1), $B_d^j = 5 \times 10^{-3} \times (-1)^{j+1}[I_{2\times 2}; 0_{2\times 2}]$, where $I_{2\times 2}$ and $0_{2\times 2}$ denote the 2×2 identity and zero matrices, respectively. Moreover, the superscript j is introduced to indicate the matrices/vectors associated to UAV j. We assume that all disturbance scenarios generated are independently extracted from a normal distribution $\mathcal{N}(0,1)$. In this particular implementation the samples are also independent along the horizon, being drawn from the same type of disturbance.

Objective function: We assume that all matrices appearing in the objective function are the same for each agent. Namely,

for all $j = 1,\ldots,4$, $Q^j = 5 \times I_{4\times 4}$, $R^j = 2 \times I_{2\times 2}$ and Q_N^j is set to be equal to the solution of the discrete time algebraic Riccati equation. Moreover, we set $\eta = 0.1$. The overall system matrices Q, R and Q_N in (5) are then constructed as the diagonal concatenation of Q^j, R^j and Q_N^j, respectively.

Constraints: Safety in this setting is encoded by avoiding all pairwise collisions. For each pair of UAVs this can be encoded by enforcing the two conditions in (4). A discrete time control barrier function to achieve this was detailed in [10]; in particular, with a slight abuse of notation, the control barrier function $h_{i,j}$ between agents i and j, was considered to be $h_{i,j} = \frac{|p_{x_i} - p_{x_j}|}{r_1} + \frac{|p_{y_i} - p_{y_j}|}{r_2} - 1$, where r_1 and r_2 represent collision radii, set to $0.25m$ and $0.5m$, respectively. Moreover, we set $\gamma = 0.2$. This control barrier function was linearized around the states x_i, x_j at time t, giving rise to constraint $A_t u_{0|t} \leq b(d_t)$.

The input in each UAV's dynamics is acceleration. This is subject to constraints of the form $-a_{\max} \times 1_{8\times 1} \leq u_{k|t} \leq a_{\max} \times 1_{8\times 1}$, where $1_{8\times 1}$ is a 8×1 column vector of ones, $u_{k|t}^j \in \mathbb{R}^{2\times 1}$ is the k steps ahead input calculated for UAV j at time t, and $u_{k|t} = [(u_{k|t}^1)^T,\ldots,(u_{k|t}^4)^T]^T$. The input limits were set to $a_{max} = 4m/s^2$.

Scenario based MPC and probabilistic guarantees: To solve the proposed scenario based MPC we used a prediction horizon $N = 3$. The matrix A_t that appears in the linearized safe constraints has a rank of 6 (full row rank) as long as the system is safe. As such, the support rank associated with the safety constraints can be calculated as $\rho = 6$. Setting $\varepsilon = 0.05$, Theorem 1 implies that we need to consider $m = 119$ scenarios to meet the desired level ε.

The resulting trajectories of the scenario based MPC problem are shown in Figure 1. Notice that these trajectories avoid collision and achieve position swapping, while they are robust with respect to the considered scenarios. At the same time, by Theorem 1, the expected value of the average number of safety violations is at most 0.05. For one of the considered UAVs, Figure 2 contrasts the deterministic trajectory with the designed (probabilistically) robust one. The associated trajectories for the other UAVs are qualitatively similar.

We also investigate the validity of Theorem 1 empirically. For this purpose we consider $\gamma = 0.9$ and $\varepsilon = 0.1$. To meet this level ε, Theorem 1 requires generating at least $m = 59$ scenarios. We consider $T = 90$ time steps. To validate ε empirically we constructed the empirical expectation $\widehat{\mathbb{E}}$ for the left-hand side of (7). To achieve this, we conducted the following procedure: At each time step, we generated $m = 59$ scenarios and solved the proposed scenario program, yielding an input $u_{0|t}^j$, which we applied to UAV j, $j = 1,\ldots,4$. We ran this for $T = 90$ time steps. We then calculated the empirical average of violating the safe constraint within the considered T time steps by counting the number of timesteps for which safety was violated and dividing it with the total number of time steps T. In this multi-agent setting any number of collisions at a given time is considered as one unsafe instance. This guarantees we do not double count unsafe occurrences.

