Probabilistic obstacle avoidance and object following: An overlap of Gaussians approach

Dhaivat Bhatt[1*] Akash Garg[2*] Bharath Gopalakrishnan[1] K. Madhava Krishna[1]

***Abstract*— Autonomous navigation and obstacle avoidance are core capabilities that enable robots to execute tasks in the real world. We propose a new approach to collision avoidance that accounts for uncertainty in the states of the agent and the obstacles. We first demonstrate that measures of entropy—used in current approaches for uncertainty-aware obstacle avoidance—are an inappropriate design choice. We then propose an algorithm that solves an *optimal* control sequence with a guaranteed risk bound, using a measure of overlap between the two distributions that represent the state of the robot and the obstacle, respectively. Furthermore, we provide closed form expressions that can characterize the overlap as a function of the control input. The proposed approach enables model-predictive control framework to generate bounded-confidence control commands. An extensive set of simulations have been conducted in various constrained environments in order to demonstrate the efficacy of the proposed approach over the prior art. We demonstrate the usefulness of the proposed scheme under tight spaces where computing risk-sensitive control maneuvers is vital. We also show how this framework generalizes to other problems, such as object-following.**

I. INTRODUCTION

Autonomous robots are required to navigate between a set of defined locations while avoiding any other agent present on its predicted path. However, uncertainty arising in the motion of a robot coupled with uncertainty in observation as well as in environment makes path planning an interesting job. This problem is necessary to address when autonomous robots and humans exist in a shared environment like semi-automated warehouses or mining sites. Strong requirement of uncertainty aware motion planning necessitates need of new algorithms that can accurately model various uncertainties arising in stochastic world of humans. Towards this, we make an effort to develop a theoretical approach that could take quantitative measure of uncertainty of robot and other objects into consideration before making a planning decision. The environment can be populated with static and dynamic objects as well as other robots. A great amount of work done in the past deals with motion planning of the robot from given start pose to goal pose in the presence of obstacles. Many of the deterministic obstacle avoidance algorithms studied in literature, when used for a robot present in an uncertain environment can lead to substantial degradation in the desired result and can even make the source robot to collide into the obstacle, in the worst case. Chance constraint is one of the efficient way of employing the probability of constraint violation which is below a user specified bound. These constraints are combined with other constraints accounting for the kinematic and dynamic feasibility in form of actuation constraints of the robot. Our approach generates smooth trajectories which can be employed on real robots. This paper has several novel findings and contributes in following ways,

- This is first such formulation, conditioned on agent and obstacle uncertainty into an MPC framework through theory of overlapping Gaussians.

[1]Affiliated with KCIS, Robotics Research Center, IIIT Hyderabad
[2]Affiliated with Delhi Technological University
*Equal contribution
This work is supported by grants made available from Rockwell Collins, India development center.

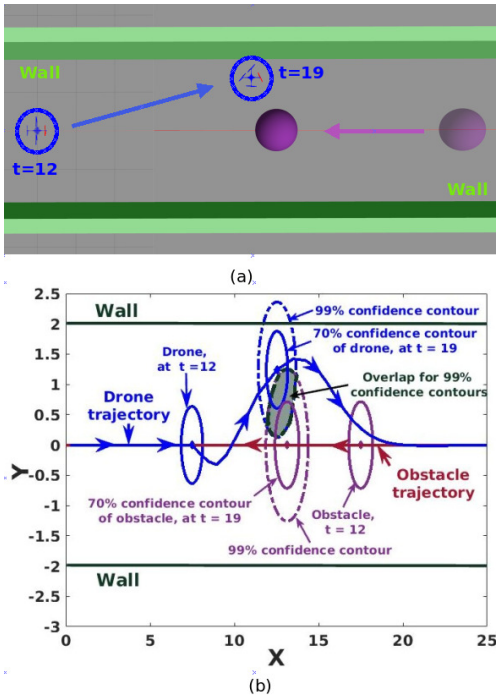


Fig. 1. We show result of our probabilistic obstacle avoidance algorithm in constrained corridor when an obstacle is approaching in antipodal configuration. Figure 1(a) shows gazebo snapshot of agent positions for two different time instances. Our objective is to ensure that 70% confidence contour of agent's Gaussian is avoiding atleast 70% confidence contour of obstacle's Gaussian distribution at any time instance during the entire trajectory. In figure 1(b), we show how the control maneuvers found through our algorithm respects user-specified safety bound. For example, at time $t = 19$, we can see clear overlap(shaded area) between 99% confidence contours of the agent and obstacle. However, confidence contours corresponding to 70% are not being penetrated which can be seen from figure 1(b), thus, we can clearly see how a user-specified lower bound is being respected.

- We demonstrate why our modeling is more consistent and appropriate compared to the widely used Bhattacharya distance, that measures the similarity of two probability distributions.
- We formulate an optimization framework that solves for chance constraints using the theory of overlapping Gaussians.
- We show effective results in various simulation settings that showcase versatility of the method. Specifically we show where the distributions are non isotropic, which is closer to real setting.

Above novelties are arising out of two important problems we are solving, First one is probabilistic collision avoidance and second is probabilistic object following. We formally define them below,

- **Probabilistic collision avoidance:**

  Having knowledge about agent's initial state, agent's belief propagation and the trajectory of obstacles' state, the objective is to find an optimal sequence of control inputs. Such a sequence ensures that overlapping area between

any obstacle and agent's probability distributions at any time instance is less than prescribed maximum allowed overlap (Υ_{max}).

- **Probabilistic object following:**

  Having knowledge about agent's initial state, agent's belief propagation and the trajectory of object's state, the objective is to find an optimal sequence of control inputs. Such a sequence ensures that overlapping area between agent and object's probability distributions at any time instance is between prescribed range of overlap. This range is characterized by lower and upper bounds of overlap $\Upsilon_{min}, \Upsilon_{max}$ respectively.

In both problem statements, state is characterized by a Gaussian distribution. Optimality is defined as a goal oriented, minimal jerk trajectory planning. Overlap between two Gaussian distributions is uniquely related to confidence contours at which they are touching each-other. Providing confidence contour is more intuitive and safe than providing raw thresholds of prescribed area of overlap. Hence, input to our algorithm would be Gaussian confidence contours which are not allowed to be penetrated, which is a user specified number.

II. Related work

This section reviews recent advances in MPC for autonomous navigation. The evident advantage of using MPC in motion planning and autonomous navigation has been well demonstrated in ([1],[2], [3], [4]) among many. Formulations along the lines of [1], [2] do a great job in achieving performance in terms of quality of trajectory, computation time and novelty of approach. However, they have been developed for a deterministic setting and hence do not take into consideration the uncertainty in state of the robot and obstacle into their collision avoidance routine. While MPC formulations along the lines of [4] does take into consideration the state uncertainty and demonstrate interesting maneuvers in complex driving scenarios, [4] considers uncertainty only in the state of the obstacle, and the collision avoidance is modeled through a Minkowski sum approach. Considering agent's uncertainty into Minkowski sum formulation would be very cumbersome as Minkowski sum between two ellipses is very complex. [5] takes into consideration the uncertainty of agent and assumes the obstacle to be static and deterministic, it models collision as a measure of entropic distance(Similar to Mahalanobis distance. It is shown in the later section of this paper that formulating probabilistic collision avoidance as an entropic distance may not be the most appropriate approach when uncertainty in the agent and obstacle is considered. [6] attempts to solve for collision avoidance in a multiagent scenario under uncertainty. It achieves it through an RRT framework and a sampling strategy to choose a path corresponding to a desired level of safety confidence. Authors in [7] solve probabilistic multi-agent motion planning problem by considering Pianka's measure([8]) to estimate overlapping probability. This measurement is similar to entropic distance which provides measure of similarity for the given Gaussian statistics. Formulating collision avoidance as a chance constraint is well explored in [9],[10] among the many. [10] demonstrates an efficient way of solving an intractable chance constraint through a series of reformulations. These were built on time scale velocity obstacle concepts [11].