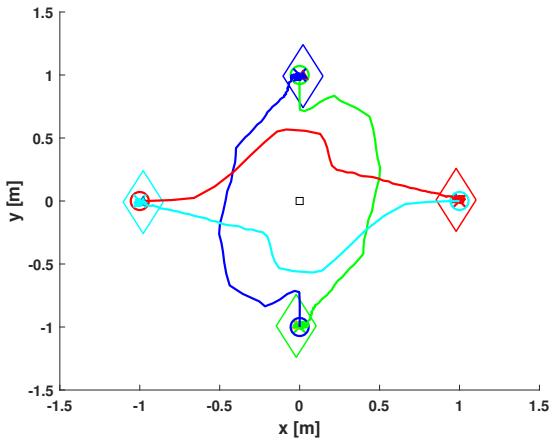

Fig. 1: UAV trajectories during the position swap maneuver. Polygons denote unsafe regions, triangles are UAVs, circles denote the initial positions, crosses the target ones, and lines denote the UAV trajectories. Trajectories depict the closed-loop solution of the formulated scenario based MPC problem.

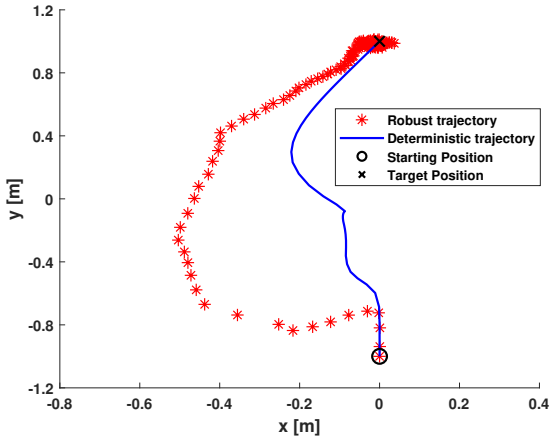

Fig. 2: Comparison between the deterministic (blue) and robust (solution of the proposed scenario based MPC) trajectories (red) for one of the UAVs of Figure 1.

We repeat this process for $M = 100$ independent runs, and for each run we recorded this empirical frequency, namely, f_ℓ, $\ell = 1,\ldots,M$. We construct the empirical expectation $\widehat{\mathbb{E}}$ as

$$\widehat{\mathbb{E}} = \frac{1}{M}\sum_{\ell=1}^{M} f_\ell.$$

Figure 3 illustrates the empirical distribution of the average safety violations for $T = 90$ time steps. The dashed red line corresponds to the theoretical bound on the expected value of the average safety violations ε (see Theorem 1), while the dashed green line corresponds to its empirical expectation $\widehat{\mathbb{E}}$. Incorporating a sampling and discarding mechanism [14], [6] would allow reducing the conservatism and shifting the empirical expectation towards the theoretical one.

The choice of $\gamma = 0.9$ was made specifically to test the

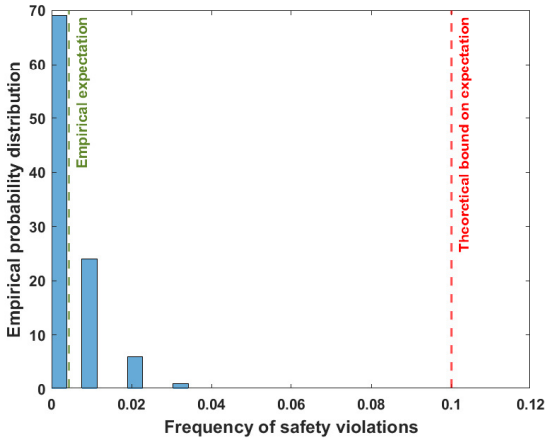

Fig. 3: Empirical distribution (calculated based on 100 independent runs) of average safety violations for $T = 90$ time steps.

bound of Theorem 1 under a smaller extra level of conservativeness. In other words, the "decay rate" γ of the control barrier function introduces by itself another layer of conservativism as the smaller the value of γ, the more cautious is the system against collisions. Therefore, the smaller the value of γ, the smaller the chance the system will actually violate the safety constraints and subsequently the lower will the empirical expectation become. This observation suggested that we needed to use a high value of gamma to reduce unnecessary conservativeness and check the bound provided by (7).

B. Comparative study

We compare our probabilistic safety guarantees with the approach proposed in [9] by contrasting them on a numerical example considered therein. To this end, consider an one-dimension linear system: $x_{t+1} = x_t + 2 + u_t + \sigma d_t$, where $d_t \sim \mathcal{N}(0,1)$, and define the safe set as $S_v = \{x | h(x) \geq -v\}$, with $h(x) = 10 - x^2$ and $v = 0$.

The methodology proposed in [9] follows a CBF tightening procedure and martingale arguments, to provide guarantees on the probability that the resulting controller exits the safe set at some time. Theoretically, our probabilistic results are of different nature as we provide bounds on the expected value for the average constraint violations of the closed-loop system. However, the bound of our main theorem relies on Lemma 1, which can be thought of as a bound on the probability of exiting the safe set in one step. As such, we use the one-step exit (of the safe set) probability as the common ground to contrast these approaches.