There has been considerable body of work([12], [13], [14]) exploring obstacle avoidance routine through linear chance constraints. However, these methods are modeling linear chance constraints for a static and deterministic obstacles. In [12], convex polyhedral obstacles are encoded by combination of a set of feasible half-spaces, which is algebraically represented by a combination of linear constraints. Obstacle avoidance problem was solved using disjunctive linear programming. Here, risk is uniformly distributed to decompose joint chance constraint. Authors in [12] continue their work in ([13]) where they proposed a risk allocation strategy to optimally distribute risk bounds to individual chance constraints. They additionally prove convexity of iterative risk allocation method for linear systems in [14]. We provide a supplementary document where we illustrate why obstacle avoidance becomes difficult problem to tackle using linear chance constraints based framework when uncertainty is involved in obstacle and agent's states. This method will fail to provide a probabilistic safety bound due to intractability arising in collision avoidance constraint because of obstacle's uncertainty. We provide a more comprehensive explanation of this in our supplementary document. We also provide comparison between our method with bounding box approach and experimentally demonstrate superiority of our method, illustrated through diagrams. We suggest reader to go through the additional document after reading the paper. It can be found at ***https://robotics.iiit.ac.in/people/dhaivat.bhatt/Supplementary.pdf***

At this point, it is important to note that we do not approximate the distribution of chance constraint as a Gaussian through a first order Taylor series approximation method, instead we come up with an alternate way of representing it as an overlap of Gaussians. This way we are differing from most of the existing approaches. We will conclude in the subsequent sections that an overlap of Gaussians method gives rise to maneuvers pertaining to user specified confidence bounds(confidence of safety). Our approach is applicable to static as well as dynamic obstacles. Our mathematical framework adapts well for 2D and 3D navigation tasks. With overlap of Gaussians based modeling, we easily avoid complexity of considering Minkowski sum between two ellipses and other approaches that model probabilistic collision avoidance through entropic distances. We further show its unique application to object following with a simple extension of collision avoidance framework. Such formulation can be used for human-following robots where there is uncertainty involved in human pose/motion estimation apart from error in agent's state estimation. To the best of author's knowledge, this is a first such attempt where non-linear chance constraints are tackled by representing them using theory of overlapping Gaussians and not any other entropic distance.

III. Prerequisite

This section describes deterministic MPC framework, formulated along the lines of [1]. A linear motion model is being considered, and collision avoidance is added as affine constraints. Entire proposed framework is solved as a sequential convex programming(SCP) routine[15].

A. Trajectory optimization in a deterministic setting

In a deterministic trajectory optimization setting, our objective is to reach desired goal in a given amount of time while ensuring a collision-free trajectory. This problem can be modeled by considering a set of cost functions and constraints.

Let the start position of an agent be $X_0 = (x_0, y_0, z_0)$. Our objective is to reach the goal position $G_f = (G_f^x, G_f^y, G_f^z)$ in N time-steps, each time-step of duration τ. Position of an agent at any time instant t_i is $X_{t_i} = (x_{t_i}, y_{t_i}, z_{t_i})$, velocity of an agent at time instant t_i is $V_{t_i} = (v_{t_i}^x, v_{t_i}^y, v_{t_i}^z)$. We have P obstacles in the environment. Their position at time t_i is

defined as $O^j_{t_i} = (o^{xj}_{t_i}, o^{yj}_{t_i}, o^{zj}_{t_i})$, for $\forall$ j= $\{1, 2, 3, ... P\}$. For static obstacles, the obstacle locations will be independent of t_i. The agent and obstacles are approximated as circular objects with radius of the agent being R_{agent} and radius of obstacle j is R_j, $\forall\ j \in \{1, 2, 3, ... P\}$.

$$\underset{V_{t_i}}{\text{argmin}} \quad J = J_{terminal} + J_{smooth} \tag{1a}$$

$$X_{t_{i+1}} = f(X_{t_i}, V_{t_i}) \tag{1b}$$

$$V_{min} \leq V_{t_i} \leq V_{max} \tag{1c}$$

$$a_{min} \leq \frac{V_{t_{i+1}} - V_{t_i}}{\tau} \leq a_{max} \tag{1d}$$

$$C^{t_i}{}_{obs_j}(x_{t_i}, y_{t_i}, z_{t_i}, o^{xj}_{t_i}, o^{yj}_{t_i}, o^{zj}_{t_i}, R_{agent}, R_j) \leq 0 \tag{1e}$$
$$\forall\ i \in \{1, 2, 3, ... N\}\ , \forall\ j \in \{1, 2, 3, ... P\}$$

The above set of equations defines the cost function as well as constraints. Equation 1a describes the objective function.

$$J_{terminal} = (x_{t_N} - G_x)^2 + (y_{t_N} - G_y)^2 + (z_{t_N} - G_z)^2 \tag{2}$$

The terminal cost forces our system to achieve goal-state(G_f) at the end of the trajectory.

$$J_{smooth} = \sum_{i=2}^{N-1} (\frac{(V_{t_{i+1}} + V_{t_{i-1}} - 2V_{t_i})}{\tau^2})^2 \tag{3}$$

The smoothness cost as described above ensures smooth trajectory with minimal jerk. It minimizes the jerk which is modeled as second order finite difference between subsequent linear velocities. This term penalizes sudden deviations in the acceleration profile and ensures smooth velocity transitions. Equation 1b is the process model of the agent. It ensures that control variables and states are adhering the motion model of the agent. Process model of holonomic agent can be described as below,

$$X_{t_i} = f(X_0, V_{t_1}, V_{t_2}, ...V_{t_i}, \tau) = X_0 + \sum_{k=1}^{i} V_{t_k}\tau \tag{4}$$

Equations 1c-1d, represents constraints modeling actuation limitations of the agent. The bounds on linear acceleration and velocity ensures that the actuation capabilities of the agent are not violated. Equation 1e models collision avoidance constraint between obstacle j and agent. For a deterministic setting, this can be a simple euclidean distance constraint as below.

$$\begin{aligned} C^{t_i}{}_{obs_j}(.) = &-(x_{t_i} - o^{xj}_{t_i})^2 - (y_{t_i} - o^{yj}_{t_i})^2 \\ &-(z_{t_i} - o^{zj}_{t_i})^2 + (R_j + R_{agent})^2 \leq 0. \\ \forall\ i \in \{1, 2, 3, ... N\}\ , &\forall\ j \in \{1, 2, 3, ... P\} \end{aligned} \tag{5}$$

Above constraint is purely non-linear in agent's position i.e. X_{t_i} = (x_{t_i}, y_{t_i}, z_{t_i}), which is our variable of interest. We linearize it along the lines of [1] and solve the proposed routine using sequential convex programming. The trajectory optimization routine is then integrated into a model predictive control framework.

IV. Collision avoidance under uncertainty

Robot motions are generally erroneous in nature. There is always some uncertainty associated with the location of the agent. Sensing modality also gives inaccurate estimate of the state of the obstacle/object. Uncertain position estimate of the obstacle leads to erroneous trajectory estimation. In such cases, the constraint in equation 1e takes the form of 6

$$\Pr(C^{t_i}{}_{obs_j}(x_{t_i}, y_{t_i}, z_{t_i}, o^{xj}_{t_i}, o^{yj}_{t_i}, o^{zj}_{t_i}, R_{agent}, R_j) \leq 0) \geq \Delta \tag{6}$$
$$\forall\ i \in \{1, 2, 3, ... N\}\ , \forall\ j \in \{1, 2, 3, ... P\}$$

Constraints of the form 6, are generally known as chance constraints, and in most cases may not have a distribution that can be computed in closed form. The nature of these chance constraints also depends on the form of the deterministic constraints that they are built on. For example chance constraint 6 arising out of 5 can be expressed as an entropic distance. Following three sections provides a comprehensive overview of three important ways to formulate a chance constraint.

A. Theoretical characterization of chance constraint

The chance constraint in 6 can take the form of a transformed distribution of 5 as shown in Equation 7,

$$\int \cdots \int_{V_j} \Pr(X_{t_i}, O^j_{t_i}) dX_{t_i} dO^j_{t_i} \tag{7}$$

Where, X_{t_i} = ($x_{t_i}, y_{t_i}, z_{t_i}$), position of the agent at time t_i. Under state uncertainty, let $X_{t_i} \sim \mathcal{N}\ (\hat{X}_{t_i}, \Sigma^d_{t_i})$ and $O^j_{t_i} \sim \mathcal{N}\ (\hat{O}^j_{t_i}, \Sigma^{o_j}_{t_i})$ be the Gaussian parameterization of the agent and obstacle j positions at time t_i. Then, $\Pr(X_{t_i}, O^j_{t_i})$ takes the following form,

$$\Pr(X_{t_i}, O^j_{t_i}) \ \sim\ \mathcal{N}(\begin{pmatrix} \hat{X}_{t_i} \\ \hat{O}^j_{t_i} \end{pmatrix}, \begin{pmatrix} \Sigma^d_{t_i} & 0 \\ 0 & \Sigma^{o_j}_{t_i} \end{pmatrix}) \tag{8}$$

Here, $\Pr(.)$ symbolizes probability of a random variable. When we substitute equation 8 in equation 7, equation 7 becomes analytically intractable. The closed form solution of equation 7 doesn't exist. Authors in [6] attempted to tackle problem of multi-robot motion planning for differential drive robots, where they numerically evaluate equation 7 over the region of interest. The region of interest here would be set of positions of the agent and obstacles for which collision occurs. Our objective would be to minimize the value of equation 7, which means we want to maximize the probability of collision avoidance. However, one drawback of this procedure is that characterization of such a region(V_j) is generally tough.

There has been a lot of work to characterize the entropic distance between two distributions. One of the commonly used techniques for entropic distances are chi-square distances, Bhattacharyya distances among the many. We describe case of Bhattacharyya distances which is an extension to Mahalanobis distance.

B. Bhattacharyya distance

Bhattacharyya distance [16] gives measure of similarity between two continuous/discrete probability distributions. It attempts to quantify the overlap between two distributions. For Gaussian distributions, Bhattacharyya distance has well-defined analytical formula to capture the overlap. For two multivariate normal distributions, $p_i = \mathcal{N}(\mu_i, \Sigma_i)$ with $i = \{1, 2\}$, Bhattacharyya distance metric is defined as below,

$$\begin{aligned} \mathcal{BC}(p_1, p_2) = &\frac{1}{8}(\mu_1 - \mu_2)^T \Sigma^{-1} (\mu_1 - \mu_2) + \\ &\frac{1}{2} \ln \frac{det(\Sigma)}{\sqrt{det(\Sigma_1) det(\Sigma_2)}} \end{aligned} \tag{9}$$

Where, $\Sigma = (\Sigma_1 + \Sigma_2)/2$. This is an extension of Mahalanobis distance. However, equation 6 can not be completely modeled through similarity/dissimilarity given by such entropic measure. Because the notion of probability is not complete in Bhattacharyya metric and particular distance doesn't have a unique mapping to certain value in probability space. In figure 2, we demonstrate that for different Gaussian distributions with same overlap have different Bhattacharyya distances. Due to this limitation, entropic distance can't be used to model our non-linear chance constraint.

C. Theory of overlapping of Gaussians

The theory of overlap between two Gaussians has been widely studied in [17], which is built upon [18]. The authors in [18] attempts to get an optimal linear separator which minimizes misclassification error when the objective is to classify the sample as coming from one of the several populations. We briefly state the theory to get approximate estimate of component of overlap between two Gaussians.

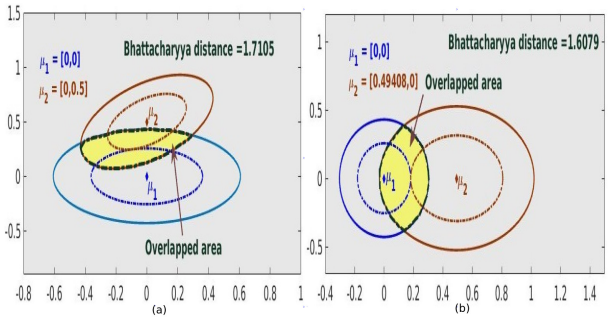


Fig. 2. Diagrammatic explanation of why Bhattacharyya distance is not appropriate metric to model chance constraint problem. In this figure, we have taken 2 sets of Gaussian distribution pairs which are touching at same confidence contour of 80.51%. Meaning, area of overlap between two Gaussians shaded in yellow is same for both sets of Gaussian distributions. For figure 2(a), $\Sigma_1 = \left(\begin{smallmatrix}0.04 & 0\\ 0 & 0.02\end{smallmatrix}\right)$, $\Sigma_2 = \left(\begin{smallmatrix}0.02 & 0.01\\ 0.01 & 0.02\end{smallmatrix}\right)$. Bhattacharyya distance evaluated using equation 9 for this set of co-variances turns out to be 1.7105. While, for figure 2(b), $\Sigma_1 = \left(\begin{smallmatrix}0.01 & 0\\ 0 & 0.02\end{smallmatrix}\right)$, $\Sigma_2 = \left(\begin{smallmatrix}0.03 & 0\\ 0 & 0.03\end{smallmatrix}\right)$. Value of Bhattacharyya distance for figure 2(b) is 1.6079. Hence, for same amount of overlap between two sets of Gaussian distributions, Bhattacharyya distances are turning out to be different. Areas shaded with yellow in figure indicate amount of overlap.

The linear separator proposed in[18] works for two Gaussians of dimension $d \geq 1$.

Let the linear separator(a hyperplane in d dimensional space) be $\alpha^T x = \beta$ where $\alpha, x \in \mathbb{R}^d$ and $\beta \in \mathbb{R}$. $\alpha^T x \leq \beta$ classifies x into first cluster and $\alpha^T x > \beta$ classifies x into second cluster. We will briefly explain the procedure to obtain α, β and estimate the area of overlap(Υ) between two Gaussian distributions $\mathcal{N}(\mu_i, \Sigma_i)$ with $i = \{1, 2\}$. Here, x is coming from one of the above two Gaussian distributions and $\alpha^T x$ is a linear transformation which transforms the original distribution into uni-variate normal distribution. The probability of misclassification when x is coming from first distribution is,

$$\mathbb{P}_1(\alpha^T x > \beta) = \mathbb{P}_1\left(\frac{\alpha^T x - \alpha^T \mu_1}{\sqrt{\alpha^T \Sigma_1 \alpha}} > \frac{\beta - \alpha^T \mu_1}{\sqrt{\alpha^T \Sigma_1 \alpha}}\right)$$
$$\mathbb{P}_1(\alpha^T x > \beta) = 1 - \Phi\left(\frac{\beta - \alpha^T \mu_1}{\sqrt{\alpha^T \Sigma_1 \alpha}}\right) = 1 - \Phi(\eta_1) = \mathbb{P}_1(\eta_1) \tag{10}$$