More specifically, for the particular example [9] considers the following CBF problem

$$\begin{aligned} u_t = \ & \text{argmin} \|u\|^2 \\ & \text{subject to} \ \ \mathbb{E}[h(x_{t+1})] \geq (1-\sigma^2) h(x_t). \end{aligned} \tag{8}$$

As the expectation operator cannot be parsed inside h (apart from specific cases), the following tightening version is considered

$$\begin{aligned} u_t = \ & \text{argmin} \|u\|^2 \\ & \text{s.t.} \ h(\mathbb{E}[x_{t+1}]) - \sigma^2 \geq (1-\sigma^2) h(x_t). \end{aligned} \tag{9}$$

Recall that σ is the coefficient of d_t in the dynamics and effectively plays the role of standard deviation of the term σd_t. It is shown that the solution of the tightened problem is also a feasible solution to (8). The one-step exit probability bound proposed in [9], when adapted to this problem is:

$$P_u \leq 1 - \frac{h(x_0)}{M}(1-\sigma^2), \tag{10}$$

where M represents the maximum value of $h(x)$ and $h(x_0)$ is its initial value. It is worth noting that the bound becomes increasingly close to 1 as the system approaches the safe set boundary.

In our approach, Lemma 1 suggests that the one-step exit probability is given by ε. We can thus fix any ε as long as the number of samples m and the confidence level β are chose so that $\min\left\{1, \sum_{j=0}^{\rho-1} \binom{m}{j} \varepsilon^j (1-\varepsilon)^{m-j}\right\} \leq \beta$.

We set $\sigma = 0.9$ and $x_0 = 3.1$. We also choose $\varepsilon = 0.1$ which, for a confidence level $\beta = 10^{-4}$, requires a minimum of $m = 88$ samples. For these numerical values, the theoretical upper-bound on the one-step exit probability for the methodology in [9] is $P_u \leq 0.99$ (follows from (10)) while for our approach this is set to $\varepsilon = 0.1$. Figure 4 provides a statistical analysis for this comparison. We have simulated the system under consideration using the controller emanating from (9) and using the controller generated by our scheme, setting $T = 1$ time step. For each controller, we counted the empirical frequency of unsafe occurrences out of these 1000 time steps. We then repeated this process for $M = 100$ independent runs. The distribution of the recorded empirical frequences across these runs is illustrated by means of the two boxplots in Figure 4. It is to be noted that the probabilistic bounds observed for the methodology of [9] are concentrated to significantly higher values, thus rendering the guarantees offered by our proposed approach sharper.

While empirical calculations do adhere to the theoretical bound, it has come to our attention that controller (9) sometimes encounters difficulties in leveraging its unbounded input capacity to ensure system safety. In contrast, the scenario-based controller maintains the system within the safe set for over 90% of the time with arbitrary confidence, irrespective of its proximity to the boundary. This suggests it presents an efficient utilization of control capacity, despite the smaller size of the feasible set arising from the numerous constraints generated by the scenarios and its increased computation time.

V. Concluding remarks and future work

In this study, we introduce a sample-based approach for safe model predictive control. This method works as an intermediary solution between stochastic QP-CBF and safety filters, presenting the advantages of eliminating pointwise safety guarantees of myopic controllers while not necessarily introducing many constraints on the problem formulation.

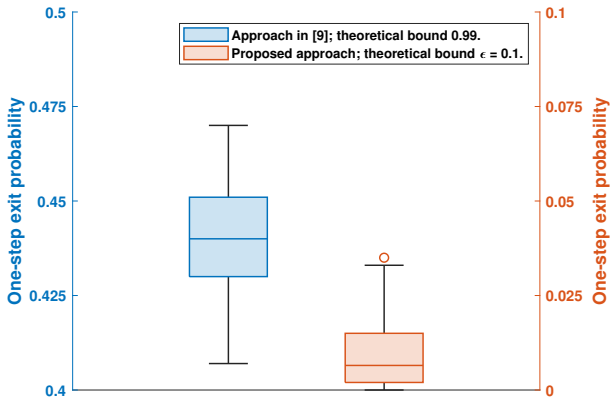

Fig. 4: Comparison of one-step exit probability: [9] and ours (Lemma 1). The left boxplot, linked to the left verical axis represents the approach in [9]. The Right boxplot, linked to the right vertical axis corresponds to our proposed approach.