Similarly, probability of misclassification when sample x belongs to population 2 equals,

$$\mathbb{P}_2(\alpha^T x \leq \beta) = \mathbb{P}_2\left(\frac{\alpha^T x - \alpha^T \mu_2}{\sqrt{\alpha^T \Sigma_2 \alpha}} \leq \frac{\beta - \alpha^T \mu_2}{\sqrt{\alpha^T \Sigma_2 \alpha}}\right)$$
$$\mathbb{P}_2(\alpha^T x \leq \beta) = 1 - \Phi\left(\frac{\alpha^T \mu_2 - \beta}{\sqrt{\alpha^T \Sigma_2 \alpha}}\right) = 1 - \Phi(\eta_2) = \mathbb{P}_2(\eta_2) \tag{11}$$

Where, $\eta_1 = \frac{\beta - \alpha^T \mu_1}{\sqrt{\alpha^T \Sigma_1 \alpha}}$ and $\eta_2 = \frac{\alpha^T \mu_2 - \beta}{\sqrt{\alpha^T \Sigma_2 \alpha}}$ are two scalars. Φ in equations 10-11 denotes a cumulative distribution function for a univariate standard normal distribution. Φ is a monotonically increasing function. Our objective is following,

$$max(\mathbb{P}_1(\eta_1), \mathbb{P}_2(\eta_2)) \rightarrow \min_{\alpha \in \mathbb{R}^d \beta \in \mathbb{R}} \tag{12a}$$
$$min(\eta_1, \eta_2) \rightarrow \max_{\alpha \in \mathbb{R}^d \beta \in \mathbb{R}} \tag{12b}$$

Equations 12a-12b are equivalent due to monotonic nature of the Φ. The objective is to maximize the minimum of (η_1, η_2). The objective is to find α, β, which will minimize the maximum probability of misclassification. Analytical characterization of α, β in terms of $\mu_1, \Sigma_1, \mu_2, \Sigma_2$ can be expressed as following,

$$\alpha = (\lambda_1 \Sigma_1 + \lambda_2 \Sigma_2)^{-1} (\mu_2 - \mu_1) \tag{13a}$$
$$\beta = \alpha^T \mu_1 + \lambda_1 \alpha^T \Sigma_1 \alpha = \alpha^T \mu_2 - \lambda_2 \alpha^T \Sigma_2 \alpha \tag{13b}$$

Here, λ_1 and λ_2 are two scalars and resulting procedure to estimate these parameters is referred to as **minmax procedure**. The minmax procedure is an admissible procedure[18] when $\eta_1 = \eta_2$. For admissible procedure, $\lambda = \lambda_1 = 1 - \lambda_2$. If we substitute analytical characterization of α, β in η_1, η_2, the following equality must hold for admissible procedure.

$$\eta_1^2 - \eta_2^2 = \alpha^T [\lambda^2 \Sigma_1 - (1 - \lambda)^2 \Sigma_2] \alpha = 0 \tag{14}$$

The above criterion is a necessary condition to get the best approximation of amount of overlap. We will call λ *"overlap parameter"*, as it is a deciding factor which completely characterizes the overlap for a given first and second order Gaussian moments. Here, value of overlap parameter λ is determined heuristically using equation 14 as a base condition. Algorithm 1 is used to determine optimal value of λ. The overlap parameter λ completely dictates the linear separator parameters α and β. Once we get optimal value of λ, we can estimate linear separator parameters α, β. Since the procedure is admissible, we can compute $\mathbb{P}_1(\eta_1)$ = $\mathbb{P}_2(\eta_2)$ = $\mathbb{P}_{minmax}$. The amount of overlap(Υ) can be computed as below,

$$\eta_1 = h_1(\alpha, \beta), \quad \alpha = h_2(\lambda), \quad \beta = h_3(\lambda)$$
$$\Upsilon = \mathbb{P}_1(\eta_1) + \mathbb{P}_2(\eta_2) = \mathbf{2}\mathbb{P}_{minmax} = h(\mu_1, \mu_2, \Sigma_1, \Sigma_2, \lambda) \tag{15}$$

Here, we can notice that component of overlap(Υ) is parameterized by linear separator parameters α and β. Overlap parameter λ is completely dictating α and β as evident from equation 13. Hence, λ dictates the amount of overlap Υ. Here, λ is determined through an iterative procedure. As seen in figure 3, the overlap is determined by substituting for different values of λ in 15, a typical iterative routine settles for some value of λ, when the overlap is correctly determined, this can be noticed in Fig 3(d)-3(f). When we estimate area of overlap for the two sets of Gaussian distributions in figure 2 using algorithm 1, the area of overlap turns out to be Υ = 0.0706, which is same for both sets of Gaussian distributions shown in figures 2(a) and 2(b). Purpose of figure 3 is to empirically establish our finding that any two sets of Gaussian distributions touching at same confidence contours have same amount of overlap calculated according to algorithm 1. We exploit this finding to uniquely determine prescribed area of overlap based on user specified confidence contour threshold. In other words, there is a unique mapping between area of overlap of two k variate Gaussians and confidence contours c_t(where they are touching each other). The current scope of this paper explores this concept for $k = 2, 3$. We create a table that would give us unique value of overlap for a given value of c_t. We show relation between contour of touch(c_t) and area of overlap(Υ), in the figure 3(g)-3(h).

Algorithm 1 OverlapEstimation(μ_1, Σ_1, μ_2, Σ_2, *precision*)

```
initialize increment, criterion, λ
repeat
    calculate α with 13a
    calculate criterion with 14
    if criterion > precision
        then λ ← λ - increment
    if criterion < -precision
        then λ ← λ + increment
    increment ← increment/2
until -precision ≤ criterion ≤ precision
return λ
```

V. Probabilistic Collision Avoidance as Overlap between Two Gaussians

Equation 7 represents analytical expression of chance constraint defined in section IV. Our objective is to maximize the probability of collision avoidance. Here, we propose a novel formulation by modeling a non-linear chance constraint as overlap between component of two Gaussian distributions. We demonstrate in figure 4 that minimization of overlap

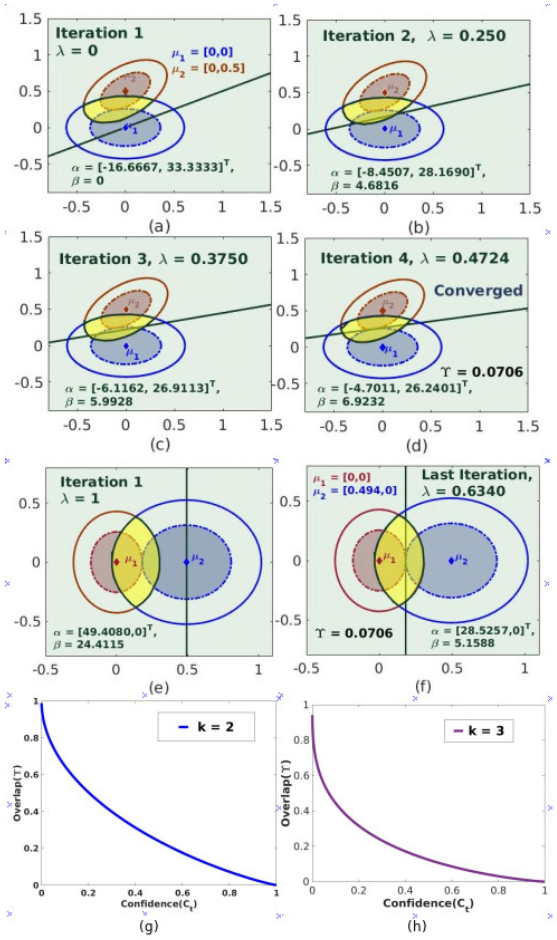


Fig. 3. This figure demonstrates how the values of overlap parameter λ evolves as algorithm 1 executes. It starts with initialized value of λ and ultimately converges to the value where the linear separator divides the two Gaussian distributions at the same confidence interval. From figure 3(a) -3(d), the Gaussian configuration considered is same as that of figure 2(a). Both the Gaussian distributions are touching at confidence contour corresponding to 80.51%. Equality constraint of equation 14 ensures that both the Gaussians are touching at the same confidence interval. The line drawn in the figure is $\alpha^T x = \beta$. We can see orientation and position of the line evolving as overlap parameter λ converges. For converged value of λ, we can see that the optimal linear separator passes through the point of touch of the two Gaussian distributions. Figure 3(e) -3(f) shows λ converging for Gaussian configuration considered in figure 2(b). We can see value of overlap turning out to be same for both sets as shown in figure 3(d)-3(f).

between two Gaussians is analogous to maximization of probability of collision avoidance.

A. Trajectory optimization with chance constraints

In section III, we discussed deterministic trajectory optimization routine. In this section, we reformulate this routine to accommodate state uncertainty of the agent and obstacle. We express collision avoidance constraint as desired measure of overlap between Gaussian populations of an agent and obstacle. The remodeling will use theory of overlapping of Gaussians described in section IV-C. Since there is an uncertainty in agent's position, we will redefine $J_{terminal}$ as below,

$$J_{terminal} = (\hat{X}_{t_N} - G_f)(\Sigma^d_{t_N})^{-1}(\hat{X}_{t_N} - G_f)^T \tag{16}$$

Equation 16 is Mahalanobis distance, which characterizes the number of standard deviations a point is away from mean of a distribution. Equation 16 will minimize number of standard deviations goal point is away from the mean position of the agent at the end of the trajectory.

Here, we are assuming no uncertainty in the actuation. Further, we assume that belief propagation for agent and obstacles for a given time-horizon is known. Hence, minimization of overlap between agent and obstacle populations can be thought of as minimum number of standard deviations an agent should deviate from its path to ensure collision free trajectory. While planning the trajectory, our constraint is to ensure that the minimum value of c_{t_i} for $i \in \{1, 2..., N\}$ is larger than certain threshold c_{min}, a user specified confidence contour which can't be penetrated. Which is analogous to saying that the overlap between an agent and obstacle at any time instant should not be greater than the prescribed overlap threshold Υ_{max}. c_{t_i} represents confidence contour percentage at which agent and obstacle/object are touching each other at time t_i.

1) Reformulation of collision avoidance constraint: Our goal is to reach G_f from the start position X_0 in N time-steps, each time-step of duration τ. Here, our objective is to find optimal set of velocity commands $V_{t_i} = (v^x_{t_i}, v^y_{t_i}, v^z_{t_i})$ which would satisfy our constraints as well as minimize the cost. We will be using the process model of the agent explained in section III.

Let the overlap between agent and obstacle j at time instance t_i be $\Upsilon^j_{t_i}$, which is dictated by overlap parameter $\lambda^j_{t_i}$. then,

$$\hat{X}_{t_i} = f(\hat{X}_{t_0}, V_{t_1}, V_{t_2}, V_{t_3}....V_{t_i}) \tag{17a}$$
$$\Upsilon^j_{t_i} = g_1(\hat{X}_{t_i}, \Sigma^d_{t_i}, \hat{O}^j_{t_i}, \Sigma^{o_j}_{t_i}, \lambda^j_{t_i}) \tag{17b}$$

Equation 17b is completely analogous to equation 15. In equation 17b, overlap is expressed in terms of agent/obstacle means, their corresponding uncertainties and overlap parameter. Mean position of the agent is expressed in terms of control commands$(V_{t_1}, V_{t_2}, ...V_{t_i})$ as explained in motion model(equation 4). Hence, equation 17b is completely parameterized by control commands$(V_{t_1}, V_{t_2}, ...V_{t_i})$ and overlap parameter$(\lambda^j_{t_i})$. Similarly, We can express condition to admissible procedure(equation 18a) in terms of control and overlap parameter as below,

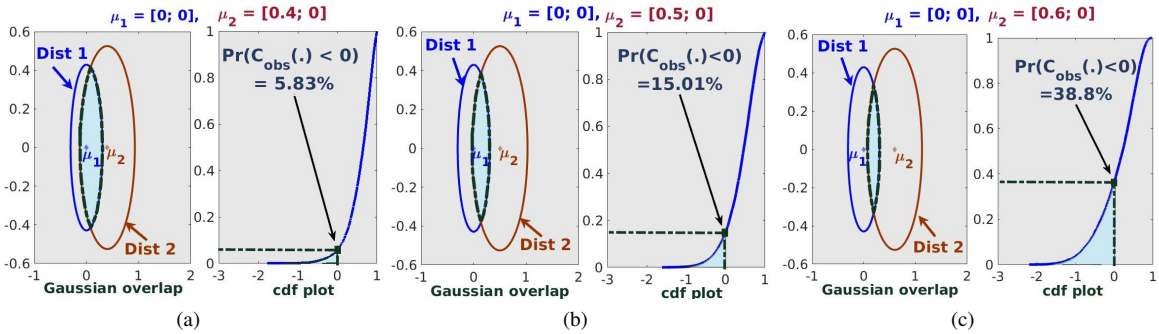
$$\left.\begin{aligned}(\eta^d_{t_i})^2 - (\eta^{o_j}_{t_i})^2 &= g_2(\hat{X}_{t_i}, \Sigma^d_{t_i}, \hat{O}^j_{t_i}, \Sigma^{o_j}_{t_i}, \lambda^j_{t_i}) \\ (\eta^d_{t_i})^2 - (\eta^{o_j}_{t_i})^2 &= f_1(\lambda^j_{t_i}, V_{t_1}, V_{t_2}, V_{t_3}...V_{t_i})\end{aligned}\right\} \tag{18a}$$
$$\Upsilon^j_{t_i} = 2\mathbb{P}^d_{t_i}(\eta^d_{t_i}) = 2\mathbb{P}^{o_j}_{t_i}(\eta^{o_j}_{t_i}) = f_2(\lambda^j_{t_i}, V_{t_1}, V_{t_2}...V_{t_i}) \tag{18b}$$

Equation 18a-18b are in accordance with equation 14-15, equation 18a models necessary condition for the procedure to be admissible in nature. It is important to note that in equations 17b and 18a, $\Sigma^d_{t_i}, \hat{O}^j_{t_i}$ and $\Sigma^{o_j}_{t_i}$ are known quantities. Hence, the only unknowns we are interested in solving are $\hat{X}_{t_i}$ and $\lambda^j_{t_i}$. Where, $\hat{X}_{t_i}$, mean of the agent at time t_i, which is a function of control commands according to motion model equation.