Our approach offers upper bounds on the expected frequency of chance constraint violations for systems in closed-loop, which is achieved by integrating a scenario-based MPC framework with control barrier functions. We go on to evaluate this approach through two numerical studies. First, we simulate a UAV swapping task under external disturbances. Second, we compare our technique with a state-of-the-art method that offers a future step exit probability upper bound. These evaluations focus on the next-step exit from the safe set. Different future research avenues are promising. Extending the "CBF horizon" for collision avoidance in non-convex safe sets. Another one is to consider the recursive feasibility of this method under mission-wide probabilistic constraints as in [20]. Finally, the approaches compared in Section IV have similarities prompting a deeper comparison.

REFERENCES

[1] Ayush Agrawal and Koushil Sreenath. Discrete Control Barrier Functions for safety-critical control of discrete systems with application to bipedal robot navigation. In *Robotics: Science and Systems*, volume 13. Cambridge, MA, USA, 2017.

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

[3] Aaron D Ames, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *53rd IEEE Conference on Decision and Control*, pages 6271–6278. IEEE, 2014.

[4] Joseph Breeden and Dimitra Panagou. Predictive control barrier functions for online safety critical control. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 924–931. IEEE, 2022.

[5] Giuseppe Carlo Calafiore and Marco C Campi. The scenario approach to robust control design. *IEEE Transactions on automatic control*, 51(5):742–753, 2006.

[6] 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.

[7] Mark Cannon, Basil Kouvaritakis, and Xingjian Wu. Probabilistic constrained mpc for multiplicative and additive stochastic uncertainty. *IEEE Transactions on Automatic Control*, 54(7):1626–1632, 2009.

[8] Andrew Clark. Control barrier functions for stochastic systems. *Automatica*, 130:109688, 2021.

[9] Ryan K Cosner, Preston Culbertson, Andrew J Taylor, and Aaron D Ames. Robust safety under stochastic uncertainty with discrete-time control barrier functions. *arXiv preprint arXiv:2302.07469*, 2023.

[10] Allan Andre do Nascimento, Antonis Papachristodoulou, and Kostas Margellos. A game theoretic approach for safe and distributed control of unmanned aerial vehicles. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 1070–1075. IEEE, 2023.

[11] Daniel Limón, Teodoro Alamo, Francisco Salas, and Eduardo F Camacho. On the stability of constrained MPC without terminal constraint. *IEEE transactions on automatic control*, 51(5):832–836, 2006.

[12] Kostas Margellos, Maria Prandini, and John Lygeros. On the connection between compression learning and scenario based single-stage and cascading optimization problems. *IEEE Transactions on Automatic Control*, 60(10):2716–2721, 2015.

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

[14] Licio Romao, Kostas Margellos, and Antonis Papachristodoulou. Tight generalization guarantees for the sampling and discarding approach to scenario optimization. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 2228–2233. IEEE, 2020.

[15] Georg Schildbach, Lorenzo Fagiano, Christoph Frei, and Manfred Morari. The scenario approach for stochastic model predictive control with bounds on closed-loop constraint violations. *Automatica*, 50(12):3009–3018, 2014.

[16] Ben Tearle, Kim P Wabersich, Andrea Carron, and Melanie N Zeilinger. A predictive safety filter for learning-based racing control. *IEEE Robotics and Automation Letters*, 6(4):7635–7642, 2021.

[17] Kim P Wabersich, Lukas Hewing, Andrea Carron, and Melanie N Zeilinger. Probabilistic model predictive safety certification for learning-based control. *IEEE Transactions on Automatic Control*, 67(1):176–188, 2021.

[18] Kim Peter Wabersich and Melanie N Zeilinger. A predictive safety filter for learning-based control of constrained nonlinear dynamical systems. *Automatica*, 129:109597, 2021.

[19] Han Wang, Kostas Margellos, and Antonis Papachristodoulou. Assessing safety for control systems using sum-of-squares programming. In *Polynomial Optimization, Moments, and Applications*, pages 207–234. Springer, 2023.

[20] Kai Wang and Sébastien Gros. Recursive feasibility of stochastic model predictive control with mission-wide probabilistic constraints. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 2312–2317. IEEE, 2021.

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

[22] Jun Zeng, Bike Zhang, and Koushil Sreenath. Safety-critical model predictive control with discrete-time Control Barrier Function. In *2021 American Control Conference (ACC)*, pages 3882–3889. IEEE, 2021.

[23] Liqun Zhao, Konstantinos Gatsis, and Antonis Papachristodoulou. Stable and safe reinforcement learning via a barrier-lyapunov actor-critic approach. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 1320–1325. IEEE, 2023.