The maximum allowed overlap Υ_{max} is uniquely related to minimum number of standard deviations not allowed to be penetrated(c_{min}). c_{min} is expressed in terms of confidence intervals directly. So for a particular confidence interval, c_{min} will have a unique scalar value. Hence, a unique Υ_{max} value as explained in section IV-C. So, a chance constraint in terms of overlap between two Gaussians can be expressed as below.

$$C^{t_i}{}_{obs_j}(.) = \begin{cases}(\eta^d_{t_i})^2 - (\eta^{o_j}_{t_i})^2 = f^{lin}_1(\lambda^j_{t_1}, V_{t_1}, ..., V_{t_i}) = 0 \\ \Upsilon^j_{t_i} = f^{lin}_2(\lambda^j_{t_1}, V_{t_1}, ..., V_{t_i}) \leq \Upsilon_{max}\end{cases} \tag{19}$$
$$\forall\ i \in \{1, 2, 3, ...\ N\}\ ,\ \forall\ j \in \{1, 2, 3, ...\ P\}$$

Both the constraints of collision avoidance constraint(equation 19) are parameterized by velocity



(a) (b) (c)

Fig. 4. We establish an analogy between minimization of overlap between Gaussians as maximization of probability of collision avoidance modeled according to equation 6. Through an illustration, we demonstrate that as the amount of overlap between two distributions decreases, probability of collision avoidance increases. This is aptly conveyed through figure 4(a)-4(b)-4(c). For example, in figure 4(a), for a significant amount of overlap, the area below **0** in the cdf plot is very less. However, in figure 4(b) and 4(c), as the overlap decreases(achieved through incrementally separating μ_2 from μ_1), the corresponding area below **0** observed in the cdf plots significantly increases. Thus, it is clear that as the overlap between these two distributions decreases, the probability of collision avoidance[6](conveyed through cdfs) increases. The cdf plots of equation 6 were generated through *ecdf()* function of Matlab. The covariances considered for this demonstration are, $\Sigma_1 = \begin{pmatrix} 0.01 & 0 \\ 0 & 0.02 \end{pmatrix}$ and $\Sigma_2 = \begin{pmatrix} 0.03 & 0 \\ 0 & 0.03 \end{pmatrix}$.

controls($V_{t_1}, V_{t_2}, ...V_{t_i}$) and overlap parameter($\lambda^j_{t_i}$). $\Upsilon^j_{t_i} \leq \Upsilon_{max}$ ensures that the agent is avoiding the obstacle with certain minimum confidence. $(\eta^d_{t_i})^2 - (\eta^{o_j}_{t_i})^2 = 0$ ensures that the overlap parameter($\lambda^j_{t_i}$) we get through SCP routine is optimal and satisfies condition to admissible procedure. Closed form expressions for $f_1(.)$ and $f_2(.)$ are functions of optimization variables($V_{t_1}, V_{t_2}, ..., V_{t_N}, \lambda^j_{t_i}$) which are computed using mathematica[19]. $f_1(.)$ and $f_2(.)$ are non-linear in our variables of interest. We linearize them as shown below,

$$f_1^{lin}() = \bar{f}_1() + \sum_{k=1}^{i} \nabla_{V_{t_k}} (V_{t_k} - \bar{V}_{t_k}) + \nabla_{\lambda^j_{t_i}} (\lambda^j_{t_i} - \bar{\lambda}^j_{t_i}) \quad (20)$$

$$f_2^{lin}() = \bar{f}_2() + \sum_{k=1}^{i} \nabla_{V_{t_k}} (V_{t_k} - \bar{V}_{t_k}) + \nabla_{\lambda^j_{t_i}} (\lambda^j_{t_i} - \bar{\lambda}^j_{t_i}) \quad (21)$$

$$\forall\ i \in \{1, 2, 3, ...\ N\}\ , \forall\ j \in \{1, 2, 3, ...\ P\}$$

$f_1^{lin}(.)$ and $f_2^{lin}(.)$ are affine approximations of $f_1(.)$ and $f_2(.)$. $\nabla_{V_{t_k}}$ and $\nabla_{\lambda^j_{t_i}}$ are partial derivatives with respect to V_{t_k} and $\lambda^j_{t_i}$ respectively.

2) Trajectory optimization algorithm: We outline complete trajectory optimization algorithm built on above scheme. Let the trajectory of obstacle j be denoted by $\Omega_j = \{O^j_{t_1}, O^j_{t_2}, O^j_{t_3}, ...O^j_{t_N}\}$ and trajectory of all P obstacles be $\mathbf{\Pi_P} = \{\Omega_1, \Omega_2, \Omega_3, ...\Omega_P\}$. Let overlap parameter between obstacle j's distribution and agent's distribution for N timesteps be $\mathbf{\Lambda_j} = \{\lambda^j_{t_1}, \lambda^j_{t_2}...\lambda^j_{t_N}\}$, $j \in \{1, 2, 3, ...\ P\}$ and controls we are solving for be $\mathbf{V(t)} = \{V_{t_1}, V_{t_2}, V_{t_3}, ..., V_{t_N}\}$. Algorithm 2 outlines proposed SCP routine where we jointly solve for control($\mathbf{V(t)}$) and overlap parameter($\mathbf{\Lambda_j}$). In this algorithm, linearization control points $\mathbf{\bar{V}(t)} = \{\bar{V}_{t_1}, \bar{V}_{t_2}, \bar{V}_{t_3}, ..., \bar{V}_{t_N}\}$ are obtained by running the algorithm without collision avoidance constraints. That will give us an optimal trajectory from start to goal without any deviation. While overlap parameter linearization points are initialized with 0.5. Which implies that, $\mathbf{\bar{\Lambda}_j} = \{0.5, 0.5, ..., 0.5\}$. Subsequently, solution of previous iteration is used as linearization point in current iteration, which is implied in stage 4 and 5 of algorithm 2.

VI. Results and Discussions

We construct a model predictive control framework by using proposed probabilistic trajectory optimization routine as a base. Our proposal has been extensively evaluated for wide range of safety critical configurations. We show two such challenging situations to evaluate our proposal. We use Matlab based CVX[20] to prototype many of these scenarios. For a faster implementation, we use python based CVXOPT[21]. The simulations are carried out using Rotors[22], which is a micro aerial vehicle simulation framework built in gazebo[23]. The proposed approach is implemented in a model predictive control framework(MPC) along the lines of [24], [24] uses the idea of a receding horizon as the basis for building an MPC. We plan for a finite horizon and upto some intermediate way-point along the original trajectory.

Algorithm 2 ProbabilisticTrajOpt(Υ_{max}, Π_P, Σ_{agent}, $\Sigma_{obstacle}$)

1: **Initialization:** Guess for $\mathbf{\bar{\Lambda}^k_j(t)}$, $\mathbf{\bar{V}^k(t)}$, iteration counter k = 0
2: $\bar{D}^k(t)$ = $InitializeTrajectory(\mathbf{\bar{V}}^k(\mathbf{t}))$
3: **while** $\mid J_{k+1} - J_k \mid \geq \delta$ **do**

$$\mathbf{V}^k(\mathbf{t}), \mathbf{\Lambda^k_j(t)} = \text{argmin} \quad J_k$$

subject to

$$X_{t_i+1} = f(X_{t_i}, V_{t_i})$$
$$V_{min} \leq V_{t_i} \leq V_{max}$$
$$a_{min} \leq (\frac{V_{t_i+1} - V_{t_i}}{\tau}) \leq a_{max}$$
$$C_{obs_j}(\mathbf{\Lambda^k_j(t)}, \mathbf{V}^k(\mathbf{t})) \leq \mathbf{0},\ \forall\ j = \{1, 2, 3, ...\ P\}$$

4: $\mathbf{\bar{\Lambda}^k_j(t)} \leftarrow \mathbf{\Lambda^k_j(t)}$
5: $\mathbf{\bar{V}}^k\mathbf{(t)} \leftarrow \mathbf{V}^k\mathbf{(t)}$
6: $k \leftarrow k + 1$
7: **end while**

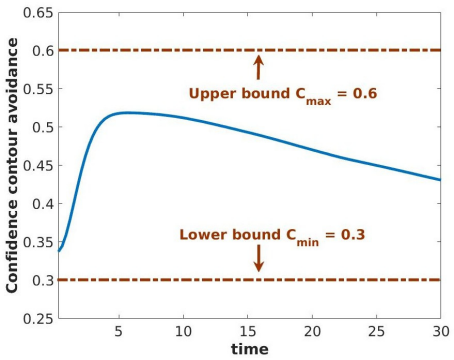


Fig. 5. This figure shows how confidence contour of touch($\mathbf{c_{t_i}}$) is changing over time for setting considered in VI-B. Throughout the trajectory, our agent is able to satisfy both, lower and upper-bound constraints. Initial rise in plot suggests that agent was slowing down first, and then it gradually accelerated to satisfy upper-bound constraint.

Intermediate way-points are placed at regular intervals to avoid extreme deviations in agent trajectory. In this section, we show results for two interesting applications. The detailed video of simulations under various safety critical situations is provided at ***https://robotics.iiit.ac.in/people/dhaivat.bhatt/video.html***

A. Multiple obstacle avoidance: Antipodal configuration

In this experiment, we show 3 obstacles attacking our agent in an antipodal configuration. The agent detects obstacles at sensing range of $S_r = 10$ meters. The acceleration bounds a_{min}, a_{max} are $-0.5\ m/s^2$, $0.5\ m/s^2$ respectively. Value of V_{min} and V_{max} are $0\ m/s$, $3.0\ m/s$ respectively. We keep a planning horizon of 28 steps, each of duration $\tau = 0.3$ seconds. We keep re-planning at every 0.3 seconds. Radius of agent/obstacles is taken as 0.5 meters. For this configuration, we consider $c_{min} = 90\%$ as our user specified confidence contour bound. In other words, our objective is to ensure that at least 90% confidence contours of the agent and obstacles do not penetrate each other during entire journey of the agent. As soon as it detects the obstacles, it starts deviating from its trajectory and 90% confidence contour of the agent avoids 90% confidence contours of all 3 obstacles. The position uncertainties considered are $\Sigma_{agent} = \Sigma_{obs1} = \Sigma_{obs2} = \Sigma_{obs3} = \begin{pmatrix} 0.02 & 0 & 0 \\ 0 & 0.02 & 0 \\ 0 & 0 & 0.02 \end{pmatrix}$. With this configuration, we show comprehensive results in figure 6. Sequence of images in figure 6 shows some snaps during obstacle avoidance maneuver, the navigation was successful and user specified lower bound was respected throughout the trajectory. As shown in confidence plots in figure 6(d)-6(e)-6(f), penetration for violet and dark green obstacles was at 95%, while for blue obstacle, it was 98%.

B. Object following in constrained corridor

We show another interesting application of our proposal. If agent and object of interest are entering in a constrained corridor, apart from putting lower bound c_{min}, we can also put upper bound(c_{max}) in such tight spaces. Putting upper bound will ensure that agent is not slowing down too much. Sub-constraint 2 of equation 19 will take a form like $\Upsilon_{min} \leq \Upsilon^j_{t_i} \leq \Upsilon_{max}$. Through demonstration, we advocate the usage of upper bound constraint in tight spaces or in situations where we need to follow/track object of interest. Upper bound ensures that agent is within certain range of object, which will reduce deviation in agent's trajectory, thus ensuring no collision with surrounding walls. We consider a case where agent and obstacle are entering in a constrained corridor at same time. In this case, user specified confidence contour bounds are $c_{min} = 30\%$ and $c_{max} = 60\%$. In other words, at least 30% confidence contours of agent and object can not penetrate into each other, while 60% confidence contours can't have 0 overlap at any time instant during the trajectory. Our planning horizon is of 40 timesteps, each of duration $\tau = 0.3$ seconds. Uncertainty being considered is, $\Sigma_{agent} = \begin{pmatrix} 0.02 & 0.01 & 0 \\ 0.01 & 0.02 & 0 \\ 0 & 0 & 0.02 \end{pmatrix}$ and $\Sigma_{object} = \begin{pmatrix} 0.03 & 0.02 & 0 \\ 0.02 & 0.03 & 0 \\ 0 & 0 & 0.02 \end{pmatrix}$. Before solving an MPC, these matrices are scaled up to incorporate radius of agent and obstacle. We consider non-isotropic uncertainty for this demonstration and show efficacy of our algorithm under tight spaces. In figure 7, We show snippets of various time-instances. The walls are modeled as stationary obstacles. For example, in figure 7(a), agent and object are entering in the corridor. Both upper bound and lower bound constraints are enforced and we can see that agent is able to maintain sufficient distance from the wall as well as the object while respecting the constraints. An absence of upper-bound constraint results in slowing down of the agent and we encounter a longer time for flight completion. Having upper bound favorably changes the velocity profile to complete trajectory in faster time. This shows usefulness of our proposal in tightly bounded spaces. We can use such modeling in object following. In figure 5, we show confidence plot for entire trajectory.

VII. Conclusion and Future Work

In this paper, a novel approach to dynamic collision avoidance and object following under uncertainty in state of robot and obstacles, modeled using Gaussian distribution, has been proposed. It has been derived by using area of overlap of the Gaussian distributions, which has unique characterization for a given confidence contour. The proposed algorithm, integrated with Linear MPC, optimizes over velocity profile and overlap parameter space to generate a navigation path in constrained dynamic environment. The paper puts forward results for two safety critical configurations. In future, we intend to model actuation dynamics into overlap of Gaussian framework and attempt to solve for challenging scenarios with unbounded co-variances. We can also model co-variance as a function of control and include into overlap of Gaussians framework. Another interesting task would be to extend this formulation for motion planning of non-holonomic vehicles. We can extend this formulation for multi-agent motion planning, where a centralized optimization routine will solve for control sequence of all agents with a guaranteed risk bound.

References

[1] M. Babu, R. R. Theerthala, A. K. Singh, B. Gopalakrishnan, and K. M. Kirshna, "Model predictive control for autonomous driving considering actuator dynamics," 2018.

[2] D. Kim, J. Kang, and K. Yi, "Control strategy for high-speed autonomous driving in structured road," in *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE, 2011, pp. 186–191.

[3] C. Katrakazas, M. Quddus, W.-H. Chen, and L. Deka, "Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 416–442, 2015.

[4] W. Schwarting, J. Alonso-Mora, L. Pauli, S. Karaman, and D. Rus, "Parallel autonomy in automated vehicles: Safe motion generation with minimal intervention," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1928–1935.

[5] G. Garimella, M. Sheckells, and M. Kobilarov, "Robust obstacle avoidance for aerial platforms using adaptive model predictive control," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5876–5882.

[6] J. Van Den Berg, P. Abbeel, and K. Goldberg, "Lqg-mp: Optimized path planning for robots with motion uncertainty and imperfect state information," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 895–913, 2011.

[7] I. Postlethwaite and M. Kothari, "Multi-agent motion planning for nonlinear gaussian systems," *International Journal of Control*, vol. 86, no. 11, pp. 2075–2089, 2013.

[8] R.-P. Lu, E. P. Smith, and I. Good, "Multivariate measures of similarity and niche overlap," *Theoretical Population Biology*, vol. 35, no. 1, pp. 1–21, 1989.

[9] B. Gopalakrishnan, A. K. Singh, M. Kaushik, K. M. Krishna, and D. Manocha, "Prvo: Probabilistic reciprocal velocity obstacle for multi robot navigation under uncertainty," 2017.

[10] B. Gopalakrishnan, A. K. Singh, and K. M. Krishna, "Closed form characterization of collision free velocities and confidence bounds for non-holonomic robots in uncertain dynamic environments," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 4961–4968.

[11] A. K. Singh and K. M. Krishna, "Reactive collision avoidance for multiple robots by non linear time scaling," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*. IEEE, 2013, pp. 952–958.

[12] L. Blackmore, H. Li, and B. Williams, "A probabilistic approach to optimal robust path planning with obstacles," in *American Control Conference, 2006*. IEEE, 2006, pp. 7–pp.

[13] M. Ono and B. C. Williams, "An efficient motion planning algorithm for stochastic dynamic systems with constraints on probability of failure." in *AAAI*, 2008, pp. 1376–1382.

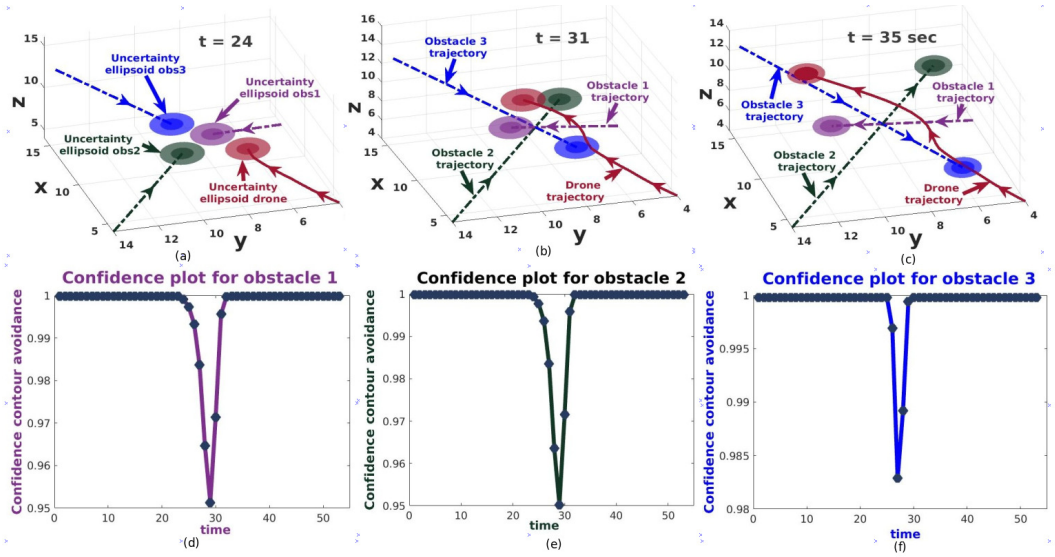


Fig. 6. Antipodal setting: The agent adopts a maneuver relating to a particular confidence of safety when it encounters obstacles in its sensing range. This is clearly shown in figure 6(a)-6(c). Figure 6(a), shows the situation, where the agent encounters obstacles, within its sensor range and starts taking appropriate control actions.Figure 6(b), highlights the resultant maneuver, that the agent adopts to achieve a targeted level of safety. Figure 6(c) shows the goal reaching ability of the agent, after avoiding obstacles. The lighter ellipsoidal shades in these figures represent the uncertainty region encompassing the mean positions of the agent and obstacles(filled with darker shades). Figures 6(d)-6(e)-6(f), shows the plots of confidence intervals for the collision avoidance maneuvers the that agent adopted in figures 6(a)-6(c). Our constraint was to ensure that 90% confidence contours of our agent avoids at least 90% confidence contours of all obstacles. From figures 6(d)-6(e)-6(f), we can observe that maximum overlap between the agent and any obstacle is 5%, i.e. only the 95% confidence contours grace each-other.

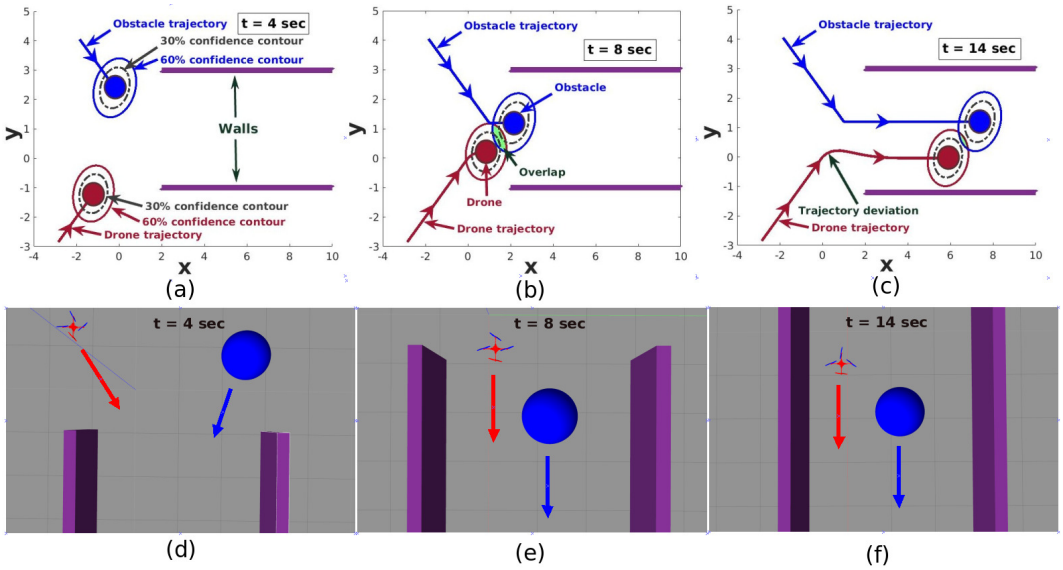


Fig. 7. Object following in constrained corridor: In figure 7(a), confidence contour for 30% and 60% are shown for both, agent and object. agent senses presence of obstacle at time $t = 4sec$. As agent enters in corridor, lower and upper bound constraints are enforced. Which ensures minimum risk behavior. For example, in figure 7(b)- 7(c), we can see overlap between agent and obstacle shaded in green for 60% confidence contours. While, 30% confidence contours which correspond to lower bound never penetrate each other as visible in figure 7(b)-7(c). Figure 7(e) shows top view of figure 7(b) in gazebo. Under such tightly bounded spaces, we observe that agent is able to safely maneuver constraints without crashing into walls. Trajectory deviation shown in 7(c) depicts that agent can maneuver in a way that would make it stay reasonably behind the object of interest while satisfying both lower and upper bound constraints. Figures 7(d)-7(e)-7(f) are gazebo results of figures 7(a)-7(b)-7(c) in bird's eye view.

[14] M. Ono, "Iterative risk allocation: A new approach to robust model predictive control with a joint chance constraint," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, 2008, pp. 3427–3432.

[15] S.Boyd, "sequential convex programming," https://web.stanford.edu/class/ee364b/lectures/seq_slides.pdf, 2008.

[16] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bull. Calcutta Math. Soc.*, vol. 35, pp. 99–109, 1943.

[17] E. Nowakowska, J. Koronacki, and S. Lipovetsky, "Tractable measure of component overlap for gaussian mixture models," *arXiv preprint arXiv:1407.7172*, 2014.

[18] T. W. Anderson, R. R. Bahadur, *et al.*, "Classification into two multivariate normal distributions with different covariance matrices," *The annals of mathematical statistics*, vol. 33, no. 2, 1962.

[19] W. R. Inc., "Mathematica, Version 11.0," champaign, IL, 2016.

[20] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[21] M. Andersen and L. Vandenberghe, "CVXOPT: A python package for convex optimization, version 1.1.9," http://cvxopt.org/index.html, 2016.

[22] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, "Rotorsa modular gazebo mav simulator framework," in *Robot Operating System (ROS)*. Springer, 2016, pp. 595–625.

[23] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 3. IEEE, 2004, pp. 2149–2154.

[24] A. Raemaekers, "Design of a model predictive controller to control uavs